

Numerical optimization and optimal control for dynamical systems [146442]

Lecture Notes

Enrico Bertolazzi



Copyright © 2025 Enrico Bertolazzi

University of Trento — Department of Industrial Engineering

Via Sommarive 9, 38123 Povo (Trento), Italy

Licensed under the Creative Commons Attribution-NonCommercial 4.0 License (the "License"). You may not use this file except in compliance with the License. You may obtain a copy of the License at https://creativecommons.org/licenses/by-nc-sa/4.0. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "as is" basis, without warranties or conditions of any kind, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Contents

I Constrained Minimization

1 Mathematical Preliminaries 7

1.1 Taylor expansions and regularity 7

1.1.1 Restriction to a fixed direction 8

1.2 Quantitative first-order remainder for a Lipschitz gradient 8

1.3 Cubic remainder for a Lipschitz Hessian 9

1.4 Global and local minima 10

1.5 Conditions for an unconstrained local minimum 11

1.5.1 First-order necessary condition 11

1.5.2 Symmetric matrices and quadratic forms 11

1.5.3 Rayleigh quotient and eigenvalue bounds 11

1.5.4 Second-order necessary condition 12

1.5.5 Second-order sufficient condition 12

1.6 Examples: classification of stationary points 13

1.6.1 Example 1: strict minimum (positive definite Hessian) 13

1.6.2 Example 2: strict maximum (negative definite Hessian) 14

1.6.3 Example 3: saddle point (indefinite Hessian) 14

1.6.4 Example 4: degenerate stationary point (inconclusive Hessian) 14

2 Lagrange Multipliers 17

2.1 Summary of constrained minimization with proofs 21

3 Sylvester's Criterion 27
3.1 Why use Sylvester's criterion 29
3.2 Positive semidefinite matrices 31
3.3 Summary 33

4 Inequality Constraints and KKT Conditions 35
4.1 Inequality Constraints 35
4.1.1 Statement of the KKT conditions 37
4.2 KKT Conditions 48
4.2.1 Feasible set and active constraints 48
4.2.2 Cones in $\mathbb{R}^n$ and local feasible geometry 48
4.2.3 Fritz–John conditions 55
4.2.4 Constraint qualifications 55
4.2.5 KKT conditions under a constraint qualification 58
4.3 Notes and references 62
4.4 Chapter summary 62

II ODE

5 Introduction to ODE 67
5.1 Application of ODEs 67
5.2 Classification of ODEs 71
5.2.1 Order of an ODE 71
5.2.2 Linearity of an ODE 71
5.2.3 Autonomous vs. Non-autonomous ODEs 71
5.3 Solution Techniques 72

6 ODE Theory and Properties 73
6.1 Notation Conventions 73
6.2 Examples of ODEs 74
6.2.1 Scalar ODEs 74
6.2.2 Systems of ODEs 75
6.3 Vector Form of ODEs 75
6.3.1 System of ODEs in Scalar Form 75
6.3.2 Compact Vector Form 76
6.3.3 Matrix Form of a Linear System 76
6.4 General ODEs and Basic Definitions 78
6.5 Well-Posedness, Existence, and Uniqueness 80
6.5.1 Definition of a Well-Posed Initial Value Problem 80
6.5.2 Existence of Solutions 80

6.5.3 Uniqueness of Solutions . . . 81
6.5.4 Existence and Uniqueness . . . 82
6.5.5 Counterexample: Existence Without Uniqueness . . . 84

III DAE

7 Linear DAEs . . . 93
7.1 Linear Differential-Algebraic Equations 93
7.1.1 Why Differential-Algebraic Equations? . . . 93
7.1.2 Three fundamental cases . . . 95
7.1.3 Differential–Algebraic Decomposition . . . 99
7.1.4 Jordan form and the infinite eigenvalues of a pencil . . . 101
7.1.5 Worked examples: indices one, two, and three . . . 101
7.1.6 Further regular and singular pencils . . . 102

8 Index Reduction for Linear DAEs . . . 103
8.1 Separating the Algebraic Equations 103
8.1.1 DAE Formulation . . . 103
8.1.2 Change of Coordinates . . . 104
8.1.3 Rank Increase . . . 105
8.2 The Reduction Algorithm 105
8.2.1 Reduction Procedure . . . 105
8.2.2 Constructing the Subspaces . . . 107
8.2.4 Rank-based algorithms: scope and verification . . . 108
8.3 Hidden Constraints as Invariants 108
8.2.3 Reduction Algorithm . . . 109
8.3.1 Invariants of a genuine ODE . . . 110
8.3.2 Invariants are destroyed by the integrator . . . 111
8.4 Examples 111
Example 1: two reductions to an ODE . . . 112
Example 2: reduction with a row exchange . . . 114
Example 3: a singular pencil . . . 117
8.4.1 Worked reduction: a regular index-two pencil . . . 120
8.4.2 Worked reduction: a singular pencil and a missing variable . . . 120
8.4.3 Exercises with solutions . . . 121

9 The Differential Index . . . 123
9.1 DAE Index 123
9.1.1 Differential index . . . 123
9.1.2 Worked exercises using the definition . . . 125
9.1.3 Why direct differentiation becomes difficult . . . 128
9.1.4 Other notions of index . . . 128

10 Semi-Explicit Form and Its Classification . . . 129
10.1 From an Implicit DAE to Semi-Explicit Form 129
10.1.1 Semi-explicit form . . . 129
10.1.2 From an implicit DAE to semi-explicit form . . . 130
10.1.3 The index-one Jacobian condition . . . 130
10.1.4 Example: an index-one system . . . 130
10.2 Classification of Semi-Explicit DAEs 131
10.2.1 Index one . . . 131
10.2.2 Index two . . . 132
10.2.3 Index three . . . 132
10.2.4 Hessenberg form . . . 133
10.3 Can a Runge–Kutta Method Be Applied Directly? 134
10.3.1 Implicit form of a Runge–Kutta method . . . 134
10.3.2 Why this is not enough . . . 134
10.4 Examples 135
An index-three nonlinear system . . . 135
11 General Index Reduction . . . 137
11.1 The General Form $E(x,t)\dot{x} = G(x,t)$ 137
11.2 Separating the Algebraic Part with a Generalized LU 138
11.2.1 LU with partial and full pivoting . . . 138
11.2.2 The form actually used: $PA = LM$ . . . 138
11.2.3 The separator $L^{-1}P$ . . . 139
11.3 Nonlinear Index Reduction 139
11.3.1 General DAE . . . 139
11.3.2 Semi-Explicit Form . . . 140
11.3.3 Matrix Form . . . 140
11.3.4 Reduction Algorithm . . . 140
11.4 Alternative Constructions and Which to Use 142
11.4.1 Kernel-Based Reduction . . . 142
11.4.2 A kernel-based separator . . . 146
11.4.3 Which construction to use . . . 146
11.5 Examples 147
A high-index chain . . . 147
12 Dummy Variables . . . 155
12.1 Dummy-Variable Reduction 155
12.1.1 Overview . . . 155
12.1.2 Requirements . . . 155
12.1.3 Preprocessing . . . 156
12.1.4 Reduction Algorithm . . . 157

12.2 Examples **161**
Coupled pendulums . . . 161

13 DAEs from Lagrangian Mechanics . . . 169

13.1 Lagrangian DAEs **169**
13.1.1 Foundations . . . 169
13.1.2 Euler–Lagrange equations . . . 170
13.1.3 Equations of Motion . . . 171
13.1.4 Constrained Mechanics . . . 173
13.2 Numerical Integration **175**
13.2.1 ODE Reduction . . . 175
13.2.2 Constraint Drift . . . 177

IV Calculus of Variations

14 Variational Foundations . . . 185

14.1 The Brachistochrone **185**
14.2 Admissible Function Spaces **186**
14.2.1 Basic Function Spaces . . . 187
14.2.2 Lebesgue Spaces . . . 187
14.3 Function-Space Geometry **188**
14.3.1 Inner Product . . . 188
14.3.2 Norm . . . 189
14.4 First Variation **190**
14.4.1 Equivalent Forms . . . 190
14.4.2 Necessary Condition . . . 190
14.4.3 Second Variation . . . 190
14.5 Integral Differentiation **191**
14.5.1 Limit Interchange . . . 192
14.5.2 Counterexample . . . 192
14.5.3 A Sufficient Condition . . . 194
14.6 Local Minima and Existence **194**
14.7 Sobolev Spaces **195**

15 Fundamental Lemma . . . 197

15.1 Statement **197**
15.2 Sign Persistence **198**
15.3 Proof Strategy: A Triangular Test Function **198**
15.3.1 Why This Is Not Yet a Proof . . . 199

15.4 Smooth Bump Function 199
15.5 Rigorous Proof 200
15.6 Interpretation 201

16 Euler–Lagrange Equations 203
16.1 Pendulum Example 203
16.2 General Formulation 206
16.3 First Integrals and Vectors 207
16.4 Brachistochrone 209
16.5 Second Variation and Legendre Condition 210
16.5.1 Jacobi Condition 210
16.6 Weierstrass Necessary Condition 211

17 Higher-Order Functionals 213
17.1 Euler–Poisson Equation 213
17.2 Natural Boundary Conditions 215
17.3 Corner Conditions 218
17.4 Endpoint Constraints 218
17.4.1 Discrete Derivation 218
17.5 Boundary Multipliers 220
17.5.1 Fixed-End Verification 222

18 Isoperimetric Constraints 227
18.1 Feasible Variations 227
18.2 Finite Differences 229
18.2.1 Discretization Setup 229
18.2.2 Discretized Problem Formulation 230
18.2.3 Discrete Lagrangian 230
18.2.4 First-Order Conditions 230
18.3 Lagrange-Multiplier Method 232
18.3.1 Worked Example 232
18.3.2 First Variation 232
18.4 General Formulation 232
18.4.1 Problem 233
18.4.2 Augmented Functional 233
18.4.3 Stationarity 233
18.4.4 Multiple Constraints 233
18.5 Normal and Abnormal Multiplier Rules 234

19 Path Constraints 235
19.1 Discrete Derivation 235
19.2 Continuous Multipliers 238
19.2.1 Problem 238
19.2.2 Augmented Functional 238
19.2.3 First Variation 239
19.2.4 Necessary Conditions 239
19.3 Inequality Path Constraints 239
19.4 Regularity of Path-Constraint Multipliers 239

20 Summary 241
20.1 Admissible Spaces 241
20.1.1 Endpoint Costs 242
20.1.2 Endpoint Constraints 242
20.1.3 Integral Constraints 243
20.1.4 Higher-Order Reduction 243
20.2 General Problem 243
20.3 Path Constraints 244
20.4 Stationarity and Minima 244
20.5 Classification and Existence 244
20.6 Constraint Qualifications 244

V Optimal Control

21 Introduction 251
21.1 Isoperimetric problem 251
21.2 Discretized solution 251
21.3 Continuous interpretation 252
21.4 Observation 253
21.5 General formulation from Euler-Lagrange 253
21.5.1 Ordinary differential equation constraint 254

22 Optimal Control Problems 257
22.1 Bolza form and first-order necessary conditions 257
22.1.1 Integral constraints 257
22.1.2 Augmented functional and transversality 257
22.2 Pontryagin minimum principle 260
22.2.1 Examples of the Pontryagin minimum principle 260

22.3 What the Necessary Conditions Do Not Prove **264**
22.4 Linear Controls and Singular Arcs **265**
22.4.1 Singular controls 267
22.4.2 Test for singular controls 267

23 Transformations of Optimal Control Problems 271
23.1 Reference Problem and Sign Convention **271**
23.2 Lagrange to Mayer Transformation **272**
23.3 Mayer to Lagrange Transformation **272**
23.4 Constant Parameters as States **273**
23.5 Integral Constraints **273**
23.6 Fixed-Interval Time Transformation **274**
23.7 Free Terminal Time **274**
23.8 Free Initial and Final Times **275**
23.9 Path and Mixed Constraints **276**
23.10 Equivalence and Numerical Conditioning **276**

24 Summary of Optimal Control Conditions 277
24.1 Canonical Bolza problem **277**
24.2 Integral-constraint augmentation **277**
24.3 Hamiltonian and endpoint Lagrangian **277**
24.4 Necessary conditions **278**
24.5 Free terminal time **278**
24.6 Singular controls **278**
24.7 Numerical solution viewpoint **279**

Appendices

A Gaussian Elimination and Echelon Forms 281
A.1 Elimination Process **281**
A.1.1 Step 1: Choose a nonzero pivot 283
A.1.2 Step 2: Eliminate the entries below and to the right of the pivot 283
A.1.3 Step 3: Repeat on the remaining submatrix 284
A.1.4 Step 4: Normalize the pivots 284
A.2 Echelon Form **286**
A.2.1 Elementary Row Operations 286
A.2.2 Echelon-Form Example 287
A.2.3 Singular-Matrix Example 288
A.2.4 Reduction Algorithm 289

A.3 Equivalence transformations for a linear DAE 291
A.4 Numerical rank, pivoting, and conditioning 291

Contents

These notes develop the mathematical and numerical tools used to model, analyse, and optimize dynamical systems. The mathematical material is organized in five parts: constrained optimization and Karush–Kuhn–Tucker conditions (KKT), ordinary differential equations (ODE), differential-algebraic equations (DAE), functional minimization, and optimal control problems (OCP). A sixth part collects the reproducible Python laboratories.

Exercises, worked examples and case studies are collected in a companion volume, Exercises and worked examples. What remains here are only the examples that carry the exposition forward. The algorithms, their order and stability analysis and the Python laboratories are in a third volume, Numerical Methods.

The presentation moves from differential models and their numerical integration to algebraic constraints and variational methods, providing the foundations required for the formulation and solution of optimal control problems.

- **Constrained Minimization**
  - Mathematical Preliminaries
  - Lagrange Multipliers
  - Sylvester's Criterion
  - Inequality Constraints and KKT Conditions
- **Ordinary Differential Equations**
  - Introduction to ODE
  - ODE Theory and Properties
- **Differential-Algebraic Equations**
  - Linear DAEs
  - Index Reduction for Linear DAEs
  - The Differential Index
  - Semi-Explicit Form and Its Classification
  - General Index Reduction

  - Dummy Variables
  - DAEs from Lagrangian Mechanics
- **Calculus of Variations**
  - Variational Foundations
  - Fundamental Lemma
  - Euler–Lagrange Equations
  - Higher-Order Functionals
  - Isoperimetric Constraints
  - Path Constraints
  - Summary
- **Optimal Control**
  - Introduction
  - Optimal Control Problems
  - Transformations of Optimal Control Problems
  - Summary of Optimal Control Conditions
- **References**

Acknowledgement



Figure 1: License: CC BY 4.0

This project is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0). Some pages may contain materials that are subject to copyright, in which case you will see the copyright notice.

I Constrained Minimization

1 Mathematical Preliminaries 7

1.1 Taylor expansions and regularity
1.2 Quantitative first-order remainder for a Lipschitz gradient
1.3 Cubic remainder for a Lipschitz Hessian
1.4 Global and local minima
1.5 Conditions for an unconstrained local minimum
1.6 Examples: classification of stationary points

2 Lagrange Multipliers 17

2.1 Summary of constrained minimization with proofs

3 Sylvester's Criterion 27

3.1 Why use Sylvester's criterion
3.2 Positive semidefinite matrices
3.3 Summary

4 Inequality Constraints and KKT Conditions 35

4.1 Inequality Constraints
4.2 KKT Conditions
4.3 Notes and references
4.4 Chapter summary

This part develops **constrained minimization**: how to recognize, derive and verify a minimum of a function subject to equality and inequality constraints. The thread runs from the method of Lagrange multipliers, which handles equalities, to the **Karush-Kuhn-Tucker (KKT) conditions**, which extend it to inequalities, and closes with the numerical methods that solve the resulting systems.

Prerequisites, objectives, and expected outcomes

Prerequisites: multivariable calculus, Taylor expansion, and elementary linear algebra.
Objectives: formulate first- and second-order optimality conditions, interpret constraint qualifications, and connect KKT systems with numerical algorithms.
Expected outcomes: the reader can derive and verify KKT conditions, classify candidate points, diagnose degeneracy, and select an unconstrained or constrained numerical method.

Contents

- Mathematical Preliminaries
- Lagrange Multipliers
- Sylvester's Criterion
- Inequality Constraints and KKT Conditions
- KKT Worked Examples (companion volume)
- Optimization Algorithms
  - Unconstrained Optimization
  - Constrained Optimization

1. Mathematical Preliminaries

This section recalls the regularity assumptions and the first- and second-order Taylor expansions used later in constrained optimization.

Throughout these notes, the gradient of a scalar-valued function is a **row vector**:

$$\nabla f(\boldsymbol{x}) = \begin{bmatrix} \partial_{x_1} f(\boldsymbol{x}) & \cdots & \partial_{x_n} f(\boldsymbol{x}) \end{bmatrix} \in \mathbb{R}^{1\times n}.$$

Vectors such as $\boldsymbol{s}, \boldsymbol{d} \in \mathbb{R}^n$ are column vectors. Consequently,

$$\nabla f(\boldsymbol{x})\, \boldsymbol{s} \in \mathbb{R},$$

and no transpose of the gradient is needed. The Hessian $\nabla^2 f(\boldsymbol{x}) \in \mathbb{R}^{n\times n}$ has the usual entries $[\nabla^2 f]_{ij} = \partial^2 f/(\partial x_i \partial x_j)$.

1.1 Taylor expansions and regularity

Definition 1.1.1 — First-order expansion: differentiability is enough. A function $f : \mathbb{R}^n \to \mathbb{R}$ is **Fréchet differentiable** at $\boldsymbol{x}$ if there exists a row vector $\nabla f(\boldsymbol{x}) \in \mathbb{R}^{1\times n}$ such that

$$f(\boldsymbol{x}+\boldsymbol{s}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\, \boldsymbol{s} + r_1(\boldsymbol{s}),$$

where

$$\lim_{\boldsymbol{s}\to \boldsymbol{0}} \frac{|r_1(\boldsymbol{s})|}{\|\boldsymbol{s}\|} = 0.$$

Thus the first-order Taylor expansion is

$$\boxed{f(\boldsymbol{x}+\boldsymbol{s}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\boldsymbol{s} + o(\|\boldsymbol{s}\|).} \tag{1.1}$$

Thus the first-order expansion with a little-o remainder requires only differentiability at the expansion point. The stronger assumption $f \in C^1$ in a neighborhood of $\boldsymbol{x}$ is sufficient, but not necessary.

1.1.1 Restriction to a fixed direction

Let $\boldsymbol{s} = t\boldsymbol{d}$, where $\boldsymbol{d}$ is fixed. From Equation 1.1,

$$f(\boldsymbol{x} + t\boldsymbol{d}) = f(\boldsymbol{x}) + t\,\nabla f(\boldsymbol{x})\boldsymbol{d} + r_1(t\boldsymbol{d}).$$

Moreover,

$$\frac{|r_1(t\boldsymbol{d})|}{|t|} = \frac{|r_1(t\boldsymbol{d})|}{\|t\boldsymbol{d}\|}\,\|\boldsymbol{d}\| \longrightarrow 0, \qquad t \to 0,$$

and therefore

$$\boxed{f(\boldsymbol{x} + t\boldsymbol{d}) = f(\boldsymbol{x}) + t\,\nabla f(\boldsymbol{x})\boldsymbol{d} + o(|t|).}$$

If one considers only $t \downarrow 0$, the remainder is usually written simply as $o(t)$.

1.2 Quantitative first-order remainder for a Lipschitz gradient

Assume that f is differentiable on an open convex neighborhood U of $\boldsymbol{x}$ and that its gradient is Lipschitz continuous there: for some $L > 0$,

$$\|\nabla f(\boldsymbol{u}) - \nabla f(\boldsymbol{v})\| \le L\|\boldsymbol{u} - \boldsymbol{v}\|, \qquad \boldsymbol{u}, \boldsymbol{v} \in U. \tag{1.2}$$

For every $\boldsymbol{s}$ such that the segment $\{\boldsymbol{x} + t\boldsymbol{s} \,:\, 0 \le t \le 1\}$ is contained in U,

$$\boxed{|f(\boldsymbol{x} + \boldsymbol{s}) - f(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\boldsymbol{s}| \le \frac{L}{2}\|\boldsymbol{s}\|^2.} \tag{1.3}$$

Consequently,

$$f(\boldsymbol{x} + \boldsymbol{s}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\boldsymbol{s} + O(\|\boldsymbol{s}\|^2).$$

Proof

Define the scalar function

$$\phi(t) := f(\boldsymbol{x} + t\boldsymbol{s}), \qquad 0 \le t \le 1.$$

Since a Lipschitz gradient is continuous, $\phi \in C^1([0,1])$, and the chain rule gives

$$\phi'(t) = \nabla f(\boldsymbol{x} + t\boldsymbol{s})\boldsymbol{s}.$$

By the fundamental theorem of calculus,

$$\begin{aligned} f(\boldsymbol{x} + \boldsymbol{s}) - f(\boldsymbol{x}) &= \phi(1) - \phi(0) \\ &= \int_0^1 \nabla f(\boldsymbol{x} + t\boldsymbol{s})\boldsymbol{s}\, dt. \end{aligned}$$

Subtracting $\nabla f(\boldsymbol{x})\boldsymbol{s}$ yields

$$R_1(\boldsymbol{s}) = \int_0^1 \big[\nabla f(\boldsymbol{x} + t\boldsymbol{s}) - \nabla f(\boldsymbol{x})\big]\boldsymbol{s}\, dt.$$

Using Cauchy–Schwarz (equivalently, the induced Euclidean operator norm) and Equation 1.2,

$$\begin{aligned} |R_1(\boldsymbol{s})| &\le \int_0^1 \|\nabla f(\boldsymbol{x} + t\boldsymbol{s}) - \nabla f(\boldsymbol{x})\|\,\|\boldsymbol{s}\|\, dt \\ &\le \int_0^1 Lt\|\boldsymbol{s}\|^2\, dt = \frac{L}{2}\|\boldsymbol{s}\|^2. \end{aligned}$$

This proves Equation 1.3.

Definition 1.2.1 — Second-order expansion under C^2 regularity. Assume that $f \in C^2(U)$ on an open neighborhood U of $\boldsymbol{x}$. Then

$$\boxed{f(\boldsymbol{x}+\boldsymbol{s}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{s} + o(\|\boldsymbol{s}\|^2).} \tag{1.4}$$

In particular, for $\boldsymbol{s} = t\boldsymbol{d}$,

$$f(\boldsymbol{x}+t\boldsymbol{d}) = f(\boldsymbol{x}) + t\nabla f(\boldsymbol{x})\boldsymbol{d} + \frac{t^2}{2}\boldsymbol{d}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{d} + o(t^2).$$

Proof

Again set

$$\phi(t) = f(\boldsymbol{x}+t\boldsymbol{s}), \qquad 0 \le t \le 1.$$

For $\boldsymbol{s}$ sufficiently small the entire segment is contained in U. Because $f \in C^2(U)$,

$$\phi'(t) = \nabla f(\boldsymbol{x}+t\boldsymbol{s})\boldsymbol{s}, \qquad \phi''(t) = \boldsymbol{s}^T\nabla^2 f(\boldsymbol{x}+t\boldsymbol{s})\boldsymbol{s}.$$

Applying the fundamental theorem of calculus twice gives the integral form of Taylor's formula,

$$\phi(1) = \phi(0) + \phi'(0) + \int_0^1 (1-t)\phi''(t)\,dt. \tag{1.5}$$

Hence

$$f(\boldsymbol{x}+\boldsymbol{s}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\boldsymbol{s} + \int_0^1 (1-t)\boldsymbol{s}^T\nabla^2 f(\boldsymbol{x}+t\boldsymbol{s})\boldsymbol{s}\,dt.$$

Add and subtract the Hessian at $\boldsymbol{x}$. Since $\int_0^1 (1-t)dt = 1/2$,

$$\begin{aligned} f(\boldsymbol{x}+\boldsymbol{s}) &= f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{s} + R_2(\boldsymbol{s}), \\ R_2(\boldsymbol{s}) &= \int_0^1 (1-t)\boldsymbol{s}^T\left[\nabla^2 f(\boldsymbol{x}+t\boldsymbol{s}) - \nabla^2 f(\boldsymbol{x})\right]\boldsymbol{s}\,dt. \end{aligned} \tag{1.6}$$

Therefore

$$\frac{|R_2(\boldsymbol{s})|}{\|\boldsymbol{s}\|^2} \le \int_0^1 (1-t)\left\|\nabla^2 f(\boldsymbol{x}+t\boldsymbol{s}) - \nabla^2 f(\boldsymbol{x})\right\|_2 dt.$$

Continuity of $\nabla^2 f$ at $\boldsymbol{x}$ implies that the right-hand side tends to zero as $\boldsymbol{s} \to \boldsymbol{0}$. Thus $R_2(\boldsymbol{s}) = o(\|\boldsymbol{s}\|^2)$, proving Equation 1.4.

The proof shows that **Lipschitz continuity of the Hessian is not required** for the little-o second-order remainder; continuity of the Hessian at the expansion point is sufficient.

1.3 Cubic remainder for a Lipschitz Hessian

Assume in addition that the Hessian is Lipschitz continuous on a neighborhood of $\boldsymbol{x}$: there exists $L_H > 0$ such that

$$\|\nabla^2 f(\boldsymbol{u}) - \nabla^2 f(\boldsymbol{v})\|_2 \le L_H\|\boldsymbol{u}-\boldsymbol{v}\|.$$

Then the remainder in Equation 1.4 satisfies the sharper bound

$$\boxed{|R_2(\boldsymbol{s})| \le \frac{L_H}{6}\|\boldsymbol{s}\|^3.} \tag{1.7}$$

Consequently,

$$f(\boldsymbol{x}+\boldsymbol{s}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{s} + O(\|\boldsymbol{s}\|^3).$$

Proof

Starting from Equation 1.6,

$$\begin{aligned}|R_2(\boldsymbol{s})| &\le \int_0^1 (1-t)\|\boldsymbol{s}\|^2\|\nabla^2 f(\boldsymbol{x}+t\boldsymbol{s}) - \nabla^2 f(\boldsymbol{x})\|_2\, dt \\ &\le L_H\|\boldsymbol{s}\|^3 \int_0^1 t(1-t)\, dt \\ &= \frac{L_H}{6}\|\boldsymbol{s}\|^3.\end{aligned}$$

Regularity assumptions: what is actually needed?

The implications used in these notes are

$$\begin{aligned}\text{Fréchet differentiability at } \boldsymbol{x} &\implies R_1(\boldsymbol{s}) = o(\|\boldsymbol{s}\|), \\ \nabla f \text{ locally Lipschitz} &\implies |R_1(\boldsymbol{s})| \le \tfrac{L}{2}\|\boldsymbol{s}\|^2, \\ f \in C^2 \text{ near } \boldsymbol{x} &\implies R_2(\boldsymbol{s}) = o(\|\boldsymbol{s}\|^2), \\ \nabla^2 f \text{ locally Lipschitz} &\implies |R_2(\boldsymbol{s})| \le \tfrac{L_H}{6}\|\boldsymbol{s}\|^3.\end{aligned}$$

Thus Lipschitz assumptions are needed only for the displayed **quantitative bounds**, not for the corresponding little-o Taylor expansions.

1.4 Global and local minima

For $\boldsymbol{a} \in \mathbb{R}^n$ and $\delta > 0$, the **open Euclidean ball** with center $\boldsymbol{a}$ and radius δ is

$$B(\boldsymbol{a}, \delta) := \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{a}\|_2 < \delta\}, \qquad \|\boldsymbol{v}\|_2 = \sqrt{\sum_{i=1}^n v_i^2}.$$

The corresponding closed ball is $\overline{B}(\boldsymbol{a}, \delta) = \{\boldsymbol{x} : \|\boldsymbol{x} - \boldsymbol{a}\|_2 \le \delta\}$. For a function defined on $D \subseteq \mathbb{R}^n$, the definitions below use $B(\boldsymbol{a}, \delta) \cap D$; the unconstrained first-order conditions require an interior point of D.

A point $\boldsymbol{x}^\star \in \mathbb{R}^n$ is a **global minimum** of $f : \mathbb{R}^n \to \mathbb{R}$ if

$$f(\boldsymbol{x}^\star) \le f(\boldsymbol{x}) \qquad \forall \boldsymbol{x} \in \mathbb{R}^n.$$

A point $\boldsymbol{x}^\star$ is a **local minimum** if there exists $\delta > 0$ such that

$$f(\boldsymbol{x}^\star) \le f(\boldsymbol{x}) \qquad \forall \boldsymbol{x} \in B(\boldsymbol{x}^\star, \delta).$$

It is a **strict local minimum** if, for some $\delta > 0$,

$$f(\boldsymbol{x}^\star) < f(\boldsymbol{x}) \qquad \forall \boldsymbol{x} \in B(\boldsymbol{x}^\star, \delta) \setminus \{\boldsymbol{x}^\star\}.$$

1.5 Conditions for an unconstrained local minimum

1.5.1 First-order necessary condition

If f is differentiable at an interior local minimizer $\boldsymbol{x}^\star$, then

$$\boxed{\nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}.}$$

Proof

Because $\boldsymbol{x}^\star$ is an interior point, for every fixed direction $\boldsymbol{d} \in \mathbb{R}^n$ there exists $\varepsilon > 0$ such that $\boldsymbol{x}^\star + t\boldsymbol{d}$ belongs to the domain for all $|t| < \varepsilon$. Since $\boldsymbol{x}^\star$ is a local minimum,

$$f(\boldsymbol{x}^\star + t\boldsymbol{d}) - f(\boldsymbol{x}^\star) \geq 0$$

for all sufficiently small positive and negative t. Differentiability at $\boldsymbol{x}^\star$ gives

$$f(\boldsymbol{x}^\star + t\boldsymbol{d}) - f(\boldsymbol{x}^\star) = t\nabla f(\boldsymbol{x}^\star)\boldsymbol{d} + o(|t|).$$

For $t > 0$, division by t and passage to the limit $t \downarrow 0$ yield

$$\nabla f(\boldsymbol{x}^\star)\boldsymbol{d} \geq 0.$$

For $t < 0$, division by the negative number t reverses the inequality; letting $t \uparrow 0$ gives

$$\nabla f(\boldsymbol{x}^\star)\boldsymbol{d} \leq 0.$$

Hence

$$\nabla f(\boldsymbol{x}^\star)\boldsymbol{d} = 0 \qquad \forall \boldsymbol{d} \in \mathbb{R}^n.$$

The only row vector whose product with every column vector $\boldsymbol{d}$ is zero is the zero row vector. Therefore

$$\nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}.$$

This condition is necessary but not sufficient: a stationary point may be a minimum, a maximum, or a saddle point.

1.5.2 Symmetric matrices and quadratic forms

Let $\boldsymbol{A} = \boldsymbol{A}^T \in \mathbb{R}^{n\times n}$. Its quadratic form is $Q_{\boldsymbol{A}}(\boldsymbol{d}) = \boldsymbol{d}^T\boldsymbol{A}\boldsymbol{d}$. We say that $\boldsymbol{A}$ is

- **positive definite** ($\boldsymbol{A} \succ 0$) if $\boldsymbol{d}^T\boldsymbol{A}\boldsymbol{d} > 0$ for every $\boldsymbol{d} \neq \boldsymbol{0}$;
- **positive semidefinite** ($\boldsymbol{A} \succeq 0$) if $\boldsymbol{d}^T\boldsymbol{A}\boldsymbol{d} \geq 0$ for every $\boldsymbol{d}$;
- **negative definite** ($\boldsymbol{A} \prec 0$) if $\boldsymbol{d}^T\boldsymbol{A}\boldsymbol{d} < 0$ for every $\boldsymbol{d} \neq \boldsymbol{0}$;
- **negative semidefinite** ($\boldsymbol{A} \preceq 0$) if $\boldsymbol{d}^T\boldsymbol{A}\boldsymbol{d} \leq 0$ for every $\boldsymbol{d}$;
- **indefinite** if there exist $\boldsymbol{u}, \boldsymbol{v}$ with $\boldsymbol{u}^T\boldsymbol{A}\boldsymbol{u} > 0$ and $\boldsymbol{v}^T\boldsymbol{A}\boldsymbol{v} < 0$.

For symmetric matrices, positive (semi)definiteness is equivalent to all eigenvalues being strictly positive (nonnegative), respectively.

1.5.3 Rayleigh quotient and eigenvalue bounds

For a real symmetric matrix $\boldsymbol{A}$, define its **Rayleigh quotient** by

$$\mathcal{R}_{\boldsymbol{A}}(\boldsymbol{d}) = \frac{\boldsymbol{d}^T\boldsymbol{A}\boldsymbol{d}}{\boldsymbol{d}^T\boldsymbol{d}}, \qquad \boldsymbol{d} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}.$$

Theorem (Rayleigh–Ritz). If the eigenvalues of $\boldsymbol{A} = \boldsymbol{A}^T$ are $\lambda_1 \leq \cdots \leq \lambda_n$, then

$$\boxed{\lambda_1 \leq \mathcal{R}_{\boldsymbol{A}}(\boldsymbol{d}) \leq \lambda_n} \quad (\boldsymbol{d} \neq \boldsymbol{0}), \qquad \min_{\boldsymbol{d}\neq\boldsymbol{0}} \mathcal{R}_{\boldsymbol{A}}(\boldsymbol{d}) = \lambda_1, \quad \max_{\boldsymbol{d}\neq\boldsymbol{0}} \mathcal{R}_{\boldsymbol{A}}(\boldsymbol{d}) = \lambda_n.$$

Proof. By the spectral theorem, $\boldsymbol{A} = U\,\mathrm{diag}(\lambda_1, \dots, \lambda_n)U^T$ with $U^TU = \boldsymbol{I}$. Setting $\boldsymbol{z} = U^T\boldsymbol{d}$ gives

$$\mathcal{R}_A(\boldsymbol{d}) = \frac{\sum_{i=1}^n \lambda_i z_i^2}{\sum_{i=1}^n z_i^2} = \sum_{i=1}^n w_i \lambda_i, \qquad w_i = \frac{z_i^2}{\|\boldsymbol{z}\|_2^2} \geq 0, \quad \sum_i w_i = 1.$$

Thus the quotient is a convex combination of the eigenvalues. The bounds are attained at eigenvectors for λ_1 and λ_n. In particular, $\lambda_1 \|\boldsymbol{d}\|_2^2 \leq \boldsymbol{d}^T \boldsymbol{A} \boldsymbol{d} \leq \lambda_n \|\boldsymbol{d}\|_2^2$.

1.5.4 Second-order necessary condition

Assume $f \in C^2$ in a neighborhood of $\boldsymbol{x}^\star$. If $\boldsymbol{x}^\star$ is a local minimum, then

$$\boxed{\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^\star) \boldsymbol{d} \geq 0 \qquad \forall \boldsymbol{d} \in \mathbb{R}^n.}$$

Equivalently, the Hessian $\nabla^2 f(\boldsymbol{x}^\star)$ is positive semidefinite.

Proof

By the first-order necessary condition,

$$\nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}.$$

Fix any $\boldsymbol{d} \in \mathbb{R}^n$. The second-order Taylor expansion along $\boldsymbol{d}$ gives

$$f(\boldsymbol{x}^\star + t\boldsymbol{d}) - f(\boldsymbol{x}^\star) = \frac{t^2}{2} \boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^\star) \boldsymbol{d} + o(t^2).$$

Since $\boldsymbol{x}^\star$ is a local minimum, the left-hand side is nonnegative for all sufficiently small t. For $t \neq 0$, divide by $t^2/2 > 0$:

$$\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^\star) \boldsymbol{d} + \frac{2o(t^2)}{t^2} \geq 0.$$

Letting $t \to 0$ gives

$$\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^\star) \boldsymbol{d} \geq 0.$$

Because $\boldsymbol{d}$ was arbitrary, the Hessian is positive semidefinite.

An equivalent contradiction argument is useful to keep in mind: if some $\boldsymbol{d}$ satisfied $\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^\star) \boldsymbol{d} < 0$, then the negative quadratic term would dominate the $o(t^2)$ remainder for all sufficiently small nonzero t, producing values strictly below $f(\boldsymbol{x}^\star)$.

1.5.5 Second-order sufficient condition

Assume $f \in C^2$ in a neighborhood of $\boldsymbol{x}^\star$, and suppose

$$\nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}.$$

If

$$\boxed{\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^\star) \boldsymbol{d} > 0 \qquad \forall \boldsymbol{d} \neq \boldsymbol{0},}$$

then $\boldsymbol{x}^\star$ is a strict local minimum. For a real symmetric Hessian, this is equivalent to all eigenvalues being strictly positive.

Proof

Let

$$\boldsymbol{H} := \nabla^2 f(\boldsymbol{x}^\star).$$

Since $f \in C^2$, the Hessian is symmetric. Positive definiteness therefore implies that its smallest eigenvalue is strictly positive. Denote it by $\lambda_{\min} > 0$. By the Rayleigh quotient,

$$\boldsymbol{s}^T \boldsymbol{H} \boldsymbol{s} \geq \lambda_{\min} \|\boldsymbol{s}\|^2 \qquad \forall \boldsymbol{s} \in \mathbb{R}^n. \tag{1.8}$$

Using stationarity and the second-order Taylor expansion,

$$f(\boldsymbol{x}^\star + \boldsymbol{s}) - f(\boldsymbol{x}^\star) = \frac{1}{2}\boldsymbol{s}^T \boldsymbol{H}\boldsymbol{s} + o(\|\boldsymbol{s}\|^2) \geq \frac{\lambda_{\min}}{2}\|\boldsymbol{s}\|^2 + o(\|\boldsymbol{s}\|^2).$$

By the definition of little-o, for $\varepsilon = \lambda_{\min}/4$ there exists $\delta > 0$ such that

$$\left|o(\|\boldsymbol{s}\|^2)\right| \leq \frac{\lambda_{\min}}{4}\|\boldsymbol{s}\|^2$$

whenever $0 < \|\boldsymbol{s}\| < \delta$. Therefore, for every such nonzero $\boldsymbol{s}$,

$$f(\boldsymbol{x}^\star + \boldsymbol{s}) - f(\boldsymbol{x}^\star) \geq \left(\frac{\lambda_{\min}}{2} - \frac{\lambda_{\min}}{4}\right)\|\boldsymbol{s}\|^2 = \frac{\lambda_{\min}}{4}\|\boldsymbol{s}\|^2 > 0.$$

Hence

$$f(\boldsymbol{x}^\star + \boldsymbol{s}) > f(\boldsymbol{x}^\star) \qquad \text{for } 0 < \|\boldsymbol{s}\| < \delta,$$

which is precisely the definition of a strict local minimum.

The proof actually establishes the stronger local **quadratic growth** estimate

$$f(\boldsymbol{x}^\star + \boldsymbol{s}) \geq f(\boldsymbol{x}^\star) + \frac{\lambda_{\min}}{4}\|\boldsymbol{s}\|^2, \qquad \|\boldsymbol{s}\| < \delta. \tag{1.9}$$

The positive-definite Hessian condition is **sufficient but not necessary** for a strict local minimum. For example,

$$f(x) = x^4$$

has a strict global minimum at $x = 0$, although $f''(0) = 0$.

Therefore there is an inconclusive case when

$$\nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}, \qquad \nabla^2 f(\boldsymbol{x}^\star) \succeq 0,$$

but the Hessian is singular. Higher-order terms or a direct analysis of the function are then required.

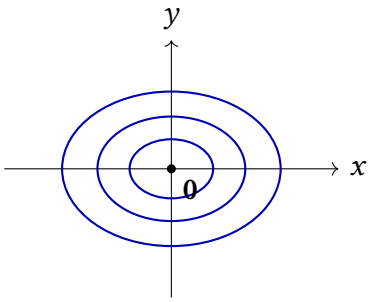
> **Note**
>
> The hierarchy to remember is
>
> $$\text{local minimum} \Longrightarrow \nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}, \quad \nabla^2 f(\boldsymbol{x}^\star) \succeq 0,$$
> $$\nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}, \quad \nabla^2 f(\boldsymbol{x}^\star) \succ 0 \Longrightarrow \text{strict local minimum}.$$
>
> Neither implication can in general be reversed.

1.6 Examples: classification of stationary points

The following four functions illustrate how the Hessian test works and why a singular Hessian can be inconclusive. Each figure is a TikZ contour sketch in the (x, y) plane; the marked point is the origin. The TikZ pictures are rendered in PDF/LaTeX output.

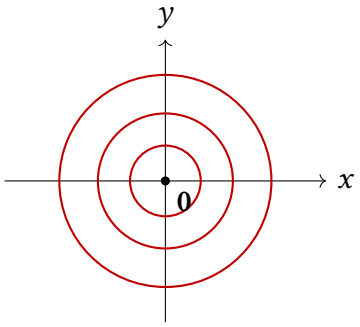
1.6.1 Example 1: strict minimum (positive definite Hessian)

Let $f(x, y) = x^2 + 2y^2$. Then $\nabla f(0, 0) = 0$ and $\nabla^2 f = \operatorname{diag}(2, 4) \succ 0$. Thus $(0, 0)$ is a strict global minimum. Its level curves are ellipses.



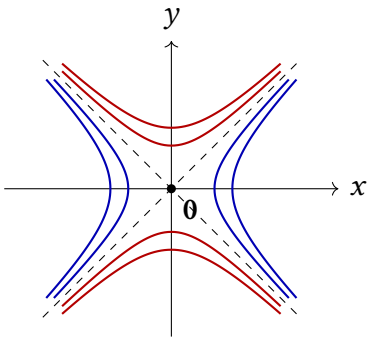
1.6.2 Example 2: strict maximum (negative definite Hessian)

Let $f(x, y) = -x^2 - y^2$. The Hessian is $-2\boldsymbol{I} \prec 0$ and $(0, 0)$ is a strict global maximum. The level curves are concentric circles, with decreasing function values away from the origin.



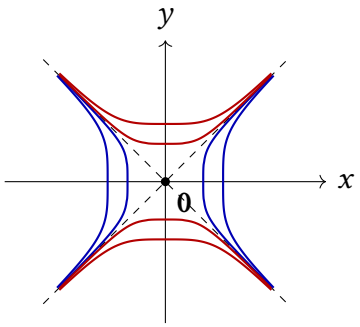
1.6.3 Example 3: saddle point (indefinite Hessian)

Let $f(x, y) = x^2 - y^2$. At the origin, $\nabla^2 f = \operatorname{diag}(2, -2)$ is indefinite. Along the x-axis $f(x, 0) > 0$ for $x \neq 0$, whereas along the y-axis $f(0, y) < 0$ for $y \neq 0$. Hence the origin is a saddle point.



1.6.4 Example 4: degenerate stationary point (inconclusive Hessian)

Let $f(x, y) = x^4 - y^4$. At $(0, 0)$ both the gradient and the Hessian vanish, so the second-order test is inconclusive. However, $f(x, 0) = x^4 > 0$ and $f(0, y) = -y^4 < 0$ for nonzero arguments: $(0, 0)$ is again a saddle point. The zero level set consists of $y = x$ and $y = -x$, but the contours differ from the quadratic case.



For comparison, $g(x, y) = x^4 + y^4$ has exactly the same zero Hessian at the origin but has a strict global minimum there. Thus a zero (or merely semidefinite) Hessian cannot by itself classify a stationary point.

2. Lagrange Multipliers

In constrained optimization, the goal is to minimize a function $f \in C^2(\mathbb{R}^n)$ (twice continuously differentiable) subject to m equality constraints defined by $h_k \in C^2(\mathbb{R}^n)$. This problem can be formulated as:

$$\begin{aligned} &\text{Minimize:} && f(\boldsymbol{x}) \\ &\text{Subject to:} && h_k(\boldsymbol{x}) = 0, \quad k = 1, \dots, m \end{aligned}$$

This means that we want to find the point $\boldsymbol{x}$ that minimizes the objective function $f(\boldsymbol{x})$ while also satisfying the m constraints given by $h_k(\boldsymbol{x}) = 0$ for each k.

The challenging analytical problem of constrained minimization can be simplified using the **Lagrange multiplier theorem**. The idea is to incorporate the constraints directly into the optimization problem through additional variables called **Lagrange multipliers**.

Let f be the function to be minimized and let $\boldsymbol{h} = (h_1, \dots, h_m)^T$ collect the constraints. Throughout these notes, gradients of scalar functions are **row vectors** and $D\boldsymbol{h}$ is the $m \times n$ Jacobian whose kth row is directly ∇h_k. Vectors in $\mathbb{R}^n$ are regarded as columns, so $D\boldsymbol{h}(\boldsymbol{x})\boldsymbol{d}$ contains the directional derivatives of all constraints along $\boldsymbol{d}$. If $\boldsymbol{x}^\star$ is a local minimum of f on the feasible set, $\boldsymbol{h}(\boldsymbol{x}^\star) = \boldsymbol{0}$, and the constraint Jacobian $D\boldsymbol{h}(\boldsymbol{x}^\star) \in \mathbb{R}^{m \times n}$ has full row rank m, then there exist m scalars $\lambda_1, \dots, \lambda_m$ such that:

$$\nabla f(\boldsymbol{x}^\star) = \sum_{k=1}^{m} \lambda_k \nabla h_k(\boldsymbol{x}^\star)$$

This equation indicates that at the optimal point $\boldsymbol{x}^\star$, the gradient of the objective function f is a linear combination of the gradients of the constraint functions h_k. The scalars λ_k are called **Lagrange multipliers**.

2.0.0.1 Lagrangian Function

The problem is transformed into a simpler form using the **Lagrangian function** $\mathcal{L}$, which incorporates both the objective function and the constraints:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) - \sum_{k=1}^{m} \lambda_k h_k(\boldsymbol{x})$$

Here, $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)^T$ is the column vector of Lagrange multipliers. The goal now is to find stationary points of the Lagrangian $\mathcal{L}$ by solving a system of equations derived from the first-order conditions.

2.0.0.2 Solving the Constrained Minimization Problem

To solve the problem, we need to find the points $\boldsymbol{x}^\star$ that satisfy the Lagrange multiplier conditions. This involves solving the following system of equations:

$$\begin{cases} \nabla_x \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \nabla f(\boldsymbol{x}) - \sum_{k=1}^{m} \lambda_k \nabla h_k(\boldsymbol{x}) = \mathbf{0} \\ \nabla_\lambda \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = -\boldsymbol{h}(\boldsymbol{x}) = \mathbf{0} \end{cases}$$

The first equation corresponds to finding points where the gradient of the Lagrangian $\mathcal{L}$ with respect to $\boldsymbol{x}$ is zero. The second equation ensures that the constraints $h_k(\boldsymbol{x}) = 0$ are satisfied.

All points $\boldsymbol{x}^\star$ that solve this system are **candidates** for being local minima, maxima, or saddle points. Since we are searching for stationary points of the Lagrangian, further analysis is required to determine whether these points are indeed minima or maxima.

2.0.0.3 Second-Order Conditions

To determine whether a candidate point $\boldsymbol{x}^\star$ is a local minimum in a constrained optimization problem, we must examine the **second-order conditions**. These conditions involve the Hessian matrix of the **Lagrangian** function. The relevant space is the tangent space to the feasible set at the candidate point, which, at a regular point, is the null space of the constraint Jacobian. Positive semidefiniteness of the Lagrangian Hessian on this space is necessary at a local minimum, whereas positive definiteness on this space is a second-order sufficient condition for a strict local minimum.

Sufficient Condition

Mathematically, the second-order condition is expressed as:

$$\boldsymbol{z}^T \nabla_x^2 \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}) \boldsymbol{z} > 0 \quad \text{for all } \boldsymbol{z} \in \ker\{\nabla \boldsymbol{h}(\boldsymbol{x}^\star)\} \setminus \{\mathbf{0}\}$$

Here:

- $\boldsymbol{z}$ is any vector in the **null space** (or kernel) of the constraint Jacobian, $\ker\{\nabla \boldsymbol{h}(\boldsymbol{x}^\star)\}$.
- The condition means that in the directions allowed by the constraints, the Hessian of the Lagrangian should be **positive**.

This condition, together with feasibility and the first-order Lagrange equations, ensures that $\boldsymbol{x}^\star$ is a **strict local minimum** of the constrained problem.

Necessary Condition

For $\boldsymbol{x}^\star$ to be a regular local minimum, a necessary second-order condition is that the Hessian $\nabla_x^2 \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda})$ is **positive semidefinite** on the tangent space, which implies:

$$\boldsymbol{z}^T \nabla_x^2 \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}) \boldsymbol{z} \geq 0 \quad \forall\, \boldsymbol{z} \in \ker\{\nabla \boldsymbol{h}(\boldsymbol{x}^\star)\}$$

In other words, the Hessian of the Lagrangian must not have negative curvature along any tangent direction. This is a **necessary condition**, but in general it is not sufficient to establish a local minimum.

▪ **Example 2.1 — constrained minimization problem.** Let's consider the problem in which we want to minimize the function $f : \mathbb{R}^2 \to \mathbb{R}$ using the constraint h defined as

$$f(x, y) = \mathrm{e}^{x^2 - y^2}, \qquad h(x, y) = x - y^2$$

We first build the Lagrangian. Since there is only one constraint, the multiplier is a scalar:

$$\mathcal{L}(x, y, \lambda) = \mathrm{e}^{x^2 - y^2} - \lambda\big(x - y^2\big)$$

The constrained stationary points are obtained by solving the following nonlinear system:

$$\begin{cases} \nabla_x \mathcal{L}(x, y, \lambda) = 2x\mathrm{e}^{x^2 - y^2} - \lambda = 0 \\ \nabla_y \mathcal{L}(x, y, \lambda) = -2y\mathrm{e}^{x^2 - y^2} + 2\lambda y = 0 \\ \nabla_\lambda \mathcal{L}(x, y, \lambda) = -x + y^2 = 0 \end{cases}$$

$$\implies \quad \big(x, y, \lambda\big) = \big(0, 0, 0\big),$$

$$\implies \quad \big(x, y, \lambda\big) = \left(\frac{1}{2}, \frac{1}{\sqrt{2}}, \mathrm{e}^{-\frac{1}{4}}\right),$$

$$\implies \quad \big(x, y, \lambda\big) = \left(\frac{1}{2}, -\frac{1}{\sqrt{2}}, \mathrm{e}^{-\frac{1}{4}}\right)$$

To classify these stationary points, we compute the Jacobian of the constraint and the Hessian of the Lagrangian with respect to $\boldsymbol{x} = (x, y)$:

$$\nabla h(x, y) = (1, -2y)$$

$$\nabla^2_{(x,y)} \mathcal{L} = \begin{bmatrix} (4x^2 + 2)\mathrm{e}^{x^2 - y^2} & -4xy\mathrm{e}^{x^2 - y^2} \\ -4xy\mathrm{e}^{x^2 - y^2} & (4y^2 - 2)\mathrm{e}^{x^2 - y^2} + 2\lambda \end{bmatrix}$$

We now examine each stationary point separately:

1. when $x = y = \lambda = 0$ we have that $\nabla h = (1, 0)$ while

$$\nabla^2_{(x,y)} \mathcal{L} = \begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix}$$

   The tangent space is the null space of $(1, 0)$, hence it consists of vectors of the form $(0, \alpha)$. Restricting the Hessian quadratic form to this space gives:

$$\begin{pmatrix} 0 & \alpha \end{pmatrix} \begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix} \begin{pmatrix} 0 \\ \alpha \end{pmatrix} = -2\alpha^2 \leq 0$$

   for all $\alpha \in \mathbb{R}$. The reduced Hessian is negative definite on the tangent space, so $(x, y) = (0, 0)$ is a strict local maximum of the constrained problem.

2. evaluating for the second point

$$x = \frac{1}{2}, \quad y = \frac{1}{\sqrt{2}}, \quad \text{and} \quad \lambda = \mathrm{e}^{-\frac{1}{4}}$$

   we obtain $\nabla h = (1, -\sqrt{2})$, whose null space consists of vectors $(\alpha\sqrt{2}, \alpha)$. Restricting the Hessian to this tangent space gives

$$\mathrm{e}^{-\frac{1}{4}} \begin{pmatrix} \alpha\sqrt{2} & \alpha \end{pmatrix} \begin{bmatrix} 3 & -\sqrt{2} \\ -\sqrt{2} & 2 \end{bmatrix} \begin{pmatrix} \alpha\sqrt{2} \\ \alpha \end{pmatrix} = 4\mathrm{e}^{-\frac{1}{4}} \alpha^2 > 0$$

   for all $\alpha \in \mathbb{R}$. This means that the point is a strict local minimum.

3. considering instead the last point

$$x = \frac{1}{2}, \quad y = -\frac{1}{\sqrt{2}} \quad \text{and} \quad \lambda = \mathrm{e}^{-\frac{1}{4}}$$

we have $\nabla h = (1, \sqrt{2})$, whose kernel consists of vectors $(\alpha\sqrt{2}, -\alpha)$. Restricting the Hessian to this tangent space gives

$$\mathrm{e}^{-\frac{1}{4}} \begin{pmatrix} \alpha\sqrt{2} & -\alpha \end{pmatrix} \begin{bmatrix} 3 & \sqrt{2} \\ \sqrt{2} & 2 \end{bmatrix} \begin{pmatrix} \alpha\sqrt{2} \\ -\alpha \end{pmatrix} = 4\mathrm{e}^{-\frac{1}{4}}\alpha^2 > 0$$

for all $\alpha \in \mathbb{R}$. Thus this point is also a strict local minimum.

We can see that the both points

$$\left(\frac{1}{2}, \frac{1}{\sqrt{2}}\right) \quad \text{and} \quad \left(\frac{1}{2}, -\frac{1}{\sqrt{2}}\right)$$

are strict local minima. They are also global minima because, on the feasible set $x = y^2$,

$$f(y^2, y) = \mathrm{e}^{y^4 - y^2} = \mathrm{e}^{(y^2 - 1/2)^2 - 1/4} \geq \mathrm{e}^{-1/4},$$

with equality exactly when $y^2 = 1/2$. Thus

$$f\left(\frac{1}{2}, \frac{1}{\sqrt{2}}\right) = f\left(\frac{1}{2}, -\frac{1}{\sqrt{2}}\right) = \mathrm{e}^{-1/4}.$$

▪

▪ **Example 2.2 — determination of the nonlinear system.** Given the problem

$$\begin{aligned} \text{minimize:} \quad & f(x, y, z) = x - y + z^2 \\ \text{subject to:} \quad & h_1(x, y, z) = x^2 + y^2 - 2 = 0 \\ & h_2(x, y, z) = x + z - 1 = 0 \end{aligned}$$

the solution can be computed by first forming the Lagrangian

$$\mathcal{L} = f - \lambda_1 h_1 - \lambda_2 h_2$$

of the problem that's

$$\mathcal{L}(x, y, z, \lambda_1, \lambda_2) = x - y + z^2 - \lambda_1\left(x^2 + y^2 - 2\right) - \lambda_2\left(x + z - 1\right)$$

The first order necessary condition related to the Lagrange multipliers states that a point, to be a minimum, must be a stationary one and so the candidate minimum points for this problem can be computed by solving the following system of nonlinear equation:

$$\begin{cases} 1 - 2\lambda_1 x - \lambda_2 = 0 & : \frac{\partial}{\partial x}\mathcal{L} \\ -1 - 2\lambda_1 y = 0 & : \frac{\partial}{\partial y}\mathcal{L} \\ 2z - \lambda_2 = 0 & : \frac{\partial}{\partial z}\mathcal{L} \\ x^2 + y^2 = 2 & : -\frac{\partial}{\partial \lambda_1}\mathcal{L} = h_1 \\ x + z = 1 & : -\frac{\partial}{\partial \lambda_2}\mathcal{L} = h_2 \end{cases}$$

▪

2.1 Summary of constrained minimization with proofs

Consider the equality-constrained optimization problem

$$\begin{aligned} &\text{minimize:} && f(\boldsymbol{x}) \\ &\text{subject to:} && h_i(\boldsymbol{x}) = 0, \quad i = 1, 2, \dots, m, \end{aligned} \tag{2.1}$$

where $f \in C^2(\mathbb{R}^n, \mathbb{R})$ and $\boldsymbol{h} = (h_1, \dots, h_m)^T \in C^2(\mathbb{R}^n, \mathbb{R}^m)$. We use the Jacobian convention

$$\boldsymbol{J}(\boldsymbol{x}) := D\boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} \nabla h_1(\boldsymbol{x}) \\ \vdots \\ \nabla h_m(\boldsymbol{x}) \end{bmatrix} \in \mathbb{R}^{m \times n},$$

and all gradients of scalar functions are row vectors. The Lagrangian is

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) - \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}) = f(\boldsymbol{x}) - \sum_{i=1}^{m} \lambda_i h_i(\boldsymbol{x}).$$

Solution Algorithm

1. **Form the Lagrangian**:
$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) - \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}).$$
2. **Solve the first-order equations**:
$$\nabla_x \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \boldsymbol{0}, \qquad \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}.$$
These equations provide candidate constrained stationary points.
3. **Check the regularity condition**: verify that
$$\operatorname{rank} \boldsymbol{J}(\boldsymbol{x}^\star) = m.$$
Thus the constraint gradients are linearly independent. For equality constraints this is the **linear independence constraint qualification (LICQ)**.
4. **Compute the tangent space**: the tangent space to the feasible set at a regular point is
$$\mathcal{T}(\boldsymbol{x}^\star) = \ker \boldsymbol{J}(\boldsymbol{x}^\star).$$
If the columns of $\boldsymbol{K}$ form a basis of this kernel, then
$$\boldsymbol{J}(\boldsymbol{x}^\star)\boldsymbol{K} = \boldsymbol{0}.$$
5. **Compute the reduced Hessian**:
$$\boldsymbol{H}_R = \boldsymbol{K}^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) \boldsymbol{K}.$$
6. **Verify the second-order conditions**:
   - **necessary condition**: $\boldsymbol{H}_R$ must be positive semidefinite at a regular local minimum;
   - **sufficient condition**: if $\boldsymbol{H}_R$ is positive definite, then $\boldsymbol{x}^\star$ is a strict local minimum.

Theorem 2.1.1 — Lagrange multipliers and second-order conditions. Let $f \in C^2(\mathbb{R}^n, \mathbb{R})$ and $\boldsymbol{h} \in C^2(\mathbb{R}^n, \mathbb{R}^m)$, with $m \leq n$. Let $\boldsymbol{x}^\star$ be feasible,

$$\boldsymbol{h}(\boldsymbol{x}^\star) = \boldsymbol{0},$$

and assume the regularity condition

$$\operatorname{rank} \boldsymbol{J}(\boldsymbol{x}^\star) = m.$$

Then the following statements hold.

1. **First-order necessary condition.**
   If $\boldsymbol{x}^\star$ is a local minimum of Equation 2.1, then there exists a unique $\boldsymbol{\lambda}^\star \in \mathbb{R}^m$ such that

$$\nabla_x \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) = \nabla f(\boldsymbol{x}^\star) - \boldsymbol{J}(\boldsymbol{x}^\star)^T \boldsymbol{\lambda}^\star = \boldsymbol{0}.$$

2. **Second-order necessary condition.**
   If $\boldsymbol{x}^\star$ is a local minimum and $\boldsymbol{\lambda}^\star$ is the multiplier above, then

$$z^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) z \geq 0 \qquad \forall\, z \in \ker \boldsymbol{J}(\boldsymbol{x}^\star).$$

3. **Second-order sufficient condition.**
   Conversely, suppose that $\boldsymbol{x}^\star$ is feasible and that there exists $\boldsymbol{\lambda}^\star$ satisfying

$$\nabla_x \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) = \boldsymbol{0}.$$

   If there exists $\gamma > 0$ such that

$$z^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) z \geq \gamma \|z\|^2 \qquad \forall\, z \in \ker \boldsymbol{J}(\boldsymbol{x}^\star),$$

   then $\boldsymbol{x}^\star$ is a **strict local minimum** of Equation 2.1.

Because the tangent space is finite dimensional, the last condition is equivalent to strict positivity

$$z^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) z > 0 \qquad \forall\, z \in \ker \boldsymbol{J}(\boldsymbol{x}^\star) \setminus \{\boldsymbol{0}\}.$$

Proof

We prove the necessary and sufficient conditions separately.

Necessary conditions

Assume that $\boldsymbol{x}^\star$ is a local minimum. Choose $r > 0$ sufficiently small that

$$f(\boldsymbol{x}^\star) \leq f(\boldsymbol{x}) \qquad \text{for every feasible } \boldsymbol{x} \in \mathcal{B}, \tag{2.2}$$

where

$$\mathcal{B} = \{\boldsymbol{x} \,:\, \|\boldsymbol{x} - \boldsymbol{x}^\star\| \leq r\}. \tag{2.3}$$

Fix an arbitrary $\alpha > 0$ and, for each positive integer k, define

$$f_k(\boldsymbol{x}) = f(\boldsymbol{x}) + k\|\boldsymbol{h}(\boldsymbol{x})\|^2 + \alpha\|\boldsymbol{x} - \boldsymbol{x}^\star\|^2. \tag{2.4}$$

Since $\mathcal{B}$ is compact and f_k is continuous, there exists at least one minimizer

$$x_k \in \operatorname*{argmin}_{x \in \mathcal{B}} f_k(x).$$

By compactness, every sequence $\{x_k\}$ has a convergent subsequence. Let

$$x_{k_j} \longrightarrow \bar{x} \in \mathcal{B}.$$

Step 1: the limit is feasible

Since $x^\star$ is feasible,

$$f_{k_j}(x_{k_j}) \leq f_{k_j}(x^\star) = f(x^\star).$$

Hence

$$k_j \|h(x_{k_j})\|^2 + \alpha \|x_{k_j} - x^\star\|^2 \leq f(x^\star) - f(x_{k_j}).$$

Because f is continuous on the compact set $\mathcal{B}$, it is bounded below there. Therefore, for some constant C independent of j,

$$k_j \|h(x_{k_j})\|^2 + \alpha \|x_{k_j} - x^\star\|^2 \leq C. \tag{2.5}$$

It follows that

$$\|h(x_{k_j})\| \longrightarrow 0.$$

By continuity of h,

$$h(\bar{x}) = 0.$$

Step 2: the limit is $x^\star$

From Equation 2.5,

$$\alpha \|x_{k_j} - x^\star\|^2 \leq f(x^\star) - f(x_{k_j}).$$

Passing to the limit gives

$$\alpha \|\bar{x} - x^\star\|^2 \leq f(x^\star) - f(\bar{x}).$$

The point $\bar{x}$ is feasible and belongs to $\mathcal{B}$, so local optimality of $x^\star$ gives

$$f(x^\star) \leq f(\bar{x}).$$

Consequently,

$$\alpha \|\bar{x} - x^\star\|^2 \leq 0,$$

and therefore

$$\bar{x} = x^\star.$$

The same argument applies to every convergent subsequence. Hence, in fact,

$$x_k \longrightarrow x^\star.$$

Since $x^\star$ is the center of the ball $\mathcal{B}$, x_k lies in the interior of $\mathcal{B}$ for all sufficiently large k. Thus, from now on, the ordinary unconstrained first- and second-order necessary conditions for f_k may be used at x_k.

Step 3: construction of the Lagrange multiplier

For all sufficiently large k,

$$\nabla f_k(\boldsymbol{x}_k) = \mathbf{0}.$$

Using row gradients and $\boldsymbol{J}(\boldsymbol{x}) = D\boldsymbol{h}(\boldsymbol{x})$, this is

$$\nabla f(\boldsymbol{x}_k) + 2k\boldsymbol{h}(\boldsymbol{x}_k)^T \boldsymbol{J}(\boldsymbol{x}_k) + 2\alpha(\boldsymbol{x}_k - \boldsymbol{x}^\star)^T = \mathbf{0}^T. \tag{2.6}$$

Right-multiplication by $\boldsymbol{J}(\boldsymbol{x}_k)^T$ gives

$$2k\boldsymbol{h}(\boldsymbol{x}_k)^T \boldsymbol{J}(\boldsymbol{x}_k)\boldsymbol{J}(\boldsymbol{x}_k)^T = -\left[\nabla f(\boldsymbol{x}_k) + 2\alpha(\boldsymbol{x}_k - \boldsymbol{x}^\star)^T\right] \boldsymbol{J}(\boldsymbol{x}_k)^T.$$

Since $\boldsymbol{J}(\boldsymbol{x}^\star)$ has full row rank, continuity implies that $\boldsymbol{J}(\boldsymbol{x}_k)$ also has full row rank for all sufficiently large k. Hence $\boldsymbol{J}(\boldsymbol{x}_k)\boldsymbol{J}(\boldsymbol{x}_k)^T$ is invertible and

$$2k\boldsymbol{h}(\boldsymbol{x}_k)^T = -\left[\nabla f(\boldsymbol{x}_k) + 2\alpha(\boldsymbol{x}_k - \boldsymbol{x}^\star)^T\right] \boldsymbol{J}(\boldsymbol{x}_k)^T \left[\boldsymbol{J}(\boldsymbol{x}_k)\boldsymbol{J}(\boldsymbol{x}_k)^T\right]^{-1}.$$

Therefore,

$$2k\boldsymbol{h}(\boldsymbol{x}_k)^T \longrightarrow -\nabla f(\boldsymbol{x}^\star)(\boldsymbol{J}^\star)^T \left[\boldsymbol{J}^\star(\boldsymbol{J}^\star)^T\right]^{-1} =: -(\boldsymbol{\lambda}^\star)^T, \tag{2.7}$$

where $\boldsymbol{J}^\star := \boldsymbol{J}(\boldsymbol{x}^\star)$. Passing to the limit in Equation 2.6 yields

$$\nabla f(\boldsymbol{x}^\star) - (\boldsymbol{\lambda}^\star)^T \boldsymbol{J}^\star = \mathbf{0}^T, \tag{2.8}$$

which is exactly

$$\nabla_x \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) = \mathbf{0}.$$

The multiplier is unique: if both $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ satisfy the stationarity condition, then

$$(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)^T \boldsymbol{J}^\star = \mathbf{0}^T.$$

Since $\boldsymbol{J}^\star$ has full row rank, its rows are linearly independent, and hence $\boldsymbol{\lambda}_1 = \boldsymbol{\lambda}_2$.

Step 4: tangent directions for the penalized problems

Let

$$\boldsymbol{z} \in \ker \boldsymbol{J}^\star.$$

Define

$$\boldsymbol{P}_k = \boldsymbol{I} - \boldsymbol{J}(\boldsymbol{x}_k)^T \left[\boldsymbol{J}(\boldsymbol{x}_k)\boldsymbol{J}(\boldsymbol{x}_k)^T\right]^{-1} \boldsymbol{J}(\boldsymbol{x}_k),$$

and

$$\boldsymbol{z}_k := \boldsymbol{P}_k \boldsymbol{z}.$$

The matrix $\boldsymbol{P}_k$ is the orthogonal projector onto $\ker \boldsymbol{J}(\boldsymbol{x}_k)$, so

$$\boldsymbol{J}(\boldsymbol{x}_k)\boldsymbol{z}_k = \mathbf{0}. \tag{2.9}$$

Moreover, by continuity,

$$\boldsymbol{P}_k \longrightarrow \boldsymbol{P}^\star := \boldsymbol{I} - (\boldsymbol{J}^\star)^T \left[\boldsymbol{J}^\star(\boldsymbol{J}^\star)^T\right]^{-1} \boldsymbol{J}^\star.$$

Since $\boldsymbol{J}^\star \boldsymbol{z} = \mathbf{0}$, $\boldsymbol{P}^\star \boldsymbol{z} = \boldsymbol{z}$. Hence

$$\boldsymbol{z}_k \longrightarrow \boldsymbol{z}.$$

Step 5: second-order necessary condition

For all sufficiently large k, $\boldsymbol{x}_k$ is an unconstrained local minimum of f_k, and thus

$$\nabla^2 f_k(\boldsymbol{x}_k) \succeq 0.$$

A direct calculation gives

$$\begin{aligned}\nabla^2 f_k(\boldsymbol{x}_k) = \nabla^2 f(\boldsymbol{x}_k) + 2k\boldsymbol{J}(\boldsymbol{x}_k)^T\boldsymbol{J}(\boldsymbol{x}_k) \\ + 2k\sum_{i=1}^{m} h_i(\boldsymbol{x}_k)\nabla^2 h_i(\boldsymbol{x}_k) + 2\alpha\boldsymbol{I}.\end{aligned} \tag{2.10}$$

Evaluate this quadratic form along $\boldsymbol{z}_k \in \ker \boldsymbol{J}(\boldsymbol{x}_k)$. The term containing $\boldsymbol{J}^T\boldsymbol{J}$ vanishes, and therefore

$$\begin{aligned}0 \le \boldsymbol{z}_k^T\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{z}_k \\ + \sum_{i=1}^{m}\left(2kh_i(\boldsymbol{x}_k)\right)\boldsymbol{z}_k^T\nabla^2 h_i(\boldsymbol{x}_k)\boldsymbol{z}_k + 2\alpha\|\boldsymbol{z}_k\|^2.\end{aligned}$$

Using Equation 2.7, $\boldsymbol{x}_k \to \boldsymbol{x}^\star$, and $\boldsymbol{z}_k \to \boldsymbol{z}$, we obtain

$$0 \le \boldsymbol{z}^T\nabla^2 f(\boldsymbol{x}^\star)\boldsymbol{z} - \sum_{i=1}^{m}\lambda_i^\star \boldsymbol{z}^T\nabla^2 h_i(\boldsymbol{x}^\star)\boldsymbol{z} + 2\alpha\|\boldsymbol{z}\|^2.$$

Equivalently,

$$\boldsymbol{z}^T\nabla^2_{\boldsymbol{xx}}\mathcal{L}(\boldsymbol{x}^\star,\boldsymbol{\lambda}^\star)\boldsymbol{z} \ge -2\alpha\|\boldsymbol{z}\|^2.$$

The preceding construction is valid for every $\alpha > 0$, and the limiting multiplier $\boldsymbol{\lambda}^\star$ is independent of α. Letting $\alpha \downarrow 0$ gives

$$\boxed{\boldsymbol{z}^T\nabla^2_{\boldsymbol{xx}}\mathcal{L}(\boldsymbol{x}^\star,\boldsymbol{\lambda}^\star)\boldsymbol{z} \ge 0 \qquad \forall\, \boldsymbol{z} \in \ker \boldsymbol{J}^\star.}$$

This proves the second-order necessary condition.

Sufficient condition

We now assume only that $\boldsymbol{x}^\star$ is feasible, that LICQ holds, and that there exists a multiplier $\boldsymbol{\lambda}^\star$ satisfying

$$\nabla_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}^\star,\boldsymbol{\lambda}^\star) = \boldsymbol{0}.$$

Let

$$\boldsymbol{H} := \nabla^2_{\boldsymbol{xx}}\mathcal{L}(\boldsymbol{x}^\star,\boldsymbol{\lambda}^\star),$$

and assume that, for some $\gamma > 0$,

$$\boldsymbol{z}^T\boldsymbol{H}\boldsymbol{z} \ge \gamma\|\boldsymbol{z}\|^2 \qquad \forall\, \boldsymbol{z} \in \ker \boldsymbol{J}^\star. \tag{2.11}$$

The essential point is that a feasible displacement is tangent to first order, but it need not itself belong to the tangent space.

Let $\boldsymbol{x} = \boldsymbol{x}^\star + \boldsymbol{\delta}$ be feasible and sufficiently close to $\boldsymbol{x}^\star$. Define the orthogonal projectors

$$\boldsymbol{P}_N = (\boldsymbol{J}^\star)^T\left[\boldsymbol{J}^\star(\boldsymbol{J}^\star)^T\right]^{-1}\boldsymbol{J}^\star, \qquad \boldsymbol{P}_T = \boldsymbol{I} - \boldsymbol{P}_N,$$

and decompose

$$\boldsymbol{\delta} = \boldsymbol{z} + \boldsymbol{r}, \qquad \boldsymbol{z} := \boldsymbol{P}_T\boldsymbol{\delta} \in \ker \boldsymbol{J}^\star, \qquad \boldsymbol{r} := \boldsymbol{P}_N\boldsymbol{\delta}.$$

Because both $\boldsymbol{x}^\star$ and $\boldsymbol{x}^\star + \boldsymbol{\delta}$ are feasible, Taylor expansion of $\boldsymbol{h}$ gives

$$\mathbf{0} = \boldsymbol{h}(\boldsymbol{x}^\star + \boldsymbol{\delta}) - \boldsymbol{h}(\boldsymbol{x}^\star) = \boldsymbol{J}^\star \boldsymbol{\delta} + O(\|\boldsymbol{\delta}\|^2).$$

Hence

$$\boldsymbol{J}^\star \boldsymbol{\delta} = O(\|\boldsymbol{\delta}\|^2),$$

and therefore

$$\boldsymbol{r} = (\boldsymbol{J}^\star)^T \left[\boldsymbol{J}^\star (\boldsymbol{J}^\star)^T\right]^{-1} \boldsymbol{J}^\star \boldsymbol{\delta} = O(\|\boldsymbol{\delta}\|^2). \tag{2.12}$$

Thus a feasible displacement satisfies

$$\boxed{\boldsymbol{\delta} = \boldsymbol{z} + O(\|\boldsymbol{\delta}\|^2), \qquad \boldsymbol{z} \in \ker \boldsymbol{J}^\star.}$$

In particular,

$$\|\boldsymbol{z}\|^2 = \|\boldsymbol{\delta}\|^2 + O(\|\boldsymbol{\delta}\|^3).$$

Using Equation 2.11 and Equation 2.12,

$$\begin{aligned}
\boldsymbol{\delta}^T \boldsymbol{H} \boldsymbol{\delta} &= (\boldsymbol{z} + \boldsymbol{r})^T \boldsymbol{H} (\boldsymbol{z} + \boldsymbol{r}) \\
&= \boldsymbol{z}^T \boldsymbol{H} \boldsymbol{z} + 2\boldsymbol{z}^T \boldsymbol{H} \boldsymbol{r} + \boldsymbol{r}^T \boldsymbol{H} \boldsymbol{r} \\
&\geq \gamma \|\boldsymbol{z}\|^2 + O(\|\boldsymbol{\delta}\|^3) \\
&= \gamma \|\boldsymbol{\delta}\|^2 + O(\|\boldsymbol{\delta}\|^3).
\end{aligned}$$

Consequently, for all sufficiently small feasible $\boldsymbol{\delta}$,

$$\boldsymbol{\delta}^T \boldsymbol{H} \boldsymbol{\delta} \geq \frac{\gamma}{2} \|\boldsymbol{\delta}\|^2. \tag{2.13}$$

Finally, because $\boldsymbol{x}^\star + \boldsymbol{\delta}$ is feasible,

$$f(\boldsymbol{x}^\star + \boldsymbol{\delta}) = \mathcal{L}(\boldsymbol{x}^\star + \boldsymbol{\delta}, \boldsymbol{\lambda}^\star),$$

and also

$$f(\boldsymbol{x}^\star) = \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star).$$

Taylor expansion of the Lagrangian gives

$$\begin{aligned}
f(\boldsymbol{x}^\star + \boldsymbol{\delta}) - f(\boldsymbol{x}^\star) &= \cancel{\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star)^T \boldsymbol{\delta}} + \frac{1}{2} \boldsymbol{\delta}^T \boldsymbol{H} \boldsymbol{\delta} + o(\|\boldsymbol{\delta}\|^2) \\
&\geq \frac{\gamma}{4} \|\boldsymbol{\delta}\|^2 + o(\|\boldsymbol{\delta}\|^2).
\end{aligned}$$

Therefore, after possibly shrinking the neighborhood once more,

$$f(\boldsymbol{x}^\star + \boldsymbol{\delta}) > f(\boldsymbol{x}^\star)$$

for every nonzero feasible $\boldsymbol{\delta}$ sufficiently small. Hence $\boldsymbol{x}^\star$ is a strict local minimum.

□

3. Sylvester's Criterion

Sylvester's criterion provides an operational test for deciding whether a real symmetric matrix is positive definite.

Theorem 3.0.1 — **Sylvester's Criterion.** Let $A \in \mathbb{R}^{n\times n}$ be symmetric, and let

$$A_k := A_{1:k,1:k}, \qquad k = 1, \dots, n,$$

be its leading principal submatrices. Then

$$\boxed{A \succ 0 \iff \det(A_k) > 0, \qquad k = 1, \dots, n.}$$

In other words, a real symmetric matrix is positive definite if and only if all its **leading principal minors** are strictly positive.

Proof

Necessity

Assume that A is symmetric positive definite. Then

$$x^\top A x > 0 \qquad \forall x \neq 0.$$

For each $k = 1, \dots, n$, take an arbitrary nonzero vector $y \in \mathbb{R}^k$ and embed it in $\mathbb{R}^n$ as

$$x = \begin{bmatrix} y \\ 0 \end{bmatrix}.$$

Then

$$y^\top A_k y = x^\top A x > 0.$$

Hence every leading principal submatrix A_k is itself symmetric positive definite. Therefore all eigenvalues of A_k are positive, and thus

$$\det(A_k) = \prod_{i=1}^{k} \lambda_i(A_k) > 0.$$

So all leading principal minors are positive.

Sufficiency

Assume now that

$$\det(\boldsymbol{A}_k) > 0, \qquad k = 1, \dots, n.$$

We prove by induction that every $\boldsymbol{A}_k$ is positive definite.
For $k = 1$,

$$\boldsymbol{A}_1 = [a_{11}], \qquad \det(\boldsymbol{A}_1) = a_{11} > 0,$$

so $\boldsymbol{A}_1 \succ 0$.
Assume that $\boldsymbol{A}_{k-1} \succ 0$. Write

$$\boldsymbol{A}_k = \begin{bmatrix} \boldsymbol{A}_{k-1} & \boldsymbol{b} \\ \boldsymbol{b}^\top & c \end{bmatrix}.$$

Since $\boldsymbol{A}_{k-1} \succ 0$, it is invertible. Define its Schur complement

$$d := c - \boldsymbol{b}^\top \boldsymbol{A}_{k-1}^{-1} \boldsymbol{b}.$$

Then

$$\boldsymbol{A}_k = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{b}^\top \boldsymbol{A}_{k-1}^{-1} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_{k-1} & \boldsymbol{0} \\ \boldsymbol{0}^\top & d \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{A}_{k-1}^{-1} \boldsymbol{b} \\ \boldsymbol{0}^\top & 1 \end{bmatrix}. \tag{3.1}$$

Because the first and third factors have determinant one,

$$\det(\boldsymbol{A}_k) = \det(\boldsymbol{A}_{k-1})\, d.$$

Both determinants on the left and in the denominator are positive, hence

$$d = \frac{\det(\boldsymbol{A}_k)}{\det(\boldsymbol{A}_{k-1})} > 0.$$

Now let

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{y} \\ z \end{bmatrix} \neq \boldsymbol{0}, \qquad \boldsymbol{w} := \boldsymbol{y} + z \boldsymbol{A}_{k-1}^{-1} \boldsymbol{b}.$$

Using Equation 3.1,

$$\boldsymbol{x}^\top \boldsymbol{A}_k \boldsymbol{x} = \boldsymbol{w}^\top \boldsymbol{A}_{k-1} \boldsymbol{w} + d z^2.$$

The transformation from $(\boldsymbol{y}, z)$ to $(\boldsymbol{w}, z)$ is invertible, so $(\boldsymbol{w}, z) \neq (\boldsymbol{0}, 0)$. Since $\boldsymbol{A}_{k-1} \succ 0$ and $d > 0$, the right-hand side is strictly positive. Therefore

$$\boldsymbol{A}_k \succ 0.$$

By induction, $\boldsymbol{A} = \boldsymbol{A}_n \succ 0$. □

Negative Definite Matrices

For completeness, applying Sylvester's criterion to $-\boldsymbol{A}$ gives the corresponding test for negative definiteness:

$$\boxed{\boldsymbol{A} \prec 0 \quad \Longleftrightarrow \quad (-1)^k \det(\boldsymbol{A}_k) > 0, \qquad k = 1, \dots, n.}$$

Equivalently, the leading principal minors alternate in sign:

$$\det(\boldsymbol{A}_1) < 0, \quad \det(\boldsymbol{A}_2) > 0, \quad \det(\boldsymbol{A}_3) < 0, \quad \dots$$

3.1 Why use Sylvester's criterion

For an equality-constrained minimization problem, let

$$\boldsymbol{A} := \nabla_{\boldsymbol{x}}^2 \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star)$$

and

$$\boldsymbol{B} := D\boldsymbol{h}(\boldsymbol{x}^\star).$$

The second-order necessary condition requires

$$\boldsymbol{z}^\top \boldsymbol{A}\boldsymbol{z} \geq 0 \qquad \forall \boldsymbol{z} \in \ker(\boldsymbol{B}).$$

The second-order sufficient condition requires the stronger inequality

$$\boldsymbol{z}^\top \boldsymbol{A}\boldsymbol{z} > 0 \qquad \forall \boldsymbol{z} \in \ker(\boldsymbol{B}) \setminus \{\boldsymbol{0}\}.$$

Let

$$p := \dim \ker(\boldsymbol{B}) = n - \operatorname{rank}(\boldsymbol{B}),$$

and let the columns of

$$\boldsymbol{K} \in \mathbb{R}^{n \times p}$$

form a basis of $\ker(\boldsymbol{B})$. Then every vector in the tangent space can be written uniquely as

$$\boldsymbol{z} = \boldsymbol{K}\boldsymbol{\alpha}, \qquad \boldsymbol{\alpha} \in \mathbb{R}^p.$$

Therefore

$$\boldsymbol{z}^\top \boldsymbol{A}\boldsymbol{z} = \boldsymbol{\alpha}^\top \boldsymbol{K}^\top \boldsymbol{A}\boldsymbol{K}\boldsymbol{\alpha}.$$

Define the **reduced Hessian**

$$\boxed{\boldsymbol{C} := \boldsymbol{K}^\top \boldsymbol{A}\boldsymbol{K} \in \mathbb{R}^{p \times p}.}$$

The definiteness of $\boldsymbol{C}$ does not depend on the particular basis $\boldsymbol{K}$ chosen for the tangent space. If $\widetilde{K} = KR$ with R nonsingular, then $\widetilde{K}^T A \widetilde{K} = R^T (K^T A K) R$, so Sylvester's law of inertia preserves the numbers of positive, negative, and zero eigenvalues. For numerical work, an **orthonormal** basis K (for example from a QR or SVD factorization) is usually preferable.

Then

$$\boldsymbol{z}^\top \boldsymbol{A}\boldsymbol{z} \geq 0 \quad \forall \boldsymbol{z} \in \ker(\boldsymbol{B})$$

is equivalent to

$$\boldsymbol{C} \succeq 0,$$

while

$$\boldsymbol{z}^\top \boldsymbol{A}\boldsymbol{z} > 0 \quad \forall \boldsymbol{z} \in \ker(\boldsymbol{B}) \setminus \{\boldsymbol{0}\}$$

is equivalent to

$$\boldsymbol{C} \succ 0.$$

If the constraints satisfy LICQ and there are m independent equality constraints, then

$$\operatorname{rank}(\boldsymbol{B}) = m, \qquad p = n - m.$$

Thus Sylvester's criterion can be applied directly to the reduced Hessian $\boldsymbol{C}$ when one wants to verify the **strict** second-order sufficient condition. If $p = 0$, the tangent space contains only the zero vector and the positivity condition on nonzero tangent directions is vacuous.

▪ **Example 3.1 — Example: reduced Hessian on the kernel.** Consider

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 0 & 1 \\ 3 & 1 & 0 \end{bmatrix}, \qquad \boldsymbol{B} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}.$$

The kernel of $\boldsymbol{B}$ is

$$\ker(\boldsymbol{B}) = \left\{ \begin{bmatrix} 0 \\ \alpha \\ \beta \end{bmatrix} : \alpha, \beta \in \mathbb{R} \right\}.$$

A convenient basis matrix is therefore

$$\boldsymbol{K} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix},$$

so that

$$\boldsymbol{z} = \boldsymbol{K} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

The reduced Hessian is

$$\begin{aligned} \boldsymbol{C} &= \boldsymbol{K}^\top \boldsymbol{A} \boldsymbol{K} \\ &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \end{aligned}$$

Its eigenvalues are $+1$ and -1, hence $\boldsymbol{C}$ is indefinite. Therefore the second-order necessary condition fails: there are tangent directions $\boldsymbol{z} \in \ker(\boldsymbol{B})$ for which

$$\boldsymbol{z}^\top \boldsymbol{A} \boldsymbol{z} < 0.$$

▪

There are two standard ways to test the reduced Hessian $\boldsymbol{C}$.

1. Since $\boldsymbol{C}$ is symmetric, it admits an orthogonal diagonalization

$$\boldsymbol{C} = \boldsymbol{Q} \boldsymbol{\Lambda} \boldsymbol{Q}^\top.$$

Hence

$$\boldsymbol{C} \succeq 0 \quad \Longleftrightarrow \quad \lambda_i(\boldsymbol{C}) \geq 0 \quad \forall i,$$

and

$$\boldsymbol{C} \succ 0 \quad \Longleftrightarrow \quad \lambda_i(\boldsymbol{C}) > 0 \quad \forall i.$$

2. For **positive definiteness**, Sylvester's criterion gives

$$\boldsymbol{C} \succ 0 \quad \Longleftrightarrow \quad \det(\boldsymbol{C}_k) > 0, \qquad k = 1, \dots, p,$$

where $\boldsymbol{C}_k$ are the leading principal submatrices of $\boldsymbol{C}$.

▪ **Example 3.2 — Application of Sylvester's criterion.** Consider

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix}.$$

The leading principal minors are

$$\Delta_1 = 1 > 0,$$

and

$$\Delta_2 = \det \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} = 5 - 4 = 1 > 0.$$

Therefore, by Sylvester's criterion,

$$\boldsymbol{A} \succ 0.$$

The same result follows from the eigenvalues. The characteristic polynomial is

$$\begin{aligned} p(\lambda) &= \det(\boldsymbol{A} - \lambda \boldsymbol{I}) \\ &= (1 - \lambda)(5 - \lambda) - 4 \\ &= \lambda^2 - 6\lambda + 1. \end{aligned}$$

Thus

$$\lambda_{1,2} = 3 \pm 2\sqrt{2}.$$

Both eigenvalues are strictly positive, in agreement with Sylvester's criterion.

▪

3.2 Positive semidefinite matrices

A crucial distinction must be made between **positive definite** and **positive semidefinite** matrices.

For a symmetric matrix, nonnegativity of only the leading principal minors is **not** sufficient to conclude positive semidefiniteness.

The correct characterization is

Theorem 3.2.1 — Principal-minor criterion for positive semidefiniteness. A real symmetric matrix $\boldsymbol{A}$ is positive semidefinite if and only if **all principal minors** of $\boldsymbol{A}$ are nonnegative:

$$\boxed{\boldsymbol{A} \succeq 0 \quad \Longleftrightarrow \quad \det(\boldsymbol{A}[I, I]) \geq 0 \quad \text{for every index set } I \subseteq \{1, \dots, n\}.}$$

Here $\boldsymbol{A}[I, I]$ denotes the principal submatrix obtained by selecting the same index set I for rows and columns.

This criterion involves, in general, more minors than Sylvester's criterion for positive definiteness. For numerical work, checking the eigenvalues of a symmetric reduced Hessian is often simpler when semidefiniteness rather than definiteness is required.

Equivalent Perturbation Characterization

There is also a useful exact perturbation characterization:

$$\boxed{\boldsymbol{A} \succeq 0 \quad \Longleftrightarrow \quad \boldsymbol{A} + \varepsilon \boldsymbol{I} \succ 0 \quad \forall \varepsilon > 0.}$$

Indeed, the eigenvalues of $\boldsymbol{A} + \varepsilon \boldsymbol{I}$ are

$$\lambda_i(\boldsymbol{A}) + \varepsilon.$$

Consequently, for a symmetric matrix,

$$\boldsymbol{A} \succeq 0 \quad \Longleftrightarrow \quad \det\left((\boldsymbol{A} + \varepsilon \boldsymbol{I})_k\right) > 0 \quad \forall k = 1, \dots, n, \quad \forall \varepsilon > 0.$$

The quantifier **for every** $\varepsilon > 0$ is essential. Merely checking the sign of a derivative at $\varepsilon = 0$ is not a valid general criterion.

▪ **Example 3.3 — Example: a positive semidefinite matrix.** Consider

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}.$$

Its eigenvalues are 0 and 5, hence

$$\boldsymbol{A} \succeq 0.$$

The principal minors are

$$1 \geq 0, \qquad 4 \geq 0, \qquad \det(\boldsymbol{A}) = 0,$$

which also verifies the principal-minor criterion.

Using the perturbation characterization,

$$\boldsymbol{A} + \varepsilon \boldsymbol{I} = \begin{bmatrix} 1+\varepsilon & 2 \\ 2 & 4+\varepsilon \end{bmatrix},$$

and its leading principal minors are

$$1 + \varepsilon > 0$$

and

$$\begin{aligned} \det(\boldsymbol{A} + \varepsilon \boldsymbol{I}) &= (1+\varepsilon)(4+\varepsilon) - 4 \\ &= 5\varepsilon + \varepsilon^2 > 0 \qquad \forall \varepsilon > 0. \end{aligned}$$

Therefore $\boldsymbol{A} + \varepsilon \boldsymbol{I} \succ 0$ for every $\varepsilon > 0$, consistently with $\boldsymbol{A} \succeq 0$.

▪

▪ **Example 3.4 — Counterexample: nonnegative leading minors do not imply semidefiniteness.** Consider

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

Its leading principal minors are

$$\Delta_1 = 1, \qquad \Delta_2 = 0, \qquad \Delta_3 = 0.$$

Thus all leading principal minors are nonnegative. Nevertheless, $\boldsymbol{A}$ is **not** positive semidefinite.

For example, the principal 2×2 submatrix obtained from rows and columns $\{1, 3\}$ is

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix},$$

whose determinant is

$$-1 < 0.$$

Hence the principal-minor criterion immediately shows that

$$\boldsymbol{A} \not\succeq 0.$$

The eigenvalues give the same conclusion:

$$\lambda_1 = 0, \qquad \lambda_{2,3} = 1 \pm \sqrt{3},$$

and

$$1 - \sqrt{3} < 0.$$

The perturbation test is also consistent. In fact,

$$\det(\boldsymbol{A} + \varepsilon \boldsymbol{I}) = \varepsilon\left(\varepsilon^2 + 2\varepsilon - 2\right),$$

which is negative for sufficiently small positive ε. Therefore $\boldsymbol{A} + \varepsilon \boldsymbol{I}$ is not positive definite for all $\varepsilon > 0$.

▪

3.3 Summary

For a real symmetric matrix $\boldsymbol{A}$:

$\boldsymbol{A} \succ 0$	$\iff$	all leading principal minors are positive,
$\boldsymbol{A} \prec 0$	$\iff$	$(-1)^k \det(\boldsymbol{A}_k) > 0 \quad \forall k,$
$\boldsymbol{A} \succeq 0$	$\iff$	all principal minors are nonnegative,
$\boldsymbol{A} \succeq 0$	$\iff$	$\boldsymbol{A} + \varepsilon \boldsymbol{I} \succ 0 \quad \forall \varepsilon > 0.$

For equality-constrained optimization, these tests should be applied to the reduced Hessian

$$\boldsymbol{C} = \boldsymbol{K}^\top \nabla_{\boldsymbol{x}}^2 \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) \boldsymbol{K},$$

where the columns of $\boldsymbol{K}$ form a basis of the tangent space

$$\ker D\boldsymbol{h}(\boldsymbol{x}^\star).$$

4. Inequality Constraints and KKT Conditions

4.1 Inequality Constraints

Let's introduce now the problem of minimizing a function $f(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ considering a set of p inequality constraints in the form $g_k(\boldsymbol{x}) \geq 0$.

$$\begin{aligned} &\text{minimize:} && f(\boldsymbol{x}) \\ &\text{subject to:} && g_k(\boldsymbol{x}) \geq 0 \qquad k = 1, \dots, p \end{aligned}$$

Throughout this section, as in the preceding chapters, gradients of scalar-valued functions are **row vectors**. Thus $\nabla g_k(\boldsymbol{x})\boldsymbol{d}$ is the directional derivative of g_k along the column direction $\boldsymbol{d}$.

The first approach to this problem is by trying to transform the inequality constraints g_k into equality constraints (so having $h_k(\boldsymbol{x}) = 0$). This can be done considering that each constraint can be expressed as

$$g_k(\boldsymbol{x}) = \varepsilon_k^2 \geq 0 \qquad \Longrightarrow \qquad h_k(\boldsymbol{x}, \varepsilon_k) = g_k(\boldsymbol{x}) - \varepsilon_k^2 = 0$$

Doing this process for each constraint we can determine a minimization problem that depends both on the function variable $\boldsymbol{x}$ but also on the so called **slack variables** $\boldsymbol{\varepsilon}$:

$$\begin{aligned} &\text{minimize:} && f(\boldsymbol{x}) \\ &\text{subject to:} && h_k(\boldsymbol{x}, \boldsymbol{\varepsilon}) = 0 \qquad k = 1, \dots, p \end{aligned}$$

This kind of problem increases the computational costs because it increases the number of optimization variables from n to $n + p$.

> **Note**
>
> We used the expression ε_k^2 and not ε_k because with this expression we don't need to specify one more inequality $\varepsilon_k \geq 0$.

▪ **Example 4.1** Let's consider the following problem:

$$\begin{aligned} \text{minimize:} \quad & f(x,y) = x^2 \\ \text{subject to:} \quad & x^2 + y^2 \leq 1 \\ & x + y \geq 0 \end{aligned}$$

In order to solve this problem we have to reduce the inequality constraints to equality constraints; considering the first constraint, that has to be expressed in the form $g_1(x,y) \geq 0$ and so

$$\begin{gathered} g_1(x,y) = 1 - x^2 - y^2 \geq 0 \\ \Downarrow \\ h_1(x,y,\varepsilon_1) = 1 - x^2 - y^2 - \varepsilon_1^2 \end{gathered}$$

In the same way it's possible to express the second inequality as the a constraint

$$h_2(x,y,\varepsilon_2) = x + y - \varepsilon_2^2$$

With this expression being set the problem reduces to the form

$$\begin{aligned} \text{minimize:} \quad & f(x,y,\epsilon_1,\epsilon_2) = x^2 \\ \text{subject to:} \quad & h_1(x,y,\epsilon_1,\epsilon_2) = 1 - x^2 - y^2 - \epsilon_1^2 = 0 \\ & h_2(x,y,\epsilon_1,\epsilon_2) = x + y - \varepsilon_2^2 = 0 \end{aligned}$$

With the problem stated in this way we can build the Lagrangian that depends on both the original variables x, y but also the slack variables ϵ_1, ϵ_2 and so

$$\begin{aligned} \mathcal{L}(\underbrace{x,y,\epsilon_1,\epsilon_2}_{x}, \underbrace{\lambda_1,\lambda_2}_{\lambda}) &= f(\boldsymbol{x}) - \lambda_1 h_1(\boldsymbol{x}) - \lambda_2 h_2(\boldsymbol{x}) \\ &= x^2 - \lambda_1\left(1 - x^2 - y^2 - \epsilon_1^2\right) - \lambda_2\left(x + y - \epsilon_2^2\right) \end{aligned}$$

To find the local minimum point we can start using the first necessary condition of the Lagrangian multiplier, so by determining all the stationary points such that $\nabla\mathcal{L} = \mathbf{0}$; in practice this means solving the following nonlinear system of 6 equations in 6 variables:

$$\begin{cases} \partial/\partial_x \mathcal{L} : & 2x + 2\lambda_1 x - \lambda_2 = 0 \\ \partial/\partial_y \mathcal{L} : & 2\lambda_1 y - \lambda_2 = 0 \\ \partial/\partial_{\epsilon_1} \mathcal{L} : & 2\lambda_1 \epsilon_1 = 0 \\ \partial/\partial_{\epsilon_2} \mathcal{L} : & 2\lambda_2 \epsilon_2 = 0 \\ \partial/\partial_{\lambda_1} \mathcal{L} : & x^2 + y^2 + \epsilon_1^2 - 1 = 0 \\ \partial/\partial_{\lambda_2} \mathcal{L} : & x + y - \epsilon_2^2 = 0 \end{cases}$$

Solving minimization problems with inequality constraints can be tricky using the Lagrangian multiplier method and so other algorithm (like the KKT that's going to be explained) are usually preferred. However this kind of system can be solved by cases; considering in fact the second and third equations we can discriminate the solution considering the various combination when ϵ_1, ϵ_2 are equal (or not) to zero.

A direct case analysis is possible, but this reformulation also shows an important drawback of squared slack variables: when an inequality is active, the corresponding slack variable is zero and the gradient of the equality constraint with respect to that slack variable vanishes. Therefore the usual equality-constraint regularity assumptions may fail exactly at active inequalities.

For this example the global minimum value is immediately seen to be

$$f_{\min} = 0,$$

because $f(x, y) = x^2 \geq 0$. It is attained at every feasible point with $x = 0$. The constraints then give

$$y^2 \leq 1, \qquad y \geq 0,$$

hence the complete set of minimizers is

$$\boxed{(x, y) = (0, y), \qquad y \in [0, 1].}$$

The slack-variable formulation reproduces these points with

$$\epsilon_1^2 = 1 - y^2, \qquad \epsilon_2^2 = y.$$

However, KKT conditions applied directly to the inequalities are cleaner and avoid the artificial degeneracy introduced by squared slack variables.

▪

4.1.1 Statement of the KKT conditions

The **Karush-Kuhn-Tucker** (KKT) conditions provide first-order necessary conditions for local optimality under a constraint qualification, and, with additional second-order assumptions, sufficient conditions for strict local optimality. They are used to analyze whether a point $\boldsymbol{x}^\star$ is a minimum point of a minimization problem subject to equality and inequality constraints, so for the problem

$$\begin{aligned} \text{minimize:} \quad & f(\boldsymbol{x}) \\ \text{subject to:} \quad & h_k(\boldsymbol{x}) = 0 \qquad k = 1, \dots, m \\ & g_k(\boldsymbol{x}) \geq 0 \qquad k = 1, \dots, p \end{aligned}$$

where $f, h_k, g_k : \mathbb{R}^n \to \mathbb{R}$ are real-valued functions.

To state these conditions, define the **set of active inequality constraints** $\mathcal{A}$ as

$$\mathcal{A}(\boldsymbol{x}^\star) = \left\{ k \mid g_k(\boldsymbol{x}^\star) = 0 \right\}$$

An inequality is active at $\boldsymbol{x}^\star$ when it holds with equality, i.e. when the point lies on the corresponding constraint boundary.

4.1.1.1 Qualified constraints

Given $\boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^p$ and $\boldsymbol{h} : \mathbb{R}^n \to \mathbb{R}^m$, with components of class C^1, the **Linear Independence Constraint Qualification** (LICQ) holds at a feasible point $\boldsymbol{x}^\star$ if

$$\left\{ \nabla g_k(\boldsymbol{x}^\star) \,:\, k \in \mathcal{A}(\boldsymbol{x}^\star) \right\} \cup \left\{ \nabla h_j(\boldsymbol{x}^\star) \,:\, j = 1, \dots, m \right\}$$

is a linearly independent set. Equivalently, the Jacobian obtained by stacking the equality-constraint gradients and the active inequality-constraint gradients has full row rank.

LICQ is a convenient constraint qualification, but it is stronger than necessary for first-order KKT conditions; weaker qualifications such as MFCQ can also be used.

▪ **Example 4.2 — qualification of a point.** Considering the minimization problem started in example Example 2.2 stated as

$$\begin{aligned} \text{minimize:} \quad & f(x, y, z) = x - y + z^2 \\ \text{subject to:} \quad & h_1(x, y, z) = x^2 + y^2 - 2 = 0 \\ & h_2(x, y, z) = x + z - 1 = 0 \end{aligned}$$

and considering the candidate point of minimum (resulting from the solution of the non-linear system)

$$
\begin{aligned}
x^\star &= \frac{\sqrt{3}}{2} - \frac{1}{2} \\
y^\star &= \frac{1}{2} + \frac{\sqrt{3}}{2} \\
z^\star &= \frac{3}{2} - \frac{\sqrt{3}}{2} \\
\lambda_1^\star &= \frac{1}{2} - \frac{\sqrt{3}}{2} \\
\lambda_2^\star &= 3 - \sqrt{3}
\end{aligned}
$$

In order to check if the chosen point is qualified it's necessary to check if the gradients of the equality constraint and the active ones (that in this case are not present because there aren't inequality constraints) are linearly independent. In order to solve this problem first we need to build the matrix H represent the gradients of the constraints:

$$
H(x,y,z) = \begin{bmatrix} \nabla h_1(x,y,z) \\ \nabla h_2(x,y,z) \end{bmatrix} = \begin{bmatrix} 2x & 2y & 0 \\ 1 & 0 & 1 \end{bmatrix}
$$

$$
\implies \qquad H^\star = H(x^\star, y^\star, z^\star) = \begin{bmatrix} \sqrt{3}-1 & 1+\sqrt{3} & 0 \\ 1 & 0 & 1 \end{bmatrix}
$$

It's straightforward to see that the lines of the matrix $H^\star$ are linearly independent, considering in fact the rank of $H^\star$ by analysing the columns we see that's equal to 2, so equalizing the number of the constraints.

▪

4.1.1.2 KKT first order necessary condition

Let $f \in C^1(\mathbb{R}^n)$ a function to minimize subject to the inequality

$$
\boldsymbol{g} \in C^1(\mathbb{R}^n, \mathbb{R}^p)
$$

and equality

$$
\boldsymbol{h} \in C^1(\mathbb{R}^n, \mathbb{R}^m)
$$

constraint maps. If $\boldsymbol{x}^\star$ is a local minimizer and LICQ holds at $\boldsymbol{x}^\star$, then there exist multipliers $\boldsymbol{\lambda}^\star \in \mathbb{R}^m$ and $\boldsymbol{\mu}^\star \in \mathbb{R}^p$ such that the following KKT conditions hold:

$$
\begin{aligned}
\nabla_x \mathcal{L}\left(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star\right) &= \mathbf{0} \\
h_k\left(\boldsymbol{x}^\star\right) &= 0 \qquad k = 1, \dots, m \\
g_k\left(\boldsymbol{x}^\star\right) &\geq 0 \qquad k = 1, \dots, p \\
\mu_k^\star g_k\left(\boldsymbol{x}^\star\right) &= 0 \qquad k = 1, \dots, p \\
\mu_k^\star &\geq 0 \qquad k = 1, \dots, p
\end{aligned}
\tag{4.1}
$$

These are, respectively, **stationarity**, **primal feasibility**, **complementary slackness**, and **dual feasibility**. With the convention $g_k(\boldsymbol{x}) \geq 0$ and $\mathcal{L} = f - \sum_k \mu_k g_k - \sum_j \lambda_j h_j$, the correct sign is $\mu_k \geq 0$.

where the Lagrangian is in this case defined as

$$
\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) - \sum_{k=1}^{p} \mu_k g_k(\boldsymbol{x}) - \sum_{k=1}^{m} \lambda_k h_k(\boldsymbol{x})
$$

▪ **Example 4.3** Solving minimization problem with the KKT conditions is generally easier; considering the case of the minimization problem in Example 4.1 the Lagrangian will depend only on 4 parameters (instead of 6 as in that example):

$$\mathcal{L}(x, y, \mu_1, \mu_2) = x^2 - \mu_1\left(1 - x^2 - y^2\right) - \mu_2\left(x + y\right)$$

Using the first-order KKT conditions, the stationarity and complementarity equations are

$$\begin{cases} \left.\begin{aligned} 2x + 2\mu_1 x - \mu_2 &= 0 \\ 2\mu_1 y - \mu_2 &= 0 \end{aligned}\right\} & \nabla_x \mathcal{L}\left(\boldsymbol{x}, \boldsymbol{\lambda}\right) \\ \left.\begin{aligned} \mu_1\left(1 - x^2 - y^2\right) &= 0 \\ \mu_2\left(x + y\right) &= 0 \end{aligned}\right\} & \mu_k g_k\left(\boldsymbol{x}\right) = 0 \\ \mu_1, \mu_2 \geq 0 \end{cases}$$

These equations must be considered together with **primal feasibility** $1 - x^2 - y^2 \geq 0$, $x + y \geq 0$ and **dual feasibility** $\mu_1, \mu_2 \geq 0$. The system can then be solved by cases:

1. If $\mu_1 = \mu_2 = 0$ then the last two equalities are always verified, while the first two becomes
$$\begin{cases} 2x = 0 \\ 0 = 0 \end{cases}$$
this means that $x = 0$, while y is a free parameter that must satisfy both the constraints $x + y \geq 0$ (so that $y \geq 0$) and $1 - x^2 - y^2 \geq 0$ (and so $1 - y^2 \geq 0$ that determines $-1 \leq y \leq 1$): considering both condition at the same time we have that
$$\begin{cases} y \geq 0 \\ -1 \leq y \leq 1 \end{cases} \implies$$
$$\text{solution: } x = \mu_1 = \mu_2 = 0 \quad y \in [0, 1]$$
2. When considering $\mu_1 \neq 0$ and $\mu_2 = 0$ the system of equations become
$$\begin{cases} 2x + 2\mu_1 x = 0 \\ 2\mu_1 y = 0 \\ \mu_1\left(1 - x^2 - y^2\right) = 0 \end{cases}$$
Based on the assumption of the value μ_1, from the second equation we determine that y must be equal to 0 and so the third equation becomes $1 - x^2 = 0$ whose solutions are ± 1. Considering that the first equation can be factorised as $2x(1 + \mu_1) = 0$ (and $x \neq 0$), then in order to satisfy the relation μ_1 must be -1, but this imply that μ_1 is negative and it's an unacceptable solution, so for $\mu_1 \neq 0, \mu_2 = 0$ no solutions of the system can be found;
3. Considering instead $\mu_1 = 0$ and $\mu_2 \neq 0$ the system becomes
$$\begin{cases} 2x - \mu_2 = 0 \\ -\mu_2 = 0 \\ \mu_2(x + y) = 0 \end{cases}$$
The second equation states that μ_2 must be equal to 0 and so the solution of this problem is the same as the first case considered;

4. Considering both variables $\mu_1, \mu_2 \neq 0$ we have to solve the full systems; considering the last two equations we have that

$$\begin{cases} x + y = 0 \\ 1 - x^2 - y^2 = 0 \end{cases}$$
$$\implies \quad x = -y$$
$$\implies \quad 1 - 2x^2 = 0$$

and so it means that the two possible solution for the systems are

$$(x, y) = \left(\pm \frac{1}{\sqrt{2}}, \mp \frac{1}{\sqrt{2}} \right).$$

Considering the first case where $x = 1/\sqrt{2}$ the first two equations become

$$\begin{cases} \frac{2}{\sqrt{2}}(1 + \mu_1) - \mu_2 = 0 \\ -\frac{2}{\sqrt{2}}\mu_1 - \mu_2 = 0 \end{cases}$$
$$\implies \quad \mu_2 = -\frac{2}{\sqrt{2}}\mu_1$$
$$\implies \quad \frac{4}{\sqrt{2}}\mu_1 + \frac{2}{\sqrt{2}} = 0$$
$$\implies \quad \mu_1 = -\frac{1}{2} \qquad \mu_2 = \frac{1}{\sqrt{2}}$$

In this case $\mu_1 < 0$ is an unacceptable solution of the problem, and also considering the point $x = -1/\sqrt{2}$ the system becomes with no considerable solutions

$$\begin{cases} -\frac{2}{\sqrt{2}}(1 + \mu_1) - \mu_2 = 0 \\ \frac{2}{\sqrt{2}}\mu_1 - \mu_2 = 0 \end{cases}$$
$$\implies \quad \mu_1 = -\frac{1}{2} \quad \mu_2 = -\frac{1}{\sqrt{2}}$$

The KKT solutions are therefore

$$(x, y) = (0, k), \qquad k \in [0, 1].$$

In this example no second-order argument is actually needed to establish optimality: $f(x, y) = x^2 \geq 0$ everywhere, and every point above has $f = 0$. Hence these points are exactly the **global minimizers**.

▪

4.1.1.3 KKT second order necessary conditions

Assume now that f, h_k and g_k are of class C^2. Let $\boldsymbol{x}^\star$ be a local minimizer at which LICQ holds, and let $(\boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star)$ be the KKT multipliers.

The relevant set of first-order feasible directions is the **critical cone**

$$C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) = \left\{ z \;:\; \begin{array}{ll} \nabla h_j(\boldsymbol{x}^\star) z = 0, & j = 1, \dots, m, \\ \nabla g_k(\boldsymbol{x}^\star) z = 0, & k \in \mathcal{A}(\boldsymbol{x}^\star),\ \mu_k^\star > 0, \\ \nabla g_k(\boldsymbol{x}^\star) z \geq 0, & k \in \mathcal{A}(\boldsymbol{x}^\star),\ \mu_k^\star = 0. \end{array} \right\}.$$

Then the second-order necessary condition is

$$\boxed{z^T \nabla^2_{xx} \mathcal{L}(x^\star, \lambda^\star, \mu^\star) z \geq 0 \qquad \forall z \in C(x^\star, \mu^\star).} \tag{4.2}$$

A simpler but weaker necessary test is obtained by restricting to the smaller subspace

$$\nabla h_j(x^\star) z = 0, \qquad \nabla g_k(x^\star) z = 0 \quad \forall k \in \mathcal{A}(x^\star).$$

If the Hessian is negative in one of these directions, the candidate cannot be a local minimum. Passing this weaker test, however, does **not** establish local minimality.

> **Warning**
>
> For second-order **sufficiency**, one must test the whole critical cone. Replacing the inequality condition for active constraints with zero makes the set of directions smaller and is therefore not a valid sufficient test in general.

Consider the problem

$$\begin{aligned} \min_{x,y} \quad & f(x,y) = -x^2 + y^2, \\ \text{subject to} \quad & g(x,y) = x \geq 0. \end{aligned}$$

At the feasible point

$$(x^\star, y^\star) = (0,0),$$

the inequality is active. With the convention

$$\mathcal{L}(x,y,\mu) = f(x,y) - \mu g(x,y), \qquad \mu \geq 0,$$

the stationarity condition is

$$\nabla f(0,0) - \mu^\star \nabla g(0,0) = 0.$$

Since

$$\nabla f(0,0) = \begin{bmatrix} 0 & 0 \end{bmatrix}, \qquad \nabla g(0,0) = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

we obtain

$$\mu^\star = 0.$$

If one incorrectly treats the active inequality as an equality, one tests only the kernel

$$\mathcal{K} = \{ z \,:\, \nabla g(0,0) z = 0 \} = \left\{ \begin{bmatrix} 0 \\ z_2 \end{bmatrix} \,:\, z_2 \in \mathbb{R} \right\}.$$

The Lagrangian Hessian is

$$\nabla^2_{xx} \mathcal{L}(0,0,0) = \begin{bmatrix} -2 & 0 \\ 0 & 2 \end{bmatrix}.$$

On the nonzero directions in $\mathcal{K}$,

$$z^T \nabla^2_{xx} \mathcal{L}\, z = 2 z_2^2 > 0.$$

Thus the incorrect "kernel of all active constraints' " test would declare the second-order sufficient condition satisfied.

However, because the active multiplier is zero, the correct critical cone is

$$\mathcal{C} = \{ z \,:\, \nabla g(0,0) z \geq 0 \} = \left\{ \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \,:\, z_1 \geq 0 \right\}.$$

The direction

$$z = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \in C$$

gives

$$z^T \nabla^2_{xx} \mathcal{L}\, z = -2 < 0.$$

Accordingly, $(0,0)$ is not a local minimizer: the feasible points $(t,0)$ with $t > 0$ satisfy

$$f(t,0) = -t^2 < f(0,0) = 0.$$

Hence

positive definiteness on the kernel of all active constraints $\not\Rightarrow$ second-order sufficiency

when an active inequality has a zero multiplier. The one-sided directions belonging to the critical cone must be retained.

▪ **Example 4.4 — kernel of the active constraints.** Continuing examples Example 2.2 and Example 4.2, on the minimum candidate point $(x^\star, y^\star, z^\star)$ we have the following matrix for the active constraints:

$$H^\star = \begin{bmatrix} \sqrt{3}-1 & 1+\sqrt{3} & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

Computing the kernel of the active constraints means so solving the following system of linear equation determined by $H^\star \boldsymbol{x} = \mathbf{0}$ and so

$$\begin{bmatrix} \sqrt{3}-1 & 1+\sqrt{3} & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

resulting in

$$\begin{cases} (\sqrt{3}-1)x + (1+\sqrt{3})y = 0 \\ x + z = 0 \end{cases}$$

Having 3 unknowns but only 2 equation means that we have to solve the problem in a parametric form; chosen $z = t$ we have $x = -t$ and so $y = t\frac{\sqrt{3}-1}{\sqrt{3}+1}$; choosing $t = \sqrt{3}+1$ (to simplify the results) we obtain the generator K of the kernel as

$$\boldsymbol{K} = \begin{pmatrix} -\sqrt{3}-1 \\ \sqrt{3}-1 \\ \sqrt{3}+1 \end{pmatrix}$$

▪

4.1.1.4 KKT second order sufficient conditions

Assume f, h_k and g_k are of class C^2, and let a feasible point $\boldsymbol{x}^\star$ satisfy the first-order KKT conditions with multipliers $(\boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star)$. If

$$\boldsymbol{z}^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star) \boldsymbol{z} > 0 \qquad \forall \boldsymbol{z} \in C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) \setminus \{0\},$$

then $\boldsymbol{x}^\star$ is a **strict local minimizer**. In finite dimension this is equivalent to the existence of a constant $\gamma > 0$ such that

$$z^T \nabla^2_{xx} \mathcal{L}\, z \geq \gamma \|z\|^2 \qquad \forall z \in C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star).$$

When all active inequality multipliers are strictly positive, the critical cone reduces to the null space of the gradients of the equality constraints and all active inequalities. If some

active multiplier is zero, the one-sided inequalities in the definition of the critical cone must be retained.

▪ **Example 4.5 — second order check.** Given the problem started in example Example 2.2 and carried on up to Example 4.4 states as

$$\begin{aligned} \text{minimize:} \quad & f(x,y,z) = x - y + z^2 \\ \text{subject to:} \quad & h_1(x,y,z) = x^2 + y^2 - 2 = 0 \\ & h_2(x,y,z) = x + z - 1 = 0 \end{aligned}$$

we found a candidate for minimum point in

$$\begin{aligned} x^\star &= \frac{\sqrt{3}}{2} - \frac{1}{2} \\ y^\star &= \frac{1}{2} + \frac{\sqrt{3}}{2} \\ z^\star &= \frac{3}{2} - \frac{\sqrt{3}}{2} \\ \lambda_1^\star &= \frac{1}{2} - \frac{\sqrt{3}}{2} \\ \lambda_2^\star &= 3 - \sqrt{3} \end{aligned}$$

that present the matrix $\boldsymbol{H}^\star$ for the active constraints with related kernel $\boldsymbol{K}$:

$$\boldsymbol{H}^\star = \begin{bmatrix} \sqrt{3}-1 & 1+\sqrt{3} & 0 \\ 1 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{K} = \begin{bmatrix} -\sqrt{3}-1 \\ \sqrt{3}-1 \\ \sqrt{3}+1 \end{bmatrix}$$

In order to check the second order necessary/sufficient condition we have to compute the hessian matrix $\nabla^2_{\boldsymbol{x}}\boldsymbol{\mathcal{L}}$ of the Lagrangian that's

$$\nabla^2_{\boldsymbol{x}}\boldsymbol{\mathcal{L}} = \boldsymbol{L} = \begin{bmatrix} -2\lambda_1 & 0 & 0 \\ 0 & -2\lambda_1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \qquad \Longrightarrow$$

$$\boldsymbol{L}^\star = \begin{bmatrix} \sqrt{3}-1 & 0 & 0 \\ 0 & \sqrt{3}-1 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

We have now to determine if this resulting matrix is positive semidefinite or positive definite on the kernel of active constraints in order to determine if the point is a minimum or not; this mean performing the following matrix multiplication:

$$\begin{aligned} \alpha \boldsymbol{K}^T \boldsymbol{L}^\star \boldsymbol{K} \alpha &= \alpha^2 \left[(\sqrt{3}-1)(\sqrt{3}+1)^2 + (\sqrt{3}-1)(\sqrt{3}-1)^2 + 2(\sqrt{3}+1)^2 \right] \\ &= 12\sqrt{3}\,\alpha^2 > 0 \qquad \text{for } \alpha \neq 0. \end{aligned}$$

Observing that for $\alpha \neq 0$ the product is always greater than zero we have that the hessian $L^\star$ is positive definite in the kernel of the active constraints, satisfying the second order sufficient condition that allow so to state that

$$\boldsymbol{x}^\star = \left(\frac{\sqrt{3}}{2} - \frac{1}{2}, \frac{1}{2} + \frac{\sqrt{3}}{2}, \frac{3}{2} - \frac{\sqrt{3}}{2} \right)$$

is a minimum.

▪

▪ **Example 4.6** Given the problem

$$\begin{aligned} \text{minimize:} \quad & f(x,y) = x^2 - xy \\ \text{subject to:} \quad & g_1(x,y) = 1 - x^2 - y^2 \geq 0 \\ & g_2(x,y) = 1 - (x-1)^2 - y^2 \geq 0 \end{aligned}$$

To solve this kind of problem using the KKT conditions we first need to build the Lagrangian of the problem that's equal to

$$\begin{aligned} \mathcal{L}(x,y,\mu_1,\mu_2) = x^2 - xy - \mu_1\left(1 - x^2 - y^2\right) \\ - \mu_2\left[1 - (x-1)^2 - y^2\right] \end{aligned}$$

At this point it's possible to construct the nonlinear system of equation whose solution are the stationary points candidate to be local minimum:

$$\begin{cases} \left.\begin{aligned} 2x - y + 2\mu_1 x + 2\mu_2(x-1) = 0 \\ -x + 2\mu_1 y + 2\mu_2 y = 0 \end{aligned}\right\} & \nabla_x \mathcal{L}\left(\boldsymbol{x}, \boldsymbol{\lambda}\right) \\ \left.\begin{aligned} \mu_1\left(1 - x^2 - y^2\right) = 0 \\ \mu_2\left[1 - (x-1)^2 - y^2\right] = 0 \end{aligned}\right\} & \mu_k g_k\left(\boldsymbol{x}\right) = 0 \\ \mu_1, \mu_2 \geq 0 \end{cases}$$

This system can be solved by cases of the value of μ_i and in particular

1. when $\mu_1 = \mu_2 = 0$ the system reduces to the form

$$\begin{cases} 2x - y = 0 \\ -x = 0 \end{cases} \qquad \Longrightarrow \qquad x = y = 0$$

   This solution satisfy the constraints, in fact $g_1(0,0) = 1 \geq 0$ and $g_2(0,0) = 0 \geq 0$, so this point can be used to compute the second order conditions;
2. considering all the other cases of μ_j the only other analytical solution to the system, found with `Mathematica`, determines the point

$$(x, y, \mu_1, \mu_2) = \left(\frac{1}{2}, \frac{\sqrt{3}}{2}, \frac{1}{\sqrt{3}} - \frac{1}{2}, \frac{1}{2} - \frac{\sqrt{3}}{6}\right)$$

In order to consider now the second order necessary/sufficient conditions we have first to evaluate the gradient $\nabla \boldsymbol{g}$ of the inequality constraint map (in respect to the variables x, y) resulting in the matrix

$$\nabla \boldsymbol{g} = \begin{bmatrix} -2x & -2y \\ -2(x-1) & -2y \end{bmatrix}$$

For the second order conditions is also important to define the hessian matrix of the Lagrangian respect to the variables x, y:

$$\nabla_x^2 \mathcal{L} = \begin{bmatrix} 2 + 2\mu_1 + 2\mu_2 & -1 \\ -1 & 2\mu_1 + 2\mu_2 \end{bmatrix}$$

We have now to check the conditions for the two local minimum candidates:

1. At $(x, y) = (0, 0)$ one has $\mu_1 = \mu_2 = 0$. The only active inequality is g_2, and

$$\nabla g_2(0,0) = \begin{bmatrix} 2 & 0 \end{bmatrix}.$$

   Since the active multiplier is zero, the critical cone is not the kernel of this gradient; it is the half-space

$$C = \left\{ z = (z_1, z_2)^T \ : \ 2z_1 \geq 0 \right\}.$$

The Lagrangian Hessian at this point is

$$\nabla^2_{xx}\mathcal{L}(0,0,0,0) = \begin{bmatrix} 2 & -1 \\ -1 & 0 \end{bmatrix}.$$

For $z = (0,1)^T \in \mathcal{C}$ the quadratic form is zero, so the second-order necessary condition is not violated, but the sufficient condition fails. In fact $(0,0)$ is **not** a local minimum. For example, points on the boundary of the second disk can be parameterized near the origin by

$$x = 1 - \sqrt{1-y^2} = \frac{y^2}{2} + O(y^4), \qquad y > 0,$$

and along these feasible points

$$f(x,y) = x^2 - xy = -\frac{1}{2}y^3 + O(y^4) < 0$$

for sufficiently small $y > 0$, whereas $f(0,0) = 0$.

2. The other KKT point is

$$(x,y) = \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right),$$

with positive multipliers

$$\mu_1 = \frac{1}{\sqrt{3}} - \frac{1}{2}, \qquad \mu_2 = \frac{1}{2} - \frac{\sqrt{3}}{6}.$$

Both inequalities are active and both multipliers are strictly positive. Therefore the critical cone is the null space of the two active gradients. Since those gradients are linearly independent, the critical cone contains only the zero vector. The second-order sufficient condition is then vacuously satisfied, and the KKT point is a strict local minimizer.

▪

▪ **Example 4.7 — Sample Exam Question.** Given the problem

$$\begin{aligned} \text{minimize:} \quad & f(x,y,z) = x - y + z^2 \\ \text{subject to:} \quad & h(x,y,z) = x^2 + y^2 - 2 = 0 \\ & g(x,y,z) = x^2 - 1 \geq 0 \end{aligned}$$

the solution can be obtained by using the KKT condition; in order to use them we first need to define the Lagrangian

$$\mathcal{L} = f - \lambda h - \mu g$$

of the problem that in this is of the form

$$\mathcal{L}(x,y,z,\lambda,\mu) = x - y + z^2 - \lambda\left(x^2 + y^2 - 2\right) - \mu\left(x^2 - 1\right)$$

The first order necessary condition allows to build the system of non-linear equation whose solution are candidates to be minimum point of f (respect to the constrains set), and so we have

$$\begin{cases} 1 - 2\lambda x - 2\mu x = 0 & : \partial/\partial_x \mathcal{L} \\ -1 - 2\lambda y = 0 & : \partial/\partial_y \mathcal{L} \\ 2z = 0 & : \partial/\partial_z \mathcal{L} \\ x^2 + y^2 - 2 = 0 & : h \\ x^2 - 1 \geq 0 & : g \\ \mu(x^2 - 1) = 0 & : \mu g \\ \mu \geq 0 & : \mu \end{cases}$$

Verifying that the point $\boldsymbol{x}^\star = (1, 1, 0)$ is a candidate minimum with $\lambda^\star = -\frac{1}{2}$ and $\mu^\star = 1$ (substituting the values in the system, all the relation are satisfied), the associated set of active constraints is represented by both h (in fact $1 + 1 - 2 = 0$) and g (indeed $1 - 1 = 0$) whose gradient can be regarded as

$$\nabla \mathcal{A} = \begin{bmatrix} \nabla h \\ \nabla g \end{bmatrix} = \begin{bmatrix} 2x & 2y & 0 \\ 2x & 0 & 0 \end{bmatrix} \quad \Longrightarrow$$

$$\boldsymbol{H} = \nabla \mathcal{A}(\boldsymbol{x}^\star) = \begin{bmatrix} 2 & 2 & 0 \\ 2 & 0 & 0 \end{bmatrix}$$

The constraints are so qualified because the gradients are linearly independent and related kernel, obtained by solving the equation $H\boldsymbol{x} = \boldsymbol{0}$, gives a generator of the space in the form of $\boldsymbol{K} = (0, 0, 1)$.

In order to determine if $\boldsymbol{x}^\star$ is a minimum point using the second order necessary/sufficient KKT conditions it's mandatory to compute the hessian $\nabla^2_{\boldsymbol{x}} \mathcal{L}$ of the Lagrangian:

$$\nabla^2_{\boldsymbol{x}} \mathcal{L} = \begin{bmatrix} -2(\lambda + \mu) & 0 & 0 \\ 0 & -2\lambda & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad \Longrightarrow$$

$$\boldsymbol{L} = \nabla^2_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^\star) = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

We now need to check if the matrix $\boldsymbol{L}$ is positive semidefinite/definite in the kernel of the active constraints, and so we have to compute

$$\begin{aligned} \alpha \boldsymbol{K}^T \boldsymbol{L} \boldsymbol{K} \alpha &= \alpha \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \alpha \\ &= \alpha \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \alpha \\ &= 2\alpha^2 \end{aligned}$$

Observing that this product is always greater than zero for $\alpha \neq 0$, then it means that L is positive definite in the kernel of the active constraints and so

$$\boldsymbol{x}^\star = (1, 1, 0) \qquad \text{is a strict local minimum point}$$

∎

▪ **Example 4.8 — Sample Exam Question.** Given the problem

$$\begin{aligned} \text{minimize:} \quad & f(x,y,z) = x^2 - y + z \\ \text{subject to:} \quad & h(x,y,z) = y - x = 1 \\ & g_1(x,y,z) = x \geq 1 \\ & g_2(x,y,z) = z \geq 1 \end{aligned}$$

the associated Lagrangian is

$$\begin{aligned} \mathcal{L}(x,y,z,\lambda,\mu_1,\mu_2) = x^2 - y + z - \lambda(y - x - 1) \\ - \mu_1(x-1) - \mu_2(z-1) \end{aligned}$$

The non-linear system associated to the the first order necessary KKT condition is so

$$\begin{cases} 2x + \lambda - \mu_1 = 0 & : \partial/\partial_x \mathcal{L} \\ -1 - \lambda = 0 & : \partial/\partial_y \mathcal{L} \\ 1 - \mu_2 = 0 & : \partial/\partial_z \mathcal{L} \\ y - x - 1 = 0 & : h \\ x - 1 \geq 0 & : g_1 \\ z - 1 \geq 0 & : g_2 \\ \mu_1(x-1) = 0 & : \mu_1 g_1 \\ \mu_2(z-1) = 0 & : \mu_2 g_2 \\ \mu_1, \mu_2 \geq 0 \end{cases}$$

The only solution of this system is

$$\begin{aligned} x^\star &= 1 \\ y^\star &= 2 \\ z^\star &= 1 \\ \lambda^\star &= -1 \\ \mu_1^\star = \mu_2^\star \quad &= 1 \end{aligned}$$

For such point the set of active constraint $\mathcal{A}$ is determined by all the constraints $\{h, g_1, g_2\}$ (in fact $g_1(\boldsymbol{x}^\star) = g_2(\boldsymbol{x}^\star) = 0$) whose gradient is so

$$\boldsymbol{H} = \begin{bmatrix} \nabla h \\ \nabla g_1 \\ \nabla g_2 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

This is a 3×3 matrix with linearly independent columns/rows and so the kernel obtained by solving $\boldsymbol{H}\boldsymbol{z} = \boldsymbol{0}$ consist only in the null vector $\boldsymbol{z} = \boldsymbol{0}$. Determined the hessian

$$\boldsymbol{L} = \nabla_{\boldsymbol{x}}^2 \mathcal{L} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

because both active inequality multipliers are strictly positive, the critical cone is obtained by imposing equality on the linearizations of h, g_1, and g_2. Hence it contains only the zero vector. The strict second-order condition is therefore satisfied vacuously. Consequently $\boldsymbol{x}^\star = (1, 2, 1)$ is a strict local minimizer. Notice that the feasible set is **not** locally isolated: feasible points with $x > 1$ or $z > 1$ exist arbitrarily close to $\boldsymbol{x}^\star$. What vanishes is the

critical cone, because every nonzero first-order feasible direction produces a strictly positive first-order increase of the objective under the KKT multipliers.

▪

4.2 KKT Conditions

Consider the nonlinear programming problem

$$\begin{aligned} \text{minimize:} \quad & f(\boldsymbol{x}) \\ \text{subject to:} \quad & h_i(\boldsymbol{x}) = 0, \quad i = 1, 2, \dots, m, \\ & g_\ell(\boldsymbol{x}) \geq 0, \quad \ell = 1, 2, \dots, p. \end{aligned} \tag{4.3}$$

Throughout this chapter, gradients of scalar-valued functions are **row vectors**, while directions and multiplier vectors are columns. Hence $\nabla f(\boldsymbol{x})\boldsymbol{d}$ is a scalar directional derivative, and the Jacobians of the constraint maps are obtained by stacking the row gradients.

We use the sign convention

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) - \sum_{k=1}^{m} \lambda_k h_k(\boldsymbol{x}) - \sum_{\ell=1}^{p} \mu_\ell g_\ell(\boldsymbol{x}). \tag{4.4}$$

With inequalities written as $g_\ell(\boldsymbol{x}) \geq 0$, the associated KKT multipliers satisfy $\mu_\ell \geq 0$.

4.2.1 Feasible set and active constraints

A point $\boldsymbol{x} \in \mathbb{R}^n$ is **feasible** if

$$h_k(\boldsymbol{x}) = 0, \qquad k = 1, \dots, m, \qquad \text{and} \qquad g_\ell(\boldsymbol{x}) \geq 0, \qquad \ell = 1, \dots, p.$$

The **feasible region** is therefore

$$\mathcal{F} = \{\boldsymbol{x} \in \mathbb{R}^n \,:\, \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{g}(\boldsymbol{x}) \geq \boldsymbol{0}\}. \tag{4.5}$$

At a feasible point $\boldsymbol{x}^\star$, an inequality constraint is **active** if it holds with equality. The active set is

$$\mathcal{A}(\boldsymbol{x}^\star) = \{\ell \,:\, g_\ell(\boldsymbol{x}^\star) = 0\}. \tag{4.6}$$

Inactive inequalities satisfy $g_\ell(\boldsymbol{x}^\star) > 0$ and, locally, do not restrict infinitesimal feasible motion. This observation is one of the reasons why tangent and linearized cones are central to first-order optimality theory.

4.2.2 Cones in $\mathbb{R}^n$ and local feasible geometry

Cones provide the natural language for describing the local geometry of a constrained optimization problem. Near a feasible point $\boldsymbol{x}^\star$, the nonlinear feasible set is replaced, to first order, by a set of admissible directions. Such sets are cones because, if a direction is admissible to first order, every positive rescaling of that direction is admissible as well. These notions are standard in convex and variational analysis; see [6, 7, 21].

Definition 4.2.1 — Cone. A set $\mathcal{K} \subseteq \mathbb{R}^n$ is a **cone** if

$$\boldsymbol{d} \in \mathcal{K}, \quad \alpha \geq 0 \qquad \Longrightarrow \qquad \alpha \boldsymbol{d} \in \mathcal{K}.$$

Thus a cone is invariant under nonnegative scaling. In particular, every nonempty cone contains the origin. Notice that a cone need not be convex.

A cone $\mathcal{K}$ is a **convex cone** if, in addition,

$$\boldsymbol{d}_1, \boldsymbol{d}_2 \in \mathcal{K}, \quad \alpha_1, \alpha_2 \geq 0 \qquad \Longrightarrow \qquad \alpha_1 \boldsymbol{d}_1 + \alpha_2 \boldsymbol{d}_2 \in \mathcal{K}.$$

Equivalently, a convex cone is closed under all nonnegative linear combinations.

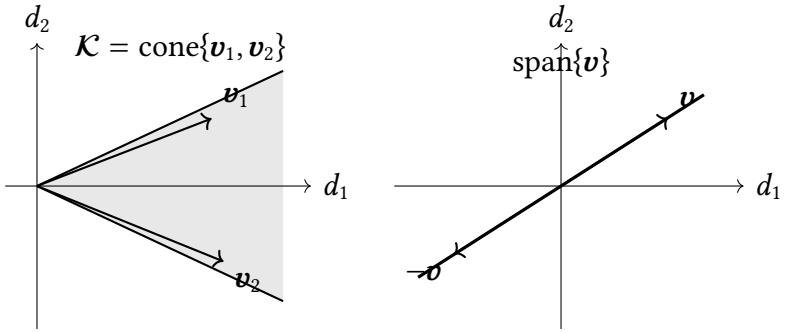
Important special classes of cones are:

- a cone is **pointed** if
$$\mathcal{K} \cap (-\mathcal{K}) = \{\boldsymbol{0}\};$$
geometrically, it contains no complete line through the origin;
- a cone is **solid** if its interior is nonempty;
- a cone is **polyhedral** if it can be represented as the intersection of finitely many homogeneous half-spaces,
$$\mathcal{K} = \{\boldsymbol{d} \in \mathbb{R}^n \,:\, \boldsymbol{A}\boldsymbol{d} \geq \boldsymbol{0}\};$$
- a linear subspace is a special convex cone because it is closed under both positive and negative scaling.

For vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_q \in \mathbb{R}^n$, the cone generated by these vectors is

$$\operatorname{cone}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_q\} = \left\{ \sum_{i=1}^{q} \alpha_i \boldsymbol{v}_i \,:\, \alpha_i \geq 0 \right\}.$$

It is a finitely generated, hence polyhedral, convex cone.



convex and pointed — a subspace is also a cone

Figure 4.1: Two elementary cones in $\mathbb{R}^2$. A pointed cone contains no complete line, whereas a linear subspace contains every direction together with its opposite.

4.2.2.1 A polyhedral cone from homogeneous inequalities

A polyhedral cone can be described either by its generating rays or by homogeneous linear inequalities. In two dimensions these two descriptions are particularly transparent. For example,

$$\mathcal{K} = \left\{ \boldsymbol{d} \in \mathbb{R}^2 \,:\, \begin{bmatrix} 1 & 0 \\ -1 & 2 \end{bmatrix} \boldsymbol{d} \geq \boldsymbol{0} \right\} = \operatorname{cone}\left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right\}.$$

Thus the half-space representation and the ray representation describe the same wedge. This primal/dual viewpoint is fundamental for polyhedral cones and for linearized feasible sets [7, 21].

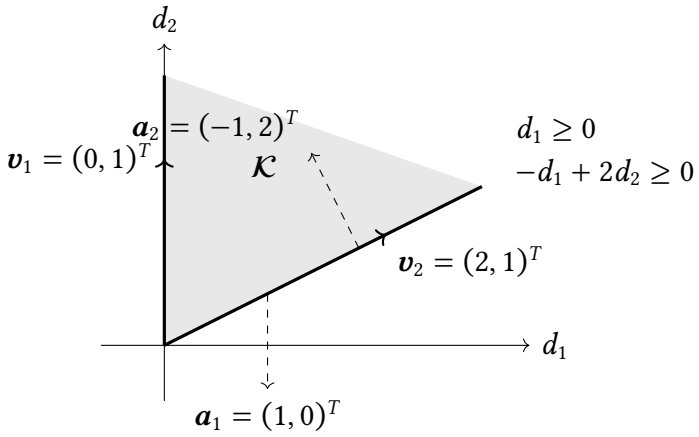


Figure 4.2: A polyhedral cone in $\mathbb{R}^2$. The same set can be represented as the intersection of homogeneous half-spaces or as the conic hull of its extreme rays.

4.2.2.2 Polar and dual cones

Let $\mathcal{K} \subseteq \mathbb{R}^n$ be a cone. Its **polar cone** is

$$\mathcal{K}^\circ = \left\{ \boldsymbol{y} \in \mathbb{R}^n \,:\, \boldsymbol{y}^T \boldsymbol{d} \leq 0 \quad \forall \boldsymbol{d} \in \mathcal{K} \right\},$$

whereas its **dual cone** is

$$\mathcal{K}^* = \left\{ \boldsymbol{y} \in \mathbb{R}^n \,:\, \boldsymbol{y}^T \boldsymbol{d} \geq 0 \quad \forall \boldsymbol{d} \in \mathcal{K} \right\}.$$

With these conventions,

$$\mathcal{K}^* = -\mathcal{K}^\circ.$$

Both $\mathcal{K}^\circ$ and $\mathcal{K}^*$ are always closed convex cones, even when $\mathcal{K}$ itself is not convex. If $\mathcal{K}$ is a closed convex cone, the bipolar theorem gives [7, 21]

$$(\mathcal{K}^\circ)^\circ = \mathcal{K}.$$

A useful two-dimensional example is the nonnegative orthant

$$\mathbb{R}^2_+ = \{(d_1, d_2)^T \,:\, d_1 \geq 0,\ d_2 \geq 0\}.$$

It is **self-dual**,

$$(\mathbb{R}^2_+)^* = \mathbb{R}^2_+,$$

while its polar is the opposite orthant,

$$(\mathbb{R}^2_+)^\circ = -\mathbb{R}^2_+.$$

This simple example is worth remembering: the dual cone collects vectors having nonnegative scalar product with every vector of the original cone, whereas the polar cone uses the opposite sign.

The polar cone is particularly convenient in constrained optimization because it converts a set of admissible directions into the corresponding set of normal directions.

For a general wedge, the polar and dual cones need not have the same opening angle as the primal cone. If the boundary rays of $\mathcal{K}$ rotate toward each other, the polar and dual cones

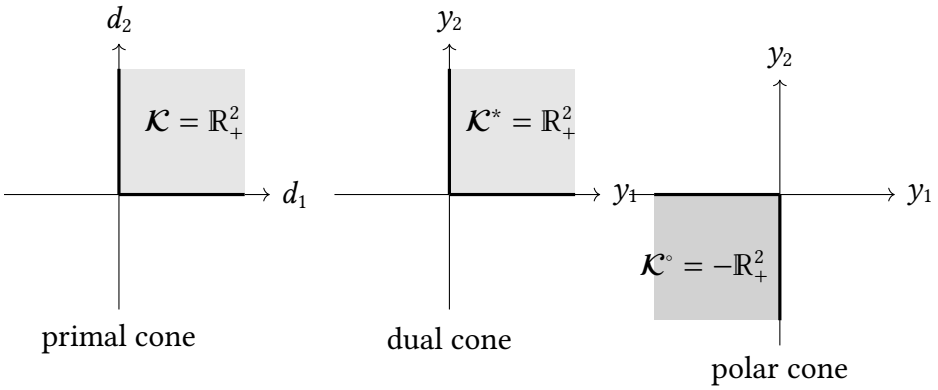


Figure 4.3: The nonnegative orthant in $\mathbb{R}^2$ is self-dual. Its polar is the opposite orthant because $\mathcal{K}^\circ = -\mathcal{K}^*$.

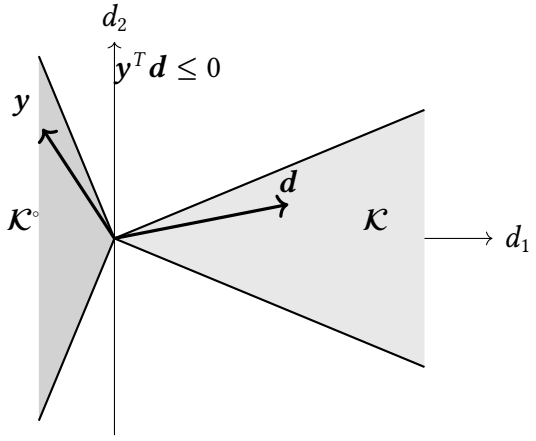


Figure 4.4: A convex cone $\mathcal{K}$ and its polar $\mathcal{K}^\circ$ in $\mathbb{R}^2$. Every polar vector forms a nonobtuse angle with $-\boldsymbol{d}$ for all $\boldsymbol{d} \in \mathcal{K}$.

rotate in the opposite sense and become wider. In $\mathbb{R}^2$ this angle picture is often the quickest way to understand duality.

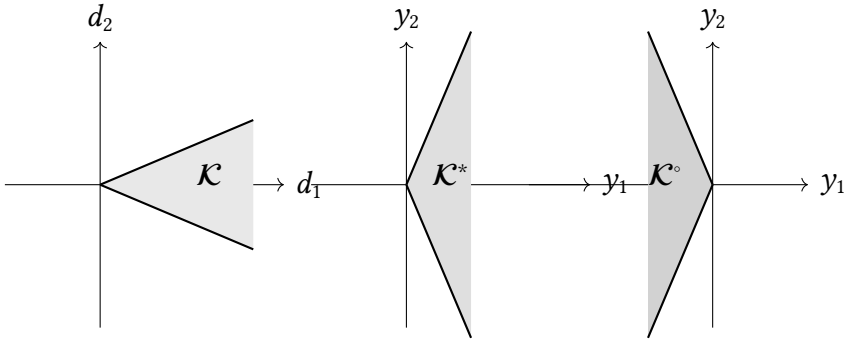


Figure 4.5: A two-dimensional wedge $\mathcal{K}$, its dual $\mathcal{K}^*$, and its polar $\mathcal{K}^\circ = -\mathcal{K}^*$. The boundary rays of the dual and polar are orthogonal to the boundary rays of the primal cone.

4.2.2.3 Tangent cone and normal cone of a feasible set

Let $\mathcal{F} \subseteq \mathbb{R}^n$ and let $\boldsymbol{x}^\star \in \mathcal{F}$. The **Bouligand tangent cone** (also called the contingent cone) is [6, 21]

$$T_{\mathcal{F}}(\boldsymbol{x}^\star) = \left\{ \boldsymbol{d} \;:\; \exists\, t_j \downarrow 0,\ \exists\, \boldsymbol{x}_j \in \mathcal{F},\ \boldsymbol{x}_j \to \boldsymbol{x}^\star,\ \frac{\boldsymbol{x}_j - \boldsymbol{x}^\star}{t_j} \to \boldsymbol{d} \right\}.$$

Thus $T_{\mathcal{F}}(\boldsymbol{x}^\star)$ contains precisely the directions that can be approached by feasible displacements at first order. If $\boldsymbol{x}^\star$ is an interior point of $\mathcal{F}$, then

$$T_{\mathcal{F}}(\boldsymbol{x}^\star) = \mathbb{R}^n.$$

At a smooth boundary point, the tangent cone is typically a half-space; at a corner, it is typically a proper polyhedral cone.

The associated **normal cone** is defined as the polar of the tangent cone [6, 21],

$$N_{\mathcal{F}}(\boldsymbol{x}^\star) = T_{\mathcal{F}}(\boldsymbol{x}^\star)^\circ.$$

Consequently,

$$\boldsymbol{y} \in N_{\mathcal{F}}(\boldsymbol{x}^\star) \qquad \Longleftrightarrow \qquad \boldsymbol{y}^T \boldsymbol{d} \le 0 \quad \forall \boldsymbol{d} \in T_{\mathcal{F}}(\boldsymbol{x}^\star).$$

For a differentiable objective, a local minimizer necessarily satisfies

$$\nabla f(\boldsymbol{x}^\star)\boldsymbol{d} \ge 0 \qquad \forall \boldsymbol{d} \in T_{\mathcal{F}}(\boldsymbol{x}^\star),$$

or, equivalently,

$$-\nabla f(\boldsymbol{x}^\star)^T \in N_{\mathcal{F}}(\boldsymbol{x}^\star). \tag{4.7}$$

Equation Equation 4.7 is the geometric form of first-order optimality: the negative objective gradient must be a feasible-set normal.

4.2.2.4 Smooth boundary versus corner

The shape of the tangent and normal cones depends strongly on the local geometry of the feasible set. At a smooth boundary point the tangent cone is a half-space and the normal cone is a ray. At a corner the tangent cone is a proper wedge and the normal cone is itself a wedge generated by the active constraint normals. This is precisely the geometry encoded by KKT multipliers [19, 21].

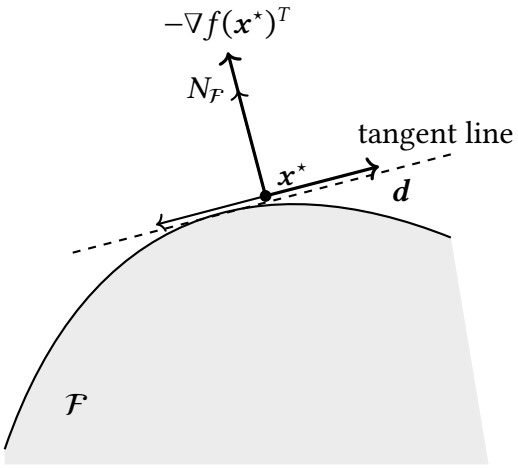
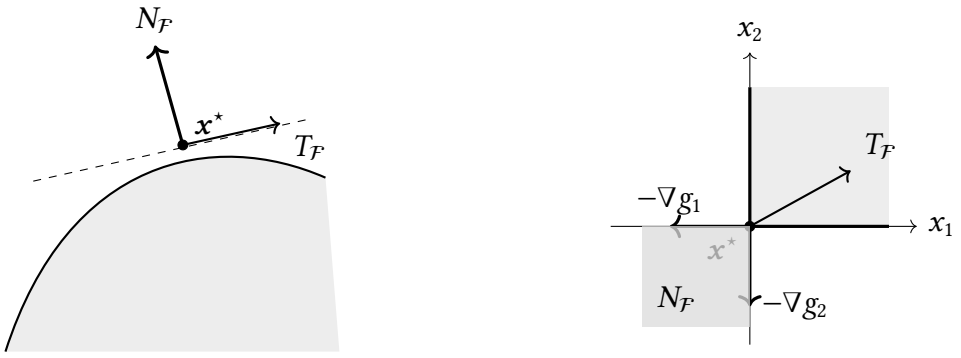


Figure 4.6: Local first-order geometry at a smooth boundary point. The tangent cone consists of feasible first-order directions (the half-space below the tangent line), while the normal cone contains outward normals. At a local minimum, $-\nabla f(x^\star)^T$ belongs to the normal cone.



(a) smooth boundary (b) corner

Figure 4.7: Local feasible geometry in $\mathbb{R}^2$. At a smooth point the normal cone is a ray. At a corner, the tangent cone is a wedge and the normal cone is generated by the active constraint normals.

4.2.2.5 Linearized cone for smooth nonlinear constraints

For the nonlinear program Equation 4.3, the first-order Taylor expansions at a feasible point $\boldsymbol{x}^\star$ are

$$h_k(\boldsymbol{x}^\star + t\boldsymbol{d}) = t\,\nabla h_k(\boldsymbol{x}^\star)\boldsymbol{d} + o(t),$$

and, for every active inequality $\ell \in \mathcal{A}(\boldsymbol{x}^\star)$,

$$g_\ell(\boldsymbol{x}^\star + t\boldsymbol{d}) = t\,\nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} + o(t).$$

This motivates the **linearized feasible cone**

$$\boxed{\mathcal{F}_L(\boldsymbol{x}^\star) = \left\{ \boldsymbol{d} \in \mathbb{R}^n \;:\; \begin{array}{ll} \nabla h_k(\boldsymbol{x}^\star)\boldsymbol{d} = 0, & k = 1, \dots, m, \\ \nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} \geq 0, & \ell \in \mathcal{A}(\boldsymbol{x}^\star) \end{array} \right\}.} \tag{4.8}$$

The true tangent cone and the linearized cone satisfy, in general,

$$T_{\mathcal{F}}(\boldsymbol{x}^\star) \subseteq \mathcal{F}_L(\boldsymbol{x}^\star). \tag{4.9}$$

The inclusion can be strict at degenerate points. Constraint qualifications are precisely the regularity assumptions that prevent the linearized model from being too large. For example, under standard first-order regularity conditions such as MFCQ, the tangent and linearized cones coincide:

$$T_{\mathcal{F}}(\boldsymbol{x}^\star) = \mathcal{F}_L(\boldsymbol{x}^\star).$$

This equality explains geometrically why a constraint qualification is needed in order to derive KKT multipliers from first-order optimality.

The distinction between $T_{\mathcal{F}}(\boldsymbol{x}^\star)$ and $\mathcal{F}_L(\boldsymbol{x}^\star)$ is easiest to see in a degenerate example. Consider

$$\mathcal{F} = \{(x_1, x_2) \;:\; -x_1^2 \geq 0\} = \{(0, x_2) \;:\; x_2 \in \mathbb{R}\}.$$

At the origin, the true tangent cone is the vertical line,

$$T_{\mathcal{F}}(\mathbf{0}) = \{(0, d_2) \;:\; d_2 \in \mathbb{R}\},$$

but $\nabla g(\mathbf{0}) = \mathbf{0}$, so the linearized inequality imposes no restriction and

$$\mathcal{F}_L(\mathbf{0}) = \mathbb{R}^2.$$

Thus the first-order linearization may contain many spurious directions. Constraint qualifications measure, in different strengths, how far this pathology is excluded.

With the present sign convention $g_\ell \geq 0$, the polar of the linearized cone is generated by arbitrary combinations of equality normals and **nonpositive** combinations of active inequality gradients. Equivalently, its negative is generated by the active gradients with nonnegative multipliers. Therefore KKT stationarity

$$\nabla f(\boldsymbol{x}^\star) - \sum_{k=1}^{m} \lambda_k^\star \nabla h_k(\boldsymbol{x}^\star) - \sum_{\ell \in \mathcal{A}(\boldsymbol{x}^\star)} \mu_\ell^\star \nabla g_\ell(\boldsymbol{x}^\star) = \mathbf{0}, \qquad \mu_\ell^\star \geq 0,$$

is exactly the multiplier representation of the geometric condition Equation 4.7.

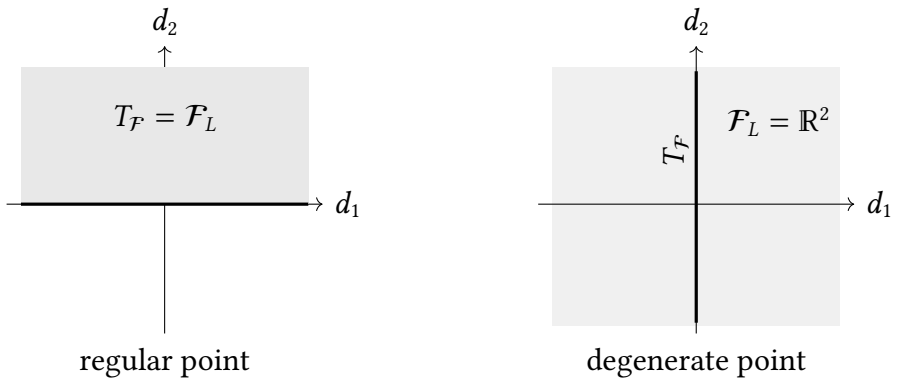


Figure 4.8: Tangent and linearized cones. At a regular point the linearized cone can reproduce the true tangent geometry. In the degenerate example $g(x_1, x_2) = -x_1^2 \geq 0$ at the origin, $\nabla g(\mathbf{0}) = \mathbf{0}$, hence $\mathcal{F}_L(\mathbf{0}) = \mathbb{R}^2$ although the true tangent cone is only the vertical line.

4.2.3 Fritz–John conditions

Before introducing KKT conditions, it is useful to state the more general Fritz–John theorem. Its importance is conceptual: it gives first-order necessary conditions without a constraint qualification. The price is the additional multiplier $\mu_0^\star$, which may vanish at a degenerate solution [4, 13].

Theorem 4.2.1 — Fritz–John theorem. Let f, $\boldsymbol{h}$, and $\boldsymbol{g}$ be continuously differentiable, and let $\boldsymbol{x}^\star$ be a local minimizer of Equation 4.3. Then there exist multipliers

$$\mu_0^\star \geq 0, \qquad \boldsymbol{\lambda}^\star \in \mathbb{R}^m, \qquad \boldsymbol{\mu}^\star \in \mathbb{R}^p, \quad \boldsymbol{\mu}^\star \geq \mathbf{0},$$

not all zero, such that

$$\mu_0^\star \nabla f(\boldsymbol{x}^\star) - \sum_{k=1}^{m} \lambda_k^\star \nabla h_k(\boldsymbol{x}^\star) - \sum_{\ell=1}^{p} \mu_\ell^\star \nabla g_\ell(\boldsymbol{x}^\star) = \mathbf{0},$$

and

$$\mu_\ell^\star g_\ell(\boldsymbol{x}^\star) = 0, \qquad \ell = 1, \dots, p.$$

The equality multipliers $\lambda_k^\star$ are unrestricted in sign. If $\mu_0^\star > 0$, all multipliers may be divided by $\mu_0^\star$, giving the normalization $\mu_0^\star = 1$. If $\mu_0^\star = 0$, the point is called **abnormal**: stationarity is then explained entirely by the constraint gradients and contains no contribution from the objective gradient.

4.2.4 Constraint qualifications

A **constraint qualification** (CQ) is a regularity condition imposed at a feasible point. Different CQs compare the nonlinear feasible geometry with its first-order linearization at different levels of strength. Their central role is to justify the passage from geometric first-order optimality to a multiplier representation of KKT type; see, for example, [3, 4, 19].

It is useful to distinguish two mechanisms:

- **LICQ** and **MFCQ** control the active constraint gradients directly; in particular, MFCQ excludes abnormal Fritz–John multipliers;
- **Abadie CQ** and **Guignard CQ** are geometric conditions stated in terms of the tangent and linearized cones. They need not exclude degeneracy by linear independence, but

they are still sufficient for first-order KKT necessity.

For smooth nonlinear programs, the standard implication chain is

$$\boxed{\text{LICQ} \implies \text{MFCQ} \implies \text{ACQ} \implies \text{GCQ}.} \tag{4.10}$$

None of the reverse implications holds in general [3, 4].

4.2.4.1 Linear independence constraint qualification (LICQ)

LICQ holds at a feasible point $\boldsymbol{x}^\star$ if

$$\{\nabla h_k(\boldsymbol{x}^\star) \,:\, k = 1, \dots, m\} \cup \{\nabla g_\ell(\boldsymbol{x}^\star) \,:\, \ell \in \mathcal{A}(\boldsymbol{x}^\star)\}$$

is linearly independent.

LICQ is convenient and strong. Besides implying the existence of KKT multipliers at a local minimizer, it also implies uniqueness of the KKT multipliers [4, 19].

4.2.4.2 Mangasarian–Fromovitz constraint qualification (MFCQ)

Definition 4.2.2 MFCQ holds at a feasible point $\boldsymbol{x}^\star$ if

1. the equality gradients $\nabla h_1(\boldsymbol{x}^\star), \dots, \nabla h_m(\boldsymbol{x}^\star)$ are linearly independent; and
2. there exists a direction $\boldsymbol{d}$ such that

$$\nabla h_k(\boldsymbol{x}^\star)\boldsymbol{d} = 0, \qquad k = 1, \dots, m,$$

and

$$\nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} > 0, \qquad \ell \in \mathcal{A}(\boldsymbol{x}^\star).$$

For the convention $g_\ell \geq 0$, the strict inequality points toward the interior of every active inequality.

An equivalent characterization is that

$$\sum_{\ell \in \mathcal{A}(\boldsymbol{x}^\star)} \alpha_\ell \nabla g_\ell(\boldsymbol{x}^\star) + \sum_{k=1}^{m} \beta_k \nabla h_k(\boldsymbol{x}^\star) = \boldsymbol{0}, \qquad \alpha_\ell \geq 0,$$

implies $\alpha_\ell = 0$ for every active inequality and $\beta_k = 0$ for every equality constraint.

Thus LICQ implies MFCQ, but MFCQ permits linear dependence among the active inequality gradients. Its essential geometric content is the existence of a direction that remains tangent to all equalities while moving strictly into the interior of all active inequalities [4, 19].

4.2.4.3 Abadie constraint qualification (ACQ)

The **Abadie constraint qualification** is stated directly in terms of the true tangent cone and the linearized cone [3, 4].

Definition 4.2.3 ACQ holds at $\boldsymbol{x}^\star$ if

$$\boxed{T_{\mathcal{F}}(\boldsymbol{x}^\star) = \mathcal{F}_L(\boldsymbol{x}^\star).} \tag{4.11}$$

Since the inclusion

$$T_{\mathcal{F}}(\boldsymbol{x}^\star) \subseteq \mathcal{F}_L(\boldsymbol{x}^\star)$$

always holds for differentiable constraints, ACQ requires that **every linearized feasible direction is a genuine tangent direction**. In other words, the first-order linear model introduces no spurious directions.

A convenient sufficient condition for ACQ is MFCQ. Hence

$$\text{MFCQ} \Longrightarrow \text{ACQ}.$$

The converse is false: the tangent and linearized cones may coincide even when no single direction moves strictly into all active inequalities.

4.2.4.4 Guignard constraint qualification (GCQ)

Guignard's condition is weaker. It does not require the tangent and linearized cones themselves to coincide; it requires only their polar cones to coincide [3, 4].

Definition 4.2.4 GCQ holds at $x^\star$ if

$$\boxed{T_{\mathcal{F}}(x^\star)^\circ = \mathcal{F}_L(x^\star)^\circ.} \tag{4.12}$$

Because

$$T_{\mathcal{F}}(x^\star) \subseteq \mathcal{F}_L(x^\star),$$

polarity reverses inclusion and always gives

$$\mathcal{F}_L(x^\star)^\circ \subseteq T_{\mathcal{F}}(x^\star)^\circ.$$

GCQ requires only that this inclusion be an equality. Therefore ACQ immediately implies GCQ:

$$\text{ACQ} \Longrightarrow \text{GCQ}.$$

The importance of GCQ follows directly from the geometric first-order condition. At a local minimizer,

$$-\nabla f(x^\star)^T \in T_{\mathcal{F}}(x^\star)^\circ.$$

Under GCQ,

$$-\nabla f(x^\star)^T \in \mathcal{F}_L(x^\star)^\circ.$$

The cone $\mathcal{F}_L(x^\star)$ is polyhedral, so its polar has the standard multiplier representation generated by equality normals and active inequality normals. This is exactly the KKT stationarity condition. Hence GCQ is already sufficient for first-order KKT necessity.

4.2.4.5 Geometric interpretation and hierarchy of the CQs

The four qualifications can be viewed as progressively weaker requirements:

- **LICQ:** all active constraint normals are linearly independent;
- **MFCQ:** equality normals are independent and there is a strict feasible first-order direction for all active inequalities;
- **ACQ:** the linearized feasible cone equals the true tangent cone;
- **GCQ:** only the polars of these two cones must coincide.

The hierarchy is illustrated schematically below.

The relation between ACQ and GCQ is particularly instructive. ACQ asks for **equality of feasible direction cones**, whereas GCQ only asks that these cones have the same normals. Since first-order optimality depends on normals, GCQ can be sufficient for KKT even when $T_{\mathcal{F}}$ and $\mathcal{F}_L$ are different.

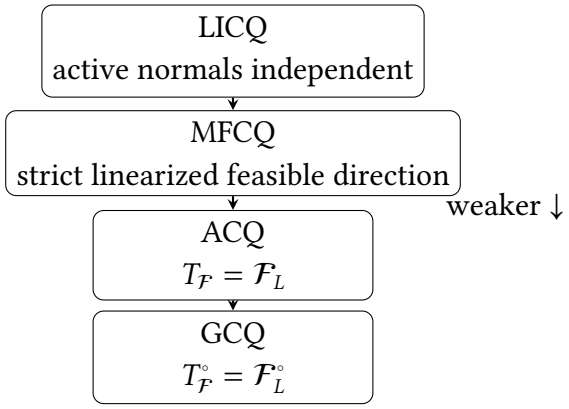


Figure 4.9: A standard hierarchy of first-order constraint qualifications for smooth nonlinear programming. Each condition implies the one below it; the reverse implications fail in general.

4.2.4.6 First-order feasible-arc qualification

A geometric condition stronger in spirit than merely requiring tangent-cone equality is to demand that every linearized feasible direction be realized by a feasible C^1 curve. Namely, for every $\boldsymbol{d}$ satisfying

$$\nabla h_k(\boldsymbol{x}^\star)\boldsymbol{d} = 0, \qquad \nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} \geq 0 \quad (\ell \in \mathcal{A}(\boldsymbol{x}^\star)),$$

there exists a feasible C^1 curve $\boldsymbol{x}(t)$ such that

$$\boldsymbol{x}(0) = \boldsymbol{x}^\star, \qquad \boldsymbol{x}'(0) = \boldsymbol{d}.$$

Such a feasible-arc property implies ACQ because every direction in the linearized cone is then, by construction, a true tangent direction.

4.2.4.7 McCormick second-order constraint qualification

McCormick's second-order constraint qualification is a stronger geometric condition used to derive strong second-order necessary conditions. In a feasible-arc formulation, for every direction $\boldsymbol{d}$ satisfying

$$\nabla h_k(\boldsymbol{x}^\star)\boldsymbol{d} = 0, \qquad \nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} = 0 \quad (\ell \in \mathcal{A}(\boldsymbol{x}^\star)),$$

one requires a twice continuously differentiable feasible curve $\boldsymbol{x}(t)$, for $t \geq 0$ sufficiently small, such that

$$\boldsymbol{x}(0) = \boldsymbol{x}^\star, \qquad \boldsymbol{x}'(0) = \boldsymbol{d},$$

and the inequalities active at $\boldsymbol{x}^\star$ remain active along the curve:

$$g_\ell(\boldsymbol{x}(t)) = 0, \qquad \ell \in \mathcal{A}(\boldsymbol{x}^\star).$$

This is a specialized second-order qualification; it should not be confused with MFCQ, which is a first-order constraint qualification [18].

4.2.5 KKT conditions under a constraint qualification

Under a suitable constraint qualification, the Fritz–John conditions reduce to the standard Karush–Kuhn–Tucker conditions. With the sign convention $g_\ell \geq 0$ used in this chapter, inequality multipliers are nonnegative [5, 14, 19].

Lemma 4.1 — First-order KKT theorem. Assume that f, h_k, and g_ℓ are continuously differentiable, $\boldsymbol{x}^\star$ is a local minimizer, and a constraint qualification sufficient for KKT necessity, such as LICQ, MFCQ, ACQ, or GCQ, holds at $\boldsymbol{x}^\star$. Then there exist multipliers $\boldsymbol{\lambda}^\star$ and $\boldsymbol{\mu}^\star$ such that

$$\begin{aligned} \nabla_x \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star) &= \mathbf{0}, \\ h_k(\boldsymbol{x}^\star) &= 0, \quad k = 1, \dots, m, \\ g_\ell(\boldsymbol{x}^\star) &\geq 0, \quad \ell = 1, \dots, p, \\ \mu_\ell^\star &\geq 0, \quad \ell = 1, \dots, p, \\ \mu_\ell^\star g_\ell(\boldsymbol{x}^\star) &= 0, \quad \ell = 1, \dots, p. \end{aligned}$$

The last condition is complementary slackness. In particular, $g_\ell(\boldsymbol{x}^\star) > 0$ implies $\mu_\ell^\star = 0$. Under LICQ the KKT multipliers are unique.

Remark on squared slack variables

The inequality $g_\ell(\boldsymbol{x}) \geq 0$ can be represented by

$$g_\ell(\boldsymbol{x}) - e_\ell^2 = 0.$$

The corresponding extended Lagrangian is

$$\mathcal{L}^e(\boldsymbol{x}, \boldsymbol{e}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) - \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}) - \sum_{\ell=1}^{p} \mu_\ell \big(g_\ell(\boldsymbol{x}) - e_\ell^2 \big).$$

Stationarity with respect to e_ℓ gives

$$2\mu_\ell e_\ell = 0,$$

which implies complementary slackness because $e_\ell^2 = g_\ell(\boldsymbol{x})$. However, the sign condition $\mu_\ell \geq 0$ does **not** follow from this first-order equation. For an active inequality, $e_\ell = 0$, it follows from the second-order necessary condition in the pure slack direction, since

$$\frac{\partial^2 \mathcal{L}^e}{\partial e_\ell^2} = 2\mu_\ell.$$

Therefore squared slack variables are useful as an interpretation, but they should not replace the standard first-order KKT theorem.

4.2.5.1 Strongly and weakly active inequalities

Once KKT multipliers are available, the active set can be split into

$$\begin{aligned} \mathcal{A}^+(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) &= \{\ell \in \mathcal{A}(\boldsymbol{x}^\star) \,:\, \mu_\ell^\star > 0\}, \\ \mathcal{A}^0(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) &= \{\ell \in \mathcal{A}(\boldsymbol{x}^\star) \,:\, \mu_\ell^\star = 0\}. \end{aligned}$$

The constraints in $\mathcal{A}^+$ are called **strongly active**, whereas those in $\mathcal{A}^0$ are **weakly active**. Thus

$$\mathcal{A}^+ \cap \mathcal{A}^0 = \emptyset, \qquad \mathcal{A}^+ \cup \mathcal{A}^0 = \mathcal{A}.$$

This distinction is immaterial for the first-order KKT equations but becomes essential in the definition of the critical cone and in second-order optimality conditions.

4.2.5.2 From the linearized cone to the critical cone

The linearized cone contains all directions that are feasible to first order. For second-order optimality, however, only directions along which the first-order variation of the objective

vanishes can be critical. This leads to the critical cone introduced next.

4.2.5.3 Critical cone

Split the active set into

$$\mathcal{A}^+(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) = \{\ell \in \mathcal{A}(\boldsymbol{x}^\star) \,:\, \mu_\ell^\star > 0\},$$
$$\mathcal{A}^0(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) = \{\ell \in \mathcal{A}(\boldsymbol{x}^\star) \,:\, \mu_\ell^\star = 0\}.$$

Using the linearized feasible cone defined in Equation 4.8, the **critical cone** consists of those first-order feasible directions for which the first-order variation of the objective vanishes [6, 19]:

$$C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) = \{\boldsymbol{d} \in \mathcal{F}_L(\boldsymbol{x}^\star) \,:\, \nabla f(\boldsymbol{x}^\star)\boldsymbol{d} = 0\}.$$

At a KKT point, stationarity gives

$$\nabla f(\boldsymbol{x}^\star)\boldsymbol{d} = \sum_{\ell \in \mathcal{A}(\boldsymbol{x}^\star)} \mu_\ell^\star \nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d},$$

because the equality terms vanish for every $\boldsymbol{d} \in \mathcal{F}_L$. Each term on the right-hand side is nonnegative. Hence the sum can vanish only when every strongly active constraint has zero directional derivative. Thus, for the convention $g_\ell \geq 0$,

$$\boxed{C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) = \left\{ \boldsymbol{d} \,:\, \begin{array}{ll} \nabla h_k(\boldsymbol{x}^\star)\boldsymbol{d} = 0, & k = 1, \dots, m, \\ \nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} = 0, & \ell \in \mathcal{A}^+, \\ \nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} \geq 0, & \ell \in \mathcal{A}^0 \end{array} \right\}.} \tag{4.13}$$

Lemma 4.2 — Second-order necessary condition. Assume f, h_k, and g_ℓ are C^2, $\boldsymbol{x}^\star$ is a local minimizer, and LICQ holds. Let $(\boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star)$ be the KKT multipliers. This critical-cone form is standard; see [6, 18, 19]. Then

$$\boxed{\boldsymbol{d}^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star)\boldsymbol{d} \geq 0 \qquad \forall \boldsymbol{d} \in C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star).}$$

A weaker but still valid necessary test is obtained by restricting to the smaller subspace in which **all** active inequalities are imposed as equalities:

$$\nabla g_\ell(\boldsymbol{x}^\star)\boldsymbol{d} = 0, \qquad \ell \in \mathcal{A}(\boldsymbol{x}^\star).$$

Because this tests fewer directions, passing this weaker test does not imply that the full second-order necessary condition holds.

Lemma 4.3 — Second-order sufficient condition. Suppose that $(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star)$ satisfies the KKT conditions. A standard sufficient condition is positivity of the Lagrangian Hessian on every nonzero critical direction [6, 19]. If

$$\boxed{\boldsymbol{d}^T \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star)\boldsymbol{d} > 0 \qquad \forall \boldsymbol{d} \in C(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) \setminus \{\boldsymbol{0}\},}$$

then $\boldsymbol{x}^\star$ is a strict local minimizer. Equivalently, in finite dimensions, one may require the uniform bound

$$\boldsymbol{d}^T \nabla^2_{xx} \mathcal{L}\, \boldsymbol{d} \geq \epsilon \|\boldsymbol{d}\|^2, \qquad \epsilon > 0, \quad \boldsymbol{d} \in C.$$

If strict complementarity holds, i.e. $\mu_\ell^\star > 0$ for every active inequality, then $\mathcal{A}^0 = \emptyset$ and the critical cone reduces to the kernel of the gradients of the equality constraints and all

active inequality constraints. In that special case the familiar reduced-Hessian test on that kernel is sufficient.

Without strict complementarity one must not simply replace the inequalities for weakly active constraints by equalities: doing so can miss feasible critical directions.

Counterexample: why weakly active constraints cannot be treated as equalities

Consider

$$\min_{x,y} \; -x^2 + y^2 \qquad \text{subject to} \qquad x \geq 0.$$

At $(0,0)$ the constraint is active and stationarity gives $\mu^\star = 0$. If the active constraint is incorrectly treated as an equality, one tests only $\boldsymbol{d} = (0, d_2)$ and obtains

$$\boldsymbol{d}^T \nabla^2 \mathcal{L}\, \boldsymbol{d} = 2d_2^2 > 0 \qquad (d_2 \neq 0).$$

The true critical cone is instead

$$C = \{(d_1, d_2) \,:\, d_1 \geq 0\}.$$

It contains $\boldsymbol{d} = (1,0)$, for which

$$\boldsymbol{d}^T \nabla^2 \mathcal{L}\, \boldsymbol{d} = -2 < 0.$$

Indeed $(t,0)$ is feasible for $t > 0$ and $f(t,0) = -t^2 < f(0,0)$. Thus $(0,0)$ is not a local minimizer.

Practical KKT checking procedure

1. **Form the Lagrangian**

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) - \sum_{k=1}^{m} \lambda_k h_k(\boldsymbol{x}) - \sum_{\ell=1}^{p} \mu_\ell g_\ell(\boldsymbol{x}).$$

2. **Solve the first-order KKT system**

$$\nabla_x \mathcal{L} = 0, \qquad \boldsymbol{h}(\boldsymbol{x}) = 0, \qquad \mu_\ell g_\ell(\boldsymbol{x}) = 0,$$

and retain only points satisfying

$$g_\ell(\boldsymbol{x}) \geq 0, \qquad \mu_\ell \geq 0.$$

3. **Identify the active sets**

$$\mathcal{A}^+ = \{\ell \,:\, g_\ell(\boldsymbol{x}^\star) = 0,\ \mu_\ell^\star > 0\}, \qquad \mathcal{A}^0 = \{\ell \,:\, g_\ell(\boldsymbol{x}^\star) = 0,\ \mu_\ell^\star = 0\}.$$

4. **Check a constraint qualification**, for example LICQ.
5. **Build the critical cone**

$$\nabla h_k \boldsymbol{d} = 0, \qquad \nabla g_\ell \boldsymbol{d} = 0 \ (\ell \in \mathcal{A}^+), \qquad \nabla g_\ell \boldsymbol{d} \geq 0 \ (\ell \in \mathcal{A}^0).$$

6. **Test the Lagrangian Hessian**

$$H_L = \nabla^2_{xx} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\mu}^\star).$$

The second-order necessary test is $\boldsymbol{d}^T H_L \boldsymbol{d} \geq 0$ on the critical cone; the sufficient test is $\boldsymbol{d}^T H_L \boldsymbol{d} > 0$ for every nonzero critical direction.

If $\mathcal{A}^0 = \emptyset$ (strict complementarity), the critical cone is a linear subspace. If the columns of K form a basis of that subspace, the tests reduce to checking

$$K^T H_L K \succeq 0 \quad \text{or} \quad K^T H_L K \succ 0.$$

4.3 Notes and references

The geometric language of tangent, normal, polar, and dual cones follows the standard treatment in variational and convex analysis [6, 7, 21]. Fritz–John and KKT conditions originate in the classical works [13, 14]. For nonlinear-programming constraint qualifications and the implication relations among LICQ, MFCQ, Abadie CQ, and Guignard CQ, see [3, 4]. The critical cone and the first- and second-order optimality conditions are treated systematically in [5, 6, 19]. McCormick's second-order condition is discussed in [18].

4.4 Chapter summary

The logical chain developed in this chapter is

feasible geometry
↓
tangent/linearized cones
↓
Fritz–John
↓
(CQ)
↓
KKT
↓
critical cone
↓
second-order tests

The key facts are:

- KKT consists of stationarity, primal feasibility, dual feasibility, and complementary slackness.
- The standard first-order CQ hierarchy is LICQ ⇒ MFCQ ⇒ ACQ ⇒ GCQ; GCQ is the weakest of these conditions and is already sufficient for KKT necessity.
- The second-order necessary condition requires the Lagrangian Hessian to be positive semidefinite on the critical cone.
- The corresponding sufficient condition requires strict positivity on every nonzero critical direction.
- Under strict complementarity, the critical cone becomes a linear subspace and the second-order test can be expressed through a reduced Hessian $K^T H_L K$.
- Weakly active constraints cannot, in general, be replaced by equalities in a second-order sufficient test.

II ODE

5 Introduction to ODE 67

5.1 Application of ODEs
5.2 Classification of ODEs
5.3 Solution Techniques

6 ODE Theory and Properties 73

6.1 Notation Conventions
6.2 Examples of ODEs
6.3 Vector Form of ODEs
6.4 General ODEs and Basic Definitions
6.5 Well-Posedness, Existence, and Uniqueness

Ordinary differential equations are a powerful and essential tool in both theoretical and applied mathematics. Their ability to model dynamic systems, whether simple or complex, makes them indispensable in understanding the behavior of many real-world systems.

With advanced techniques and numerical methods, ODEs continue to play a crucial role in scientific research, engineering design, and beyond. Understanding ODEs is therefore central to many fields of study, from physics and engineering to biology and economics.

Prerequisites, objectives, and expected outcomes

Prerequisites: calculus, elementary linear algebra, Taylor expansions, and basic programming literacy.
Objectives: connect qualitative ODE theory with Taylor, one-step and multistep methods, order conditions, adaptive error control, stiffness, and geometric integration.
Expected outcomes: the reader can derive consistency and stability conditions, determine the order of a Runge–Kutta tableau from its coefficients, verify that order numerically, select tolerances and step sizes, interpret a stability region, and test accuracy and invariant behavior computationally.

Contents

- Introduction to ODE
- ODE Theory and Properties
- Numerical integration of these equations — one-step and multistep methods, order conditions, absolute stability — is in the ODE part of the companion volume *Numerical Methods*, together with the Numerical ODE Solvers laboratory.

5. Introduction to ODE

In mathematics, an *ordinary differential equation* (ODE) is a type of *differential equation* that involves one or more unknown functions of a single independent variable, along with their derivatives. The term *ordinary* is used to distinguish these equations from *partial differential equations* (PDEs), which involve multiple independent variables. ODEs describe the relationship between a function and its rates of change with respect to one independent variable, providing a framework to model dynamic systems in various fields.

5.1 Application of ODEs

The importance of ordinary differential equations (ODEs) lies in their ability to model a vast range of phenomena across numerous scientific and engineering disciplines. ODEs are used to describe processes that evolve over time or space, making them powerful tools in understanding dynamic systems. These processes could include the motion of objects, population growth, the spread of diseases, fluid flow, and other natural or man-made systems that change continuously. The solutions to ODEs describe how a system evolves from one state to another, which allows us to predict, optimize, and gain deeper insights into the behavior of such systems.

Physics.

In **physics**, ODEs play a critical role in describing the motion of particles and objects under the influence of forces. One of the simplest yet most foundational examples is **Newton's second law of motion**, which states that the net force acting on an object equals its mass times its acceleration. This leads to the second-order differential equation:

$$m\frac{\mathrm{d}^2 z}{\mathrm{d}t^2} = \mathcal{F}\big(z(t), z'(t), t\big)$$

where $\mathcal{F}$ is the force law, $z(t)$ is the position of the object at time t, and m is its mass. In general the force may depend on time, position, and velocity. Solving this equation allows us

to determine the trajectory of an object when a known force is applied. For example, this law is used to predict the movement of planets in space or the motion of a pendulum.

Charged particle in an electromagnetic field.

The motion of a charged particle in the presence of electric ($\boldsymbol{E}$) and magnetic ($\boldsymbol{B}$) fields is described by the **Lorentz force law**. The Lorentz force law states that the force acting on a charged particle is the sum of the electric and magnetic forces:

$$\boldsymbol{F} = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B})$$

where: - $\boldsymbol{F}$ is the force acting on the particle, - q is the charge of the particle, - $\boldsymbol{E}$ is the electric field, - $\boldsymbol{B}$ is the magnetic field, - $\boldsymbol{v}$ is the velocity of the particle.

The Lorentz force law can be expressed as a second-order ordinary differential equation that governs the motion of the particle. Using Newton's second law ($\boldsymbol{F} = m\boldsymbol{a}$), where m is the mass of the particle and $\boldsymbol{a}$ is its acceleration, we get the following system of ODEs for the particle's motion:

$$m\frac{\mathrm{d}^2\boldsymbol{r}}{\mathrm{d}t^2} = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B})$$

where $\boldsymbol{r}(t)$ is the position vector of the particle at time (t), and the velocity

$$\boldsymbol{v}(t) = \frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}t}$$

This equation describes how the position and velocity of the particle change over time under the influence of electric and magnetic fields.

Chemistry.

In **chemistry**, ODEs are crucial in modeling reaction rates, chemical kinetics, and the processes of diffusion and mixing. A well-known example is the rate law in chemical reactions. For a simple first-order reaction, the rate at which a reactant is consumed is proportional to its concentration, which can be modeled with the following first-order ODE:

$$\frac{\mathrm{d}[A]}{\mathrm{d}t} = -k[A]$$

where $[A]$ is the concentration of the reactant at time t, and k is the rate constant. The solution to this equation describes how the concentration of the reactant decreases over time as the reaction progresses. For more complex reactions, such as second-order reactions or systems involving multiple reactants, systems of coupled ODEs are often used to model the interdependence between concentrations.

In **chemical engineering**, ODEs are used to describe the rates of flow in reactors, heat exchangers, and distillation columns, as well as in the modeling of fluid dynamics and transport processes. These equations are crucial for designing processes in which materials and energy are exchanged between different systems.

Engineering.

In **mechanical engineering**, ODEs are fundamental in the study of vibrations and structural dynamics. For instance, the motion of a damped harmonic oscillator, which can be modeled as:

$$m\frac{\mathrm{d}^2x}{\mathrm{d}t^2} + c\frac{\mathrm{d}x}{\mathrm{d}t} + kx = 0$$

where m is the mass, c is the damping coefficient, and k is the spring constant. This second-order ODE describes how an object behaves when subject to both restoring forces

(from a spring) and damping forces (such as friction). These equations are used in the design of suspension systems, buildings, and machinery that need to withstand vibrations.

In **civil engineering**, ODEs are applied to model the deflection and stress in beams and bridges, particularly in response to forces such as wind, gravity, or earthquakes. Structural engineers use these models to ensure safety and optimal design. A famous example is the beam bending equation, which can be modeled by the following ODE:

$$EI\frac{\mathrm{d}^4 y}{\mathrm{d}x^4} = q(x)$$

where E is the Young's modulus of the material, I is the moment of inertia, $y(x)$ is the deflection of the beam, and $q(x)$ is the load applied at position x. The solution of this equation helps engineers determine the bending of beams under various loading conditions.

Aerospace engineering.

In **aerospace engineering**, ODEs are used extensively in the modeling and simulation of flight dynamics, orbital mechanics, and spacecraft trajectory planning. For example, in the study of **orbital mechanics**, the motion of a satellite under the influence of gravity is governed by Newton's law of gravitation, leading to an ODE that describes the satellite's trajectory:

$$\frac{\mathrm{d}^2 \boldsymbol{r}}{\mathrm{d}t^2} = -GM\,\frac{\boldsymbol{r}}{\|\boldsymbol{r}\|^3}$$

where $\boldsymbol{r}(t)$ is the position vector relative to the center of the Earth, G is the gravitational constant, and M is the mass of the Earth. The scalar distance is $r = \|\boldsymbol{r}\|$. Solving this ODE allows engineers to predict satellite orbits, plan space missions, and ensure that spacecraft stay on the correct trajectory.

In **aircraft dynamics**, ODEs are used to describe the motion of an aircraft in the atmosphere. The aircraft's acceleration, velocity, and position are governed by forces such as lift, drag, thrust, and weight, and these forces are modeled by a system of ODEs. By solving these equations, aerospace engineers can optimize the performance of the aircraft, design control systems, and simulate flight behavior under different conditions, including turbulence or extreme weather.

Biology.

In **biology**, ODEs are used to model the growth of populations, the spread of diseases, and the dynamics of ecosystems. For example, the **logistic growth model**, a first-order ODE, describes how a population grows over time given a limited carrying capacity of the environment:

$$\frac{\mathrm{d}P}{\mathrm{d}t} = rP\left(1 - \frac{P}{K}\right)$$

where P is the population size, r is the growth rate, and K is the carrying capacity of the environment. This model is widely used in ecology to predict population dynamics, taking into account both growth and environmental limitations.

Epidemiology.

In **epidemiology**, ODEs are fundamental in modeling the spread of diseases. The classic **SIR model** (Susceptible-Infected-Recovered) is a system of ODEs used to study infectious diseases. It describes the rates at which individuals move from one compartment (e.g., susceptible, infected, recovered) to another:

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\beta\frac{SI}{N}, \quad \frac{\mathrm{d}I}{\mathrm{d}t} = \beta\frac{SI}{N} - \gamma I, \quad \frac{\mathrm{d}R}{\mathrm{d}t} = \gamma I$$

where S is the number of susceptible individuals, I is the number of infected individuals, R is the number of recovered individuals, and $N = S + I + R$ is the total population. (If S, I, R denote population fractions rather than counts, the factor $1/N$ is omitted.) These models help public health officials predict the spread of diseases and determine strategies for controlling outbreaks.

Economics.

In **economics**, ODEs are used to model the growth of markets, the value of assets, and various macroeconomic factors over time. For example, the **Solow growth model**, which is used to understand long-term economic growth, involves an ODE that describes the change in capital per worker over time:

$$\frac{\mathrm{d}k}{\mathrm{d}t} = sf(k) - \delta k$$

where k is the capital per worker, s is the savings rate, $f(k)$ is the output per worker, and δ is the depreciation rate. This equation helps economists understand how economies evolve, accounting for factors like savings, investment, and technological progress.

In **mathematical finance**, deterministic ODEs arise in models such as bond accumulation and deterministic term-structure dynamics. Asset prices are often modeled instead by **stochastic differential equations (SDEs)**, while option pricing leads, for example, to the Black–Scholes **partial differential equation (PDE)**. These are related subjects, but they should not be confused with ODEs.

Electronics.

In **electronics**, ODEs are crucial for modeling circuits, especially in the context of RC (resistor-capacitor) and RLC (resistor-inductor-capacitor) circuits, which are used in signal processing, filtering, and oscillation. For example, in an **RC circuit**, the voltage across the capacitor $V_C(t)$ changes over time according to the following first-order ODE:

$$\frac{\mathrm{d}V_C}{\mathrm{d}t} = -\frac{1}{RC}V_C(t)$$

where R is the resistance, C is the capacitance, and $V_C(t)$ is the voltage across the capacitor. This ODE describes how the capacitor charges and discharges over time, and solving it allows engineers to design circuits with specific time constants for various applications, such as smoothing signals or timing applications.

In an **RLC circuit**, which includes a resistor, an inductor, and a capacitor, the second-order ODE governing the current $I(t)$ is given by:

$$L\frac{\mathrm{d}^2I}{\mathrm{d}t^2} + R\frac{\mathrm{d}I}{\mathrm{d}t} + \frac{1}{C}I = 0$$

where L is the inductance, R is the resistance, and C is the capacitance. This ODE is essential for analyzing the behavior of oscillating circuits, resonance, and the decay of current in various electrical systems.

Medicine.

In **medicine**, ODEs are widely used to model physiological processes, drug interactions, and the spread of diseases. A classic example is **pharmacokinetics**, which describes the absorption, distribution, metabolism, and excretion (ADME) of drugs in the human body. For instance, the **one-compartment model** for the elimination of a drug can be represented by the following first-order ODE:

$$\frac{\mathrm{d}C}{\mathrm{d}t} = -kC$$

where $C(t)$ is the concentration of the drug in the bloodstream at time t, and k is the elimination rate constant. Solving this ODE helps determine how the drug concentration decreases over time, which is crucial for dosing and treatment scheduling.

In **cardiovascular modeling**, ODEs are used to describe the heart's pumping action and blood flow. The dynamics of the heart's contraction can be modeled by a system of ODEs that account for factors such as heart rate, stroke volume, and the resistance in the circulatory system. One example is the **Windkessel model**, which represents the pressure and volume in the arterial system as a set of ODEs:

$$C\frac{\mathrm{d}p}{\mathrm{d}t} = Q_{\text{in}}(t) - \frac{p(t) - p_{\text{out}}}{R}$$

where $p(t)$ is the arterial pressure, $Q_{\text{in}}(t)$ is the inflow, p_{out} is the downstream pressure, R is the peripheral resistance, and C is the arterial compliance. This model helps in understanding and treating cardiovascular diseases such as hypertension.

5.2 Classification of ODEs

ODEs can be classified in several ways, including by the order of the equation, its linearity, and whether the equation is autonomous or non-autonomous.

5.2.1 Order of an ODE

The order of an ODE refers to the highest derivative that appears in the equation. For example, an ODE involving the first derivative of a function is called a *first-order ODE*, while one that involves the second derivative is called a *second-order ODE*. For a scalar explicit ODE of order n, one normally prescribes n initial data, $y(t_0), y'(t_0), \dots, y^{(n-1)}(t_0)$. Whether these data actually determine a unique solution depends on the hypotheses of the existence-and-uniqueness theorem; order alone does not guarantee uniqueness.

5.2.2 Linearity of an ODE

An ODE is said to be *linear* if the unknown function and its derivatives appear only to the first power and are not multiplied together. For example, a first-order linear ODE can be written as:

$$\frac{\mathrm{d}}{\mathrm{d}t}y(t) + p(t)y(t) = q(t)$$

where $p(t)$ and $q(t)$ are known functions of the independent variable t, and $y(t)$ is the unknown function. Linear ODEs are well understood and are often solvable using analytical methods.

In contrast, a *nonlinear ODE* is one where the unknown function and its derivatives appear in nonlinear terms, such as products or powers. Nonlinear ODEs are more difficult to solve and often require numerical methods for finding approximate solutions. They are crucial in modeling complex real-world phenomena like turbulence in fluids or nonlinear population dynamics.

5.2.3 Autonomous vs. Non-autonomous ODEs

An ODE is called *autonomous* if the independent variable does not explicitly appear in the equation. In other words, the equation depends only on the dependent variable and its

derivatives. For example, the ODE:

$$\frac{\mathrm{d}}{\mathrm{d}t}y(t) = f(y(t))$$

is autonomous. If the independent variable appears explicitly in the equation, the ODE is *non-autonomous*.

5.3 Solution Techniques

While simple ODEs can often be solved analytically, many real-world problems lead to more complex equations that cannot be solved explicitly. In such cases, several advanced techniques come into play:

1. **Numerical Methods** For many ODEs, particularly nonlinear ones, explicit analytical solutions are not possible. In these cases, numerical methods such as Euler methods, Runge–Kutta methods, linear multistep methods, or collocation methods are used to approximate solutions. These methods involve discretizing the independent variable and solving for the function at each step.
2. **Qualitative Analysis** In addition to finding explicit solutions, it is often useful to study the *qualitative behavior* of ODE solutions. This involves analyzing properties such as stability, equilibrium points, and periodic solutions. The study of dynamical systems and stability theory allows for a deeper understanding of the behavior of systems described by ODEs, even without finding exact solutions. Techniques like phase plane analysis and Lyapunov's stability criteria are often employed to investigate the behavior of solutions in the long term.
3. **Series Solutions** For some ODEs, particularly those with variable coefficients or singularities, solutions may be found using power series expansions. Methods such as the Frobenius method are used to find series solutions near singular points, allowing for approximate solutions to be obtained.
4. **Transform Methods** In some cases, integral transforms convert an ODE into an algebraic equation. Once the equation is solved in the transformed domain, the inverse transform gives the solution to the original problem.

6. ODE Theory and Properties

A system of Ordinary Differential Equations (ODEs) consists of one or more equations that describe the relationship between one or more unknown functions and their derivatives (see [8],[12])

6.1 Notation Conventions

Throughout this part, t denotes the independent variable and y the unknown. The theory is stated for $\boldsymbol{y}(t)$ with values in $\mathbb{R}^d$, but the numerical chapters that follow are written in the **scalar case** so as not to burden the formulas: every scheme and every estimate carries over to systems by replacing y with $\boldsymbol{y}$, f with $\boldsymbol{f}$, and the absolute value with a norm. Quantities that are vectors or matrices even in the scalar case — such as the coefficients $\boldsymbol{A}$, $\boldsymbol{b}$, $\boldsymbol{c}$ of a Butcher tableau — stay bold.

The symbols used consistently are:

Table 6.1: Notation used throughout the ODE chapters.

symbol	meaning
t_k, h	mesh points and step size, $t_{k+1} = t_k + h$
$y(t_k)$	exact solution at t_k
y_k, $\boldsymbol{z}_k$	numerical approximation of $y(t_k)$ (scalar, vector)
$e_k = y(t_k) - y_k$	global error
τ_k	local truncation error / local defect
$\Psi(t, y; h)$	increment function of a one-step method
L	Lipschitz constant of f with respect to y
$\boldsymbol{J}_f$	Jacobian $\partial \boldsymbol{f} / \partial \boldsymbol{y}$

symbol	meaning
p	order of the method
$q = h\lambda$, $R(q)$, $\mathcal{S}$	scaled eigenvalue, stability function, stability region

6.2 Examples of ODEs

6.2.1 Scalar ODEs

In this section, we present several examples of **first-order** ordinary differential equations, highlighting their structure and how they are defined in terms of a function $f(y, x)$, with the state variable first and the independent variable second.

▪ **Example 6.1** **Nonlinear first-order ODE.**

$$\begin{cases} y' = y^2 + x \\ y(0) = 1 \end{cases} \implies \begin{array}{c} y'(x) = y(x)^2 + x = f(y(x), x), \\ f(y, x) = y^2 + x \end{array}$$

▪

▪ **Example 6.2** **ODE with mixed linear and nonlinear terms.**

$$\begin{cases} y' = xy + y^2 \\ y(0) = 2 \end{cases} \implies \begin{array}{c} y'(x) = xy(x) + y(x)^2 = f(y(x), x), \\ f(y, x) = xy + y^2 \end{array}$$

▪

▪ **Example 6.3** **ODE with a trigonometric nonlinearity.**

$$\begin{cases} z' = x^2 + \sin z \\ z(-1) = 0 \end{cases} \implies \begin{array}{c} z'(x) = x^2 + \sin z(x) = f(z(x), x), \\ f(z, x) = x^2 + \sin z \end{array}$$

▪

▪ **Example 6.4** **Logistic growth model.**

$$\begin{cases} p' = rp\left(1 - \dfrac{p}{K}\right) \\ p(0) = p_0 \end{cases} \implies p'(t) = rp(t)\left(1 - \frac{p(t)}{K}\right)$$

- Models **population growth**, where p represents the population size.
- **Parameters:**
  - r = growth rate.
  - K = carrying capacity of the environment.
- **Applications:** Used in **ecology** and **epidemiology**.

▪

▪ **Example 6.5** **Newton's law of cooling.**

$$\begin{cases} T' = -k(T - T_{\text{env}}) \\ T(0) = T_0 \end{cases} \implies T'(t) = -k(T - T_{\text{env}})$$

- Describes how an object cools down over time.
- **Parameters:**

- k = heat transfer coefficient.
- T_{env} = ambient temperature.

▪

6.2.2 Systems of ODEs

▪ **Example 6.6 A simple two-variable system.**

The following system describes the evolution of two state variables $x(t)$ and $y(t)$:

$$\left\{\begin{array}{l} \left.\begin{array}{l} x' = x + y \\ y' = \mathrm{e}^{x} - y \end{array}\right\} \text{ ODE equations} \\ \left.\begin{array}{l} x(0) = 1 \\ y(0) = 1 \end{array}\right\} \text{ Initial conditions} \end{array}\right.$$

- The first equation describes the rate of change of x, which depends on both x and y.
- The second equation models the evolution of y, which depends on an exponential function of x and y itself.
- The system starts from the initial state $(x(0), y(0)) = (1, 1)$.

▪

▪ **Example 6.7 A three-variable coupled system.**

This example involves three interdependent variables x, w, and z, with an explicit time dependence in the third equation.

$$\left\{\begin{array}{l} \left.\begin{array}{l} z' = x + z \\ w' = z \\ x' = wz + t \end{array}\right\} \text{ ODE equations} \\ \left.\begin{array}{l} x(0) = 1 \\ w(0) = 1 \\ z(0) = 1 \end{array}\right\} \text{ Initial conditions} \end{array}\right.$$

▪

6.3 Vector Form of ODEs

A system of first-order ODEs can be expressed in vector notation, which provides a more compact and structured representation of the problem. This notation is particularly useful in numerical methods and theoretical analysis.

6.3.1 System of ODEs in Scalar Form

Consider a system of n first-order ordinary differential equations:

$$y_1(t), y_2(t), \ldots, y_n(t), \qquad \text{functions of } t$$

The system is given by:

$$\left\{\begin{array}{l} \left.\begin{array}{l} y_1' = f_1(t, y_1, y_2, \dots, y_n) \\ y_2' = f_2(t, y_1, y_2, \dots, y_n) \\ \quad \vdots \\ y_n' = f_n(t, y_1, y_2, \dots, y_n) \end{array}\right\} \text{ODE system} \\ \\ \left.\begin{array}{l} y_1(0) = y_1^0, \\ y_2(0) = y_2^0, \\ \quad \vdots \\ y_n(0) = y_n^0, \end{array}\right\} \text{Initial conditions} \end{array}\right.$$

6.3.2 Compact Vector Form

To write this system in vector notation, define:

- The state vector:

$$\boldsymbol{y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_n(t) \end{pmatrix}$$

- The function that describes the system dynamics:

$$\boldsymbol{f}(\boldsymbol{y}, t) = \begin{pmatrix} f_1(y_1, y_2, \dots, y_n, t) \\ f_2(y_1, y_2, \dots, y_n, t) \\ \vdots \\ f_n(y_1, y_2, \dots, y_n, t) \end{pmatrix}$$

- The initial condition vector:

$$\boldsymbol{y}_0 = \begin{pmatrix} y_1^0 \\ y_2^0 \\ \vdots \\ y_n^0 \end{pmatrix}$$

Thus, the system can be rewritten in **vector form** as:

$$\begin{cases} \boldsymbol{y}' = \boldsymbol{f}(\boldsymbol{y}, t), \\ \boldsymbol{y}(0) = \boldsymbol{y}_0 \end{cases} \quad \Longrightarrow \quad \boldsymbol{y}'(t) = \boldsymbol{f}(\boldsymbol{y}(t), t)$$

This notation simplifies analysis and computation, particularly in numerical solvers.

6.3.3 Matrix Form of a Linear System

When the system is **linear**, meaning the functions $f_i(t, \boldsymbol{y})$ depend linearly on $\boldsymbol{y}$, the system can be written as:

$$\boldsymbol{y}' = \boldsymbol{A}(t)\boldsymbol{y} + \boldsymbol{b}(t)$$

where:

- $\boldsymbol{A}(t)$ is an $n \times n$ matrix of coefficients.
- $\boldsymbol{b}(t)$ is a vector of external inputs (or forcing terms).
- If $\boldsymbol{b}(t) = \boldsymbol{0}$, the system is **homogeneous**; otherwise, it is **inhomogeneous**.

A simple example of a **homogeneous linear system** is

$$\boldsymbol{y}' = \boldsymbol{A}\boldsymbol{y}, \qquad \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, \qquad \boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}.$$

This formulation allows for powerful solution techniques, such as matrix exponentiation and eigenvalue analysis.

Constant-coefficient linear systems

For the autonomous homogeneous system

$$\boldsymbol{y}'(t) = \boldsymbol{A}\boldsymbol{y}(t), \qquad \boldsymbol{y}(t_0) = \boldsymbol{y}_0,$$

the exact solution is

$$\boxed{\boldsymbol{y}(t) = e^{(t-t_0)\boldsymbol{A}}\boldsymbol{y}_0},$$

where the matrix exponential is defined by the absolutely convergent series

$$e^{t\boldsymbol{A}} = \boldsymbol{I} + t\boldsymbol{A} + \frac{t^2\boldsymbol{A}^2}{2!} + \frac{t^3\boldsymbol{A}^3}{3!} + \cdots.$$

Indeed,

$$\frac{\mathrm{d}}{\mathrm{d}t} e^{(t-t_0)\boldsymbol{A}} = \boldsymbol{A} e^{(t-t_0)\boldsymbol{A}}, \qquad e^{0\boldsymbol{A}} = \boldsymbol{I}.$$

If $\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}$ is diagonalizable, then

$$e^{t\boldsymbol{A}} = \boldsymbol{V} e^{t\boldsymbol{\Lambda}} \boldsymbol{V}^{-1},$$

so each eigenvalue λ_i produces a modal factor $e^{\lambda_i t}$. In particular, modes with $\operatorname{Re}\lambda_i < 0$ decay and modes with $\operatorname{Re}\lambda_i > 0$ grow. This observation is the bridge between the continuous dynamics of a linear system and the absolute-stability analysis of numerical methods.

For the inhomogeneous constant-coefficient system

$$\boldsymbol{y}' = \boldsymbol{A}\boldsymbol{y} + \boldsymbol{b}(t),$$

the variation-of-constants formula is

$$\boxed{\boldsymbol{y}(t) = e^{(t-t_0)\boldsymbol{A}}\boldsymbol{y}_0 + \int_{t_0}^{t} e^{(t-s)\boldsymbol{A}}\boldsymbol{b}(s)\,\mathrm{d}s}.$$

Equilibria and linearization of autonomous systems

Consider an autonomous system

$$\boldsymbol{y}' = \boldsymbol{f}(\boldsymbol{y}), \qquad \boldsymbol{y} \in \mathbb{R}^d.$$

A point $\boldsymbol{y}^\star$ is an **equilibrium** (or stationary point) if

$$\boldsymbol{f}(\boldsymbol{y}^\star) = \boldsymbol{0}.$$

The constant function $\boldsymbol{y}(t) \equiv \boldsymbol{y}^\star$ is then an exact solution. To understand the behavior of nearby trajectories, write

$$\boldsymbol{y} = \boldsymbol{y}^\star + \boldsymbol{u}.$$

If $\boldsymbol{f} \in C^1$, Taylor expansion gives

$$\boldsymbol{f}(\boldsymbol{y}^\star + \boldsymbol{u}) = \boldsymbol{f}(\boldsymbol{y}^\star) + \boldsymbol{J}_{\boldsymbol{f}}(\boldsymbol{y}^\star)\boldsymbol{u} + \boldsymbol{r}(\boldsymbol{u}), \qquad \frac{\|\boldsymbol{r}(\boldsymbol{u})\|}{\|\boldsymbol{u}\|} \to 0.$$

Since $\boldsymbol{f}(\boldsymbol{y}^\star) = \boldsymbol{0}$,

$$\boxed{\boldsymbol{u}' = J_f(\boldsymbol{y}^\star)\boldsymbol{u} + \boldsymbol{r}(\boldsymbol{u})}$$

and the associated **linearized system** is

$$\boxed{\boldsymbol{u}' = J_f(\boldsymbol{y}^\star)\boldsymbol{u}.}$$

The eigenvalues of the Jacobian matrix $J_f(\boldsymbol{y}^\star)$ determine the local behavior whenever the equilibrium is **hyperbolic**, i.e. no eigenvalue has zero real part.

- If every eigenvalue satisfies $\operatorname{Re}\lambda_i < 0$, the equilibrium is locally asymptotically stable.
- If at least one eigenvalue has $\operatorname{Re}\lambda_i > 0$, the equilibrium is unstable.
- If eigenvalues lie on the imaginary axis, linearization alone is generally inconclusive; nonlinear terms may determine stability.

For a hyperbolic equilibrium, the Hartman–Grobman theorem gives a much stronger qualitative statement: sufficiently close to the equilibrium, the nonlinear flow is topologically equivalent to its linearization. For numerical analysis, the same eigenvalues also reveal the time scales of the local dynamics and are therefore closely related to stiffness.

Example: planar saddle and stable node

For

$$\boldsymbol{y}' = A\boldsymbol{y}, \qquad A = \begin{pmatrix} -1 & 0 \\ 0 & -10 \end{pmatrix},$$

both eigenvalues are negative, so the origin is asymptotically stable. The two decay time scales, 1 and $1/10$, are already separated. Replacing the second eigenvalue by $+10$ makes the origin a saddle and therefore unstable.

6.4 General ODEs and Basic Definitions

Before exploring numerical solution methods, it is useful to briefly review some key definitions and properties of ordinary differential equations (ODEs) and initial value problems (IVPs).

Definition 6.4.1 — Ordinary Differential Equation. An *ordinary differential equation* (ODE) is an equation involving a function $z(t)$, defined on an interval $I \subset \mathbb{R}$, with values in either real or complex numbers, or in the vector spaces $\mathbb{R}^d$ or $\mathbb{C}^d$. The general form of an ODE is:

$$f\left(z(t), z'(t), z''(t), \ldots, z^{(n)}(t), t\right) = 0$$

Here, f is a function that depends on time t, the unknown function $z(t)$, and its derivatives up to order n. The *order* of the differential equation is determined by the highest derivative $z^{(n)}(t)$ that appears.

If $z(t)$ takes values in a space of dimension $d > 1$, the equation is called a *system of differential equations*.

This implicit form is very general. In many IVPs of interest, the highest derivative can be isolated, leading to an explicit ODE:

Definition 6.4.2 — Explicit ODE. An *explicit* first-order differential equation has the form:

$$z'(t) = f\left(z(t), t\right) \quad \text{or briefly:} \quad z' = f(z, t) \tag{6.1}$$

For a higher-order equation (order n), the equation is called explicit if it can be written

as:

$$z^{(n)}(t) = f\left(z(t), z'(t), z''(t), \dots, z^{(n-1)}(t), t\right)$$

Definition 6.4.3 — Autonomous ODE. A differential equation of the form Equation 6.1 is called *autonomous* if the right-hand side f does not explicitly depend on t, meaning:

$$z'(t) = f\left(z(t)\right)$$

Numerical methods are often designed specifically for first-order ODEs. Therefore, higher-order ODEs must be transformed into an equivalent system of first-order equations before applying these methods.

Lemma 6.1 — Reduction to First-Order Systems. Every explicit ODE of order n can be rewritten as an equivalent first-order system by introducing the derivatives as new state variables. A general implicit higher-order equation can likewise be written as a first-order **implicit** system $F(Y, Y', t) = 0$; it need not define an explicit ODE unless suitable solvability conditions hold.

Proof. To transform a higher-order differential equation into a system of first-order equations, we introduce auxiliary variables for each derivative of $z(t)$:

$$\begin{aligned} z_0(t) &= z(t), \\ z_1(t) &= z'(t), \\ &\vdots \\ z_{n-1}(t) &= z^{(n-1)}(t) \end{aligned}$$

This allows us to rewrite the original n-th order differential equation as a system of n first-order equations:

$$\begin{pmatrix} z_0'(t) - z_1(t) \\ z_1'(t) - z_2(t) \\ \vdots \\ z_{n-2}'(t) - z_{n-1}(t) \\ f\left(z_0(t), z_1(t), \dots, z_{n-1}(t), z_{n-1}'(t), t\right) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

In the case of an explicit equation, this system becomes:

$$\begin{pmatrix} z_0'(t) \\ z_1'(t) \\ \vdots \\ z_{n-2}'(t) \\ z_{n-1}'(t) \end{pmatrix} = \begin{pmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ z_{n-1}(t) \\ f\left(z_0(t), z_1(t), \dots, z_{n-1}(t), t\right) \end{pmatrix}$$

■

Most often, ODEs come with additional information in the form of an *initial condition*, leading to what is known as an *initial value problem* (IVP), also referred to as a **Cauchy problem**. The initial condition specifies the value of the function z at a given point t_0 within the domain, helping to define a unique solution to the differential equation.

Definition 6.4.4 — Initial Value Problem (IVP). Given an initial point

$$(t_0, z_0) \in \mathbb{R} \times \mathbb{R}^d$$

let $f(z(t), t)$ be a function with values in $\mathbb{R}^d$, defined in a neighborhood

$$I \times U \subset \mathbb{R} \times \mathbb{R}^d$$

around the initial point. The *initial value problem* (IVP) is then the task of finding a function $z(t)$ such that:

$$\begin{cases} z'(t) = f(z(t), t) \\ z(t_0) = z_0 \end{cases}$$

In the study of ordinary differential equations (ODEs), the concept of a local solution plays a fundamental role in understanding the behavior of dynamical systems. Given an initial value problem (IVP), a local solution refers to a function that satisfies the differential equation and initial condition within a limited region around the initial time.

Local solutions are particularly important because existence theorems, such as the Picard-Lindelöf Theorem 6.5.3, often guarantee solutions only within a small interval rather than over the entire domain. This is due to factors like nonlinearities in the system, singularities, or rapidly growing solutions that may prevent global existence.

Definition 6.4.5 — Local Solution. A *local solution* to the IVP is a continuously differentiable function $z(t)$ that satisfies $z(t_0) = z_0$. This solution exists within some neighborhood J of t_0, where both $z(t)$ and $f(z(t), t)$ are well-defined, and the equation holds for all $t \in J$.

6.5 Well-Posedness, Existence, and Uniqueness

6.5.1 Definition of a Well-Posed Initial Value Problem

A mathematical problem is said to be **well-posed** if it satisfies the following conditions (**Hadamard conditions**):

- **Existence**: a solution exists.
- **Uniqueness**: the solution is unique.
- **Continuous Dependence**: the solution continuously depends on the initial data.

If any of these conditions fail, the problem is considered **ill-posed**, meaning small changes in the initial conditions can lead to large deviations in the solution.

6.5.2 Existence of Solutions

The following result provides a **general existence theorem** for first-order differential equations.

Theorem 6.5.1 — Peano's Existence Theorem. Let $f(z, t)$ be a continuous function defined on the closed set:

$$\bar{D} = \left\{ (t, z) \in \mathbb{R} \times \mathbb{R}^d \,\middle|\, |t - t_0| \leq \alpha, \|z - z_0\| \leq \beta \right\}$$

where $\alpha, \beta > 0$. Then, there exists a continuously differentiable solution $z(t) \in C^1(I)$ on the interval $I = [t_0 - T, t_0 + T]$, where:

$$T = \min\left(\alpha, \frac{\beta}{M}\right), \quad M = \max_{(t,z)\in\bar{D}} \|f(z, t)\|$$

This theorem **guarantees existence** but does **not** ensure uniqueness. If $M = 0$, the solution is constant and one may simply take $T = \alpha$.

6.5.2.1 Continuation of solutions

Local existence does not by itself imply existence for all time. A precise continuation statement is the following.

Theorem 6.5.2 — Continuation theorem. Let $D \subset \mathbb{R} \times \mathbb{R}^d$ be open and let $f : D \to \mathbb{R}^d$ be continuous and locally Lipschitz with respect to z. Let $z : (\alpha, \beta) \to \mathbb{R}^d$ be the maximal solution of the IVP. If $\beta < +\infty$, then the graph $(t, z(t))$ leaves every compact subset of D as $t \uparrow \beta$. Equivalently, if the graph remains in a compact subset $K \Subset D$, the solution can be extended beyond β.

Thus finite-time termination can occur because the solution becomes unbounded or because its graph approaches the boundary of the domain. Merely saying that f is bounded at the points already reached is not, by itself, a complete continuation criterion.

6.5.3 Uniqueness of Solutions

To ensure *uniqueness*, we require stronger conditions such as **Lipschitz continuity**. We also introduce **Grönwall's Lemma**, a crucial tool for proving uniqueness.

6.5.3.1 Lipschitz Condition

A function $f : D \to \mathbb{R}^d$, with $D = I \times \Omega \subset \mathbb{R} \times \mathbb{R}^d$, is **Lipschitz continuous with respect to the state variable**, uniformly in t, if there exists $L \geq 0$ such that

$$\|f(x, t) - f(z, t)\| \leq L\|x - z\|, \qquad \forall t \in I, \quad \forall x, z \in \Omega.$$

If such a constant exists on the whole domain, f is globally Lipschitz in the state variable. If every point has a neighborhood on which such an estimate holds (equivalently, one has a suitable constant on each compact subset contained in the domain), f is locally Lipschitz in the state variable.

Local Lipschitz continuity with respect to the state variable, together with continuity in t, is a standard sufficient condition for local uniqueness. It is sufficient, not necessary.

6.5.3.2 Grönwall's Lemma

Lemma 6.2 — Grönwall Inequality. Let $u : [t_0, T] \to [0, \infty)$ be continuous and suppose

$$u(t) \leq A + \int_{t_0}^{t} b(s)u(s)\,\mathrm{d}s, \qquad A \geq 0,$$

where $b : [t_0, T] \to [0, \infty)$ is integrable. Then

$$\boxed{u(t) \leq A \exp\left(\int_{t_0}^{t} b(s)\,\mathrm{d}s\right)}.$$

Proof

Define

$$v(t) = A + \int_{t_0}^{t} b(s)u(s)\,\mathrm{d}s.$$

Then $u(t) \leq v(t)$ and v is absolutely continuous. For almost every t,

$$v'(t) = b(t)u(t) \le b(t)v(t).$$

Let

$$B(t) = \int_{t_0}^{t} b(s)\,\mathrm{d}s.$$

Multiplying the differential inequality by the integrating factor $e^{-B(t)}$ gives, for almost every t,

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(e^{-B(t)}v(t)\right) = e^{-B(t)}\big(v'(t) - b(t)v(t)\big) \le 0.$$

Hence $e^{-B(t)}v(t) \le v(t_0) = A$, and therefore

$$v(t) \le Ae^{B(t)}.$$

Since $u(t) \le v(t)$, the claimed estimate follows. □

A useful special case is $b(s) = L$, which gives

$$u(t) \le Ae^{L(t-t_0)}.$$

The case $A = 0$ is especially important: then the estimate forces $u(t) = 0$ identically. This is the mechanism behind the uniqueness proof below.

A slightly more general form, useful for perturbation estimates, is

$$u(t) \le a(t) + \int_{t_0}^{t} b(s)u(s)\,\mathrm{d}s,$$

where a is non-negative and non-decreasing. The same argument yields

$$u(t) \le a(t)\exp\left(\int_{t_0}^{t} b(s)\,\mathrm{d}s\right).$$

6.5.4 Existence and Uniqueness

6.5.4.1 Picard–Lindelöf Theorem

The **Picard-Lindelöf theorem** (also called the **Cauchy-Lipschitz theorem**) strengthens Peano's theorem by guaranteeing **both existence and uniqueness** of solutions.

Theorem 6.5.3 — Picard–Lindelöf. Let $f(z, t)$ be continuous on the **cylinder**:

$$D = \{(t, z) \in \mathbb{R} \times \mathbb{R}^d \mid |t - t_0| \le a, \|z - z_0\| \le b\}$$

Because D is compact and f is continuous, it is automatically bounded; define

$$M = \max_{(t,z)\in D} \|f(z, t)\|.$$

Assume in addition that f satisfies a **Lipschitz condition with respect to the state variable** with constant L:

$$\|f(x, t) - f(z, t)\| \le L\|x - z\|, \quad \forall (t, x), (t, z) \in D$$

Then, the initial value problem:

$$\begin{cases} z'(t) = f(z(t), t) \\ z(t_0) = z_0 \end{cases}$$

has a **unique solution** on the interval $I = [t_0 - T, t_0 + T]$, where:

$$T = \min\left(a, \frac{b}{M}\right)$$

with the convention $b/M = +\infty$ if $M = 0$.

Proof by Picard Iteration

It is enough to prove the result on the forward interval

$$I_+ = [t_0, t_0 + T].$$

The proof on $[t_0 - T, t_0]$ is identical after reversing time, and uniqueness allows the two solutions to be joined. In what follows write $I = I_+$.

Start from the constant function

$$z^{(0)}(t) = z_0.$$

Define recursively the **Picard iterates**

$$\boxed{z^{(n+1)}(t) = z_0 + \int_{t_0}^{t} f\big(s, z^{(n)}(s)\big)\,\mathrm{d}s}.$$

We prove existence in four steps.

1. Every Picard iterate remains inside the cylinder. If $z^{(n)}(t)$ belongs to the state ball $\|z - z_0\| \le b$, then

$$\begin{aligned}\|z^{(n+1)}(t) - z_0\| &\le \int_{t_0}^{t} \|f(z^{(n)}(s), s)\|\,\mathrm{d}s \\ &\le M|t - t_0| \le MT \le b.\end{aligned}$$

Since $z^{(0)}$ is in the ball, induction shows that every iterate is well defined on all of I and remains in the cylinder D.

2. The sequence is uniformly Cauchy. The first difference satisfies

$$\|z^{(1)}(t) - z^{(0)}(t)\| \le M|t - t_0|.$$

Using the Lipschitz condition and induction, one obtains

$$\boxed{\|z^{(n+1)}(t) - z^{(n)}(t)\| \le ML^n \frac{|t - t_0|^{n+1}}{(n+1)!}}.$$

Indeed, assuming the estimate at level $n - 1$,

$$\begin{aligned}\|z^{(n+1)}(t) - z^{(n)}(t)\| &\le L \int_{t_0}^{t} \|z^{(n)}(s) - z^{(n-1)}(s)\|\,\mathrm{d}s \\ &\le ML^n \frac{|t - t_0|^{n+1}}{(n+1)!}.\end{aligned}$$

Therefore, uniformly for $t \in I$,

$$\|z^{(n+1)}(t) - z^{(n)}(t)\| \le ML^n \frac{T^{n+1}}{(n+1)!}.$$

The scalar series

$$\sum_{n=0}^{\infty} ML^n \frac{T^{n+1}}{(n+1)!}$$

converges. By the Weierstrass test, $z^{(n)}$ converges uniformly on I to a continuous function z. Notice that this factorial estimate proves convergence without requiring $LT < 1$.

3. The limit satisfies the integral equation. Because f is Lipschitz in the state variable,

$$\|f(z^{(n)}(t), t) - f(z(t), t)\| \le L\|z^{(n)}(t) - z(t)\|,$$

so the integrands also converge uniformly. Passing to the limit in the Picard equation gives

$$z(t) = z_0 + \int_{t_0}^{t} f(z(s), s)\,\mathrm{d}s.$$

Since the integrand is continuous, the fundamental theorem of calculus implies

$$z'(t) = f(z(t), t), \qquad z(t_0) = z_0.$$

Thus a solution exists.

4. The solution is unique. Let y and z be two solutions with the same initial value. Then

$$\|y(t) - z(t)\| \le L \int_{t_0}^{t} \|y(s) - z(s)\|\,\mathrm{d}s$$

for $t \ge t_0$; the backward-time argument is identical after reversing the integration limits. Grönwall's lemma with $A = 0$ yields

$$y(t) = z(t)$$

throughout I. Hence the solution is unique. □

> **Remark.** A shorter proof can be obtained from Banach's fixed-point theorem on a sufficiently short interval satisfying $LT < 1$. The factorial estimate above is slightly stronger pedagogically because it shows directly why the Picard sequence converges on the whole interval allowed by the cylinder bound $MT \le b$.

6.5.4.2 Why the Lipschitz Condition Gives Uniqueness

Suppose y and z are two solutions with the same initial value. Their integral equations give

$$\|y(t) - z(t)\| \le \int_{t_0}^{t} \|f(y(s), s) - f(z(s), s)\|\,\mathrm{d}s \le L \int_{t_0}^{t} \|y(s) - z(s)\|\,\mathrm{d}s.$$

Grönwall's inequality with $A = 0$ yields $\|y(t) - z(t)\| = 0$. Hence $y(t) = z(t)$ throughout their common interval of existence.

6.5.4.3 Continuous Dependence on Initial Data

If $y(t_0) = y_0$ and $z(t_0) = z_0$, the same argument gives

$$\boxed{\|y(t) - z(t)\| \le e^{L|t-t_0|}\|y_0 - z_0\|.}$$

Thus the same hypothesis that gives uniqueness also yields a quantitative form of continuous dependence and therefore local well-posedness.

6.5.5 Counterexample: Existence Without Uniqueness

Consider the ordinary differential equation (ODE):

$$y' = \sqrt{|y|}$$

with the initial condition:

$$y(0) = 0.$$

6.5.5.1 Existence of the Solution

Peano's Theorem guarantees that if a function $f(y,t)$ is continuous, then there exists at least one solution within some interval around the initial point. In this case, the function $f(y) = \sqrt{|y|}$ is continuous for all values of y, so Peano's theorem tells us that there exists **at least one solution** for the given ODE, at least in a small interval around $t = 0$.

Thus, we can assert that there exists a function $y(t)$ that satisfies the differential equation and the initial condition $y(0) = 0$.

6.5.5.2 Failure of Uniqueness

Picard-Lindelöf Theorem 6.5.3 states that, in addition to existence, a solution is also **unique** if the function $f(y,t)$ is **Lipschitz continuous** with respect to y in an interval around the initial condition y_0. However, in our case, the function $f(y) = \sqrt{|y|}$ is not Lipschitz continuous at $y = 0$.

To verify this, let's compute the derivative of $f(y) = \sqrt{|y|}$:

$$f'(y) = \begin{cases} \dfrac{1}{2\sqrt{y}} y > 0 \\ -\dfrac{1}{2\sqrt{-y}} y < 0 \end{cases} = \frac{\operatorname{sign}(y)}{2\sqrt{|y|}}, \qquad y \neq 0$$

Notice that this derivative becomes infinite as $y \to 0$, so the function is not Lipschitz continuous at $y = 0$. A bounded derivative on a neighborhood would imply Lipschitz continuity by the mean-value theorem. Here one can prove failure directly: for $y > 0$, $|f(y) - f(0)|/|y| = 1/\sqrt{y} \to \infty$.

Since $f(y) = \sqrt{|y|}$ does not satisfy the Lipschitz condition at $y = 0$, **Picard-Lindelöf** Theorem 6.5.3 does not guarantee the uniqueness of the solution. This means there may be more than one solution that satisfies the same initial condition $y(0) = 0$.

6.5.5.3 Multiple Solutions

Even though a solution exists, it is not unique. In fact, there are **many solutions** that satisfy the ODE and the initial condition $y(0) = 0$. Here are a few examples:

1. **Trivial (Zero) Solution**
   One solution is the **zero solution**, where $y(x) = 0$ for all x. This is a solution that satisfies both the ODE and the initial condition $y(0) = 0$, and it holds for every value of x.
   $$y(x) = 0 \quad \text{for all} \quad x.$$
2. **Explicit (Non-Zero) Solution**
   Another solution can be found by solving the ODE by separating the variables. We start by writing the ODE as:
   $$\frac{\mathrm{d}y}{\sqrt{|y|}} = \mathrm{d}x.$$
   Integrating both sides:
   $$2\sqrt{|y|} = x + C.$$
   Setting the initial condition $y(0) = 0$, we get $C = 0$. Therefore, the solution becomes:
   $$y(x) = \frac{x^2}{4}, \quad \text{for} \quad x \geq 0.$$

This is an explicit solution for $x \geq 0$ and satisfies the ODE and the initial condition. For $x < 0$, the solution must be:

$$y(x) = -\frac{x^2}{4}, \quad \text{for} \quad x < 0,$$

to match the sign of the right-hand side. Thus, the complete solution is:

$$y(x) = \frac{x^2}{4} \begin{cases} 1 & x \geq 0, \\ -1 & x < 0. \end{cases}$$

This can also be written more compactly as:

$$y(x) = \frac{x|x|}{4}.$$

The derivative of this function is:

$$y'(x) = \frac{x}{2} \begin{cases} 1 & x \geq 0, \\ -1 & x < 0. \end{cases}$$

Which simplifies to:

$$y'(x) = \frac{|x|}{2}.$$

Notice that:

$$\sqrt{|y(x)|} = \sqrt{\left|\frac{x|x|}{4}\right|} = \sqrt{\frac{|x|^2}{4}} = \frac{|x|}{2} = y'(x).$$

Thus, the function $y(x) = \frac{x|x|}{4}$ satisfies the ODE, and it is indeed a valid solution.

6.5.5.4 A Whole Family of Solutions

The non-uniqueness is stronger than the two examples above. For arbitrary numbers $a \leq 0 \leq b$,

$$y_{a,b}(t) = \begin{cases} -\frac{(t-a)^2}{4}, & t < a, \\ 0, & a \leq t \leq b, \\ \frac{(t-b)^2}{4}, & t > b \end{cases}$$

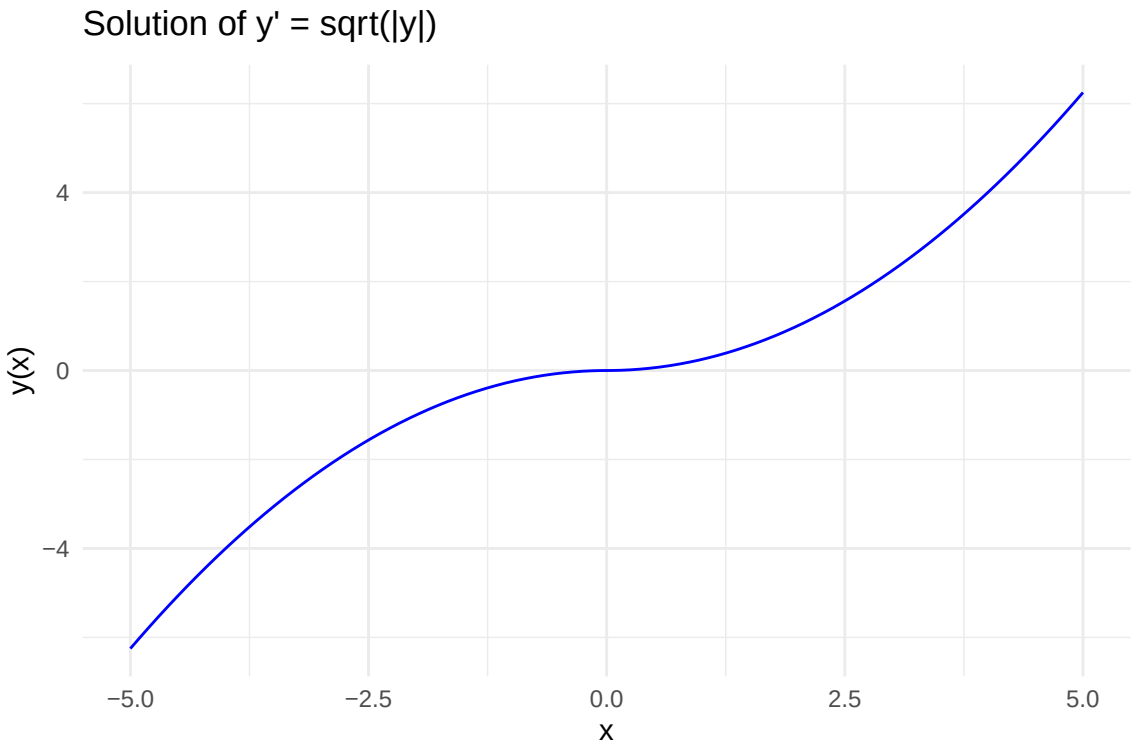
is continuously differentiable, satisfies $y_{a,b}(0) = 0$, and obeys $y'_{a,b} = \sqrt{|y_{a,b}|}$. The solution may therefore remain at the equilibrium for an arbitrary waiting time before leaving it. This explicitly proves non-uniqueness; failure of the Lipschitz hypothesis alone would only tell us that Picard–Lindelöf cannot be used.

6.5.5.5 Conclusion

The ODE $y' = \sqrt{|y|}$ with the initial condition $y(0) = 0$ has multiple solutions. In addition to the trivial solution $y(x) = 0$, we also have non-zero solutions of the form:

$$y(x) = \frac{x|x|}{4}.$$

These multiple solutions are possible due to the lack of uniqueness in this case, as the function $f(y) = \sqrt{|y|}$ is not Lipschitz continuous at $y = 0$.



Summary

- An initial value problem is **well posed** when a solution exists, is unique, and depends continuously on the data; the three requirements are independent and each can fail separately.
- **Peano's theorem** gives existence from continuity alone, but not uniqueness: $y' = \sqrt{|y|}$ with $y(0) = 0$ has infinitely many solutions.
- A **Lipschitz** condition in y restores uniqueness. Picard–Lindelöf turns the IVP into a fixed-point problem for the Volterra integral operator and gives local existence and uniqueness by contraction.
- **Grönwall's lemma** is the workhorse: it converts an integral inequality into an exponential bound, and it yields both uniqueness and continuous dependence on the initial data, with the amplification factor $e^{L(t-t_0)}$ that will reappear in every convergence estimate of the numerical chapters.
- A local solution extends until it leaves every compact subset of the domain; a global Lipschitz condition rules out blow-up in finite time.
- For autonomous systems, linearization at an equilibrium reduces the local question to the eigenvalues of the Jacobian — the same eigenvalues that govern stiffness and the choice of step size later on.

III DAE

7 Linear DAEs 93
7.1 Linear Differential-Algebraic Equations

8 Index Reduction for Linear DAEs 103
8.1 Separating the Algebraic Equations
8.2 The Reduction Algorithm
8.3 Hidden Constraints as Invariants
8.4 Examples

9 The Differential Index 123
9.1 DAE Index

10 Semi-Explicit Form and Its Classification 129
10.1 From an Implicit DAE to Semi-Explicit Form
10.2 Classification of Semi-Explicit DAEs
10.3 Can a Runge–Kutta Method Be Applied Directly?
10.4 Examples

11 General Index Reduction 137
11.1 The General Form $E(x, t)\dot{x} = G(x, t)$
11.2 Separating the Algebraic Part with a Generalized LU
11.3 Nonlinear Index Reduction
11.4 Alternative Constructions and Which to Use
11.5 Examples

12 Dummy Variables 155
12.1 Dummy-Variable Reduction
12.2 Examples

13 DAEs from Lagrangian Mechanics 169
13.1 Lagrangian DAEs
13.2 Numerical Integration

Differential-algebraic equations (DAEs) combine differential dynamics with algebraic constraints. They arise naturally in constrained mechanical systems, electrical circuits, chemical processes, and optimal control.

Prerequisites, objectives, and expected outcomes

Prerequisites: ODE initial-value problems, multivariable calculus, and basic matrix factorizations (Appendix A).
Objectives: recognize DAE structure, determine and reduce index, construct consistent initial data, and select a constraint-aware numerical method.
Expected outcomes: the reader can distinguish differential and algebraic variables, diagnose hidden constraints and drift, perform a convergence study, and justify a solver choice from index, stiffness, and geometry.

The part follows a single line of argument. The linear constant-coefficient case is treated first, because there the whole theory is available in closed form; the reduction is then rebuilt with Gaussian elimination, which is what a computer can actually do; the index is defined in general; any DAE is brought to semi-explicit form and classified; the reduction is extended to an arbitrary state-dependent system; and dummy variables are introduced to keep its cost bounded.

Contents

1. Linear DAEs — matrix pencils, the Weierstrass canonical form, the Kronecker index.
2. Index Reduction for Linear DAEs — separating the algebraic equations by elimination, differentiating them, and filing them as invariants of the resulting ODE.
3. The Differential Index — the general definition and the other notions of index.
4. Semi-Explicit Form and Its Classification — the transformation $\dot{\boldsymbol{x}} = \boldsymbol{y}$, $\mathbf{0} = \boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y}, t)$, the index-1, index-2 and index-3 conditions, Hessenberg form.
5. General Index Reduction — the form $\boldsymbol{E}(\boldsymbol{x}, t)\dot{\boldsymbol{x}} = \boldsymbol{G}(\boldsymbol{x}, t)$, the generalized LU that separates the algebraic part, and the reduction algorithm.
6. Dummy Variables — keeping the size of the reduced system bounded.
7. DAEs from Lagrangian Mechanics — where index-three systems come from.
8. Numerical Methods for DAEs — consistent initialization, drift, stabilization, projection, order reduction.
9. Case Studies and Exercises, in the companion volume.

7. Linear DAEs

This chapter introduces linear differential-algebraic equations, the matrix pencil that governs them, and the canonical form that makes their solution explicit.

We consider linear implicit differential equations of the form

$$E\dot{x}(t) = Ax(t) + f(t), \tag{7.1}$$

where $E, A \in \mathbb{R}^{n\times n}$ are constant matrices and $f : \mathbb{R} \to \mathbb{R}^n$ is a prescribed forcing term. Such equations arise from linearized nonlinear models, electrical circuits, constrained mechanical systems, and systems with algebraic conservation laws.

The treatment here is theoretical: it answers *when* a solution exists and *how many* initial conditions may be prescribed, through the Weierstrass canonical form and the Kronecker index. The computational counterpart — how to obtain the same information without computing a canonical form — is the subject of Chapter 8.

7.1 Linear Differential-Algebraic Equations

7.1.1 Why Differential-Algebraic Equations?

A differential-algebraic equation (DAE) has the implicit form

$$F(x, \dot{x}, t) = 0.$$

Unlike an explicit ODE, it need not determine every component of $\dot{x}$ directly. This is useful when physical constraints are simpler to express than to eliminate. In multibody dynamics, each body has its own equations of motion, while joints impose geometric constraints. In electrical circuits, element constitutive equations are supplemented by Kirchhoff current and voltage laws. The present chapter focuses on *constant-coefficient linear* DAEs; the nonlinear examples below motivate the subject.

7.1.1.1 Pendulum: minimal coordinates versus constrained coordinates

For a mass m on a rigid rod of length ℓ, with y positive upwards, minimal coordinates give

$$\begin{cases} \dot{\theta} = \omega, \\ \dot{\omega} = -\dfrac{g}{\ell} \sin\theta. \end{cases}$$

Cartesian coordinates with a Lagrange multiplier instead give

$$\begin{cases} \dot{x} = u, \\ \dot{y} = v, \\ m\dot{u} = \lambda x, \\ m\dot{v} = -mg + \lambda y, \\ 0 = x^2 + y^2 - \ell^2. \end{cases}$$

The multiplier λ represents the radial constraint force per unit distance. This system is a *nonlinear* DAE, not an instance of the linear pencil studied below. Differentiating its position constraint produces the hidden velocity constraint $xu+yv = 0$, followed by $u^2+v^2+x\dot{u}+y\dot{v} = 0$.

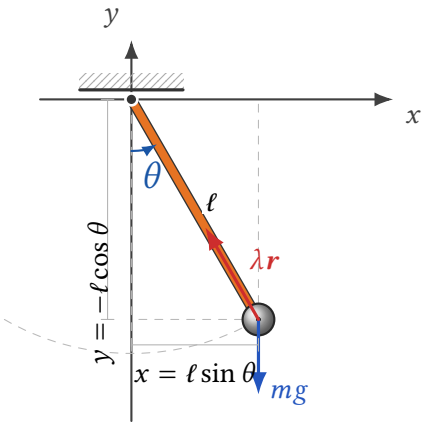


Figure 7.1: Pendulum in Cartesian coordinates: geometry, gravitational force, and constraint force.

7.1.1.2 Electrical circuits

For a series RLC circuit driven by a voltage source $e(t)$, eliminating the resistor and inductor voltages gives the minimal-coordinate ODE

$$\begin{cases} \dot{v}_C = \dfrac{i}{C}, \\ \dot{i} = \dfrac{e(t) - Ri - v_C}{L}. \end{cases}$$

If the branch voltages v_R and v_L are retained as unknowns, the same circuit is described by the DAE

$$\begin{cases} C\dot{v}_C - i = 0, \\ L\dot{i} - v_L = 0, \\ v_R - Ri = 0, \\ v_R + v_C + v_L - e(t) = 0. \end{cases}$$

The last two equations are algebraic: they encode the resistor constitutive law and Kirchhoff's voltage law. More generally, Kirchhoff's current law enforces $\sum_j i_j = 0$ at each node.

Retaining branch currents and node voltages usually gives a sparse DAE, whereas eliminating all constraints first can produce a more complicated ODE.

7.1.2 Three fundamental cases

Consider $E\dot{x} = Ax + f(t)$ with square constant matrices.

Structure of E	Interpretation	Initial data
$\det E \neq 0$	Explicit ODE after multiplying by E^{-1}	All components of $x(0)$ are free.
$E = 0$	Algebraic system $Ax = -f$	If A is invertible, $x(0)$ is fixed by $f(0)$.
E singular and nonzero	Genuine mixed DAE, possibly with hidden constraints	Consistency and uniqueness depend on the pencil.

Singularity of E does **not** imply singularity of the pencil $A - \lambda E$.
A primary consideration is the existence and uniqueness of a solution

$$x(t) : \mathbb{R} \to \mathbb{R}^n$$

The intriguing case arises when E is singular (i.e., not invertible). In contrast, if E is non-singular, we can simply invert it, allowing us to express the DAEs as ordinary differential equations (ODEs) as follows:

$$\dot{x}(t) = E^{-1}Ax(t) + E^{-1}f(t). \tag{7.2}$$

For the equation in Equation 7.2, it suffices to assume that $f(t)$ is piecewise continuous[1], which guarantees the existence and uniqueness of solutions. For a detailed discussion, see, for instance, [8].

Conversely, when the matrix E is singular, the existence of a solution becomes uncertain. A singular matrix E indicates that ODEs are coupled with linear algebraic constraints. Therefore, it is essential to consider certain regularity conditions for the pair (E, A), as illustrated in the following example.

Introductory Example

Consider the differential algebraic equations (DAEs) with the state vect

$$x(t) = (x_1(t); x_2(t)) \in \mathbb{R}^2$$

and the external perturbation $f(t) = 0$ for all $t \in [0, \infty)$. The matrices E and A are defined as follows:

$$E = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

This DAE corresponds to the scalar differential equation:

$$\dot{x}_1(t) = x_2(t).$$

[1] Weaker conditions can be applied to $f(t)$; for instance, it is sufficient for $f(t)$ to be Lebesgue measurable.

Assume $\boldsymbol{x}(0) = (0,0)^T$. For any continuously differentiable function φ satisfying $\varphi(0) = 0$, set

$$x_2(t) = \varphi(t), \qquad x_1(t) = \int_0^t \varphi(s)\,\mathrm{d}s.$$

For example, $\varphi(t) = \sin t$ gives $\boldsymbol{x}(t) = (1 - \cos t, \sin t)^T$, whereas $\varphi(t) = t$ gives $\boldsymbol{x}(t) = (t^2/2, t)^T$. Both satisfy the *same* initial condition and the DAE.
This illustrates that the solution is not unique, highlighting the need for certain regularity assumptions regarding the pair $(\boldsymbol{E}, \boldsymbol{A})$.

To establish the existence and uniqueness of a solution for Equation 7.1, we need to introduce the following definition.

Definition 7.1.1 — Matrix Pencils. A matrix pair $(\boldsymbol{E}, \boldsymbol{A})$ is defined as a *regular pencil* if there exists a scalar $\lambda_0 \in \mathbb{R}$ such that:

$$\left|\boldsymbol{A} - \lambda_0 \boldsymbol{E}\right| \neq 0.$$

Regular-Pencil Example

Consider the following pair of matrices $\boldsymbol{E}, \boldsymbol{A} \in \mathbb{R}^{3\times 3}$, which represents a non-regular (singular) pencil:

$$\boldsymbol{E} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 0 & 0 \\ 0 & -2 & 0 \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

For this pair, the determinant of the pencil evaluates to zero for every value of $\lambda_0 \in \mathbb{R}$:

$$\left|\boldsymbol{A} - \lambda_0 \boldsymbol{E}\right| = \begin{vmatrix} 1 - \lambda_0 & 0 & 0 \\ 2\lambda_0 & 1 & 0 \\ 0 & 2\lambda_0 & 0 \end{vmatrix} = 0.$$

The central idea is that if the pair $(\boldsymbol{E}, \boldsymbol{A})$ is a regular pencil, then we can simplify the problem through an appropriate change of coordinates. This concept is further clarified by the following theorem.

Theorem 7.1.1 — Weierstrass Form. A matrix pair $\boldsymbol{E}, \boldsymbol{A} \in \mathbb{R}^{n\times n}$ is classified as a *regular pencil* if, and only if, there exist invertible matrices

$$\boldsymbol{P} \in \mathbb{R}^{n\times n} \quad \text{and} \quad \boldsymbol{Q} \in \mathbb{R}^{n\times n}$$

such that:

$$(\boldsymbol{PEQ}, \boldsymbol{PAQ}) = \left(\begin{bmatrix} \boldsymbol{N} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix}, \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & \boldsymbol{J} \end{bmatrix} \right), \tag{7.3}$$

where $\boldsymbol{N}$ is an upper triangular nilpotent matrix, $\boldsymbol{J}$ is a matrix in Jordan canonical form, and $\boldsymbol{I}$ is the identity matrix of appropriate dimensions.

To enhance understanding, we will briefly recall the definition of a nilpotent matrix.

Definition 7.1.2 — Nilpotent Block. A square matrix $\boldsymbol{N}$ is termed *nilpotent* if there exists an integer $\nu \in \mathbb{N}$ such that:

$$\boldsymbol{N}^{\nu} = 0.$$

R It is important to note that the smallest integer ν for which $\boldsymbol{N}^{\nu} = 0$ is referred to as the *degree* of the nilpotent matrix $\boldsymbol{N}$. Furthermore, any upper triangular matrix that has zeros along its main diagonal qualifies as a nilpotent matrix.

Canonical-Form Example

Consider the matrix $\boldsymbol{N} \in \mathbb{R}^{3\times 3}$ defined as follows:

$$\boldsymbol{N} = \begin{bmatrix} 0 & 1 & 3 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}.$$

This matrix is nilpotent with a **degree** $\nu = 3$. To illustrate this, we compute the following powers of $\boldsymbol{N}$:

$$\boldsymbol{N}^2 = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{N}^3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The decoupling form presented in Equation 7.3 is known as the *Weierstrass canonical form*. For further details, refer to [9]. The matrices $\boldsymbol{N}$ and $\boldsymbol{J}$ in this form possess the following structures:

$$\boldsymbol{N} = \begin{bmatrix} \boldsymbol{N}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{N}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{N}_{n_1} \end{bmatrix}, \quad \boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{J}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{J}_{n_2}. \end{bmatrix}$$

Here, $\boldsymbol{N}_i$ for $i = 1, \dots, n_1$ and $\boldsymbol{J}_i$ for $i = 1, \dots, n_2$ are Jordan blocks. The blocks $\boldsymbol{N}_i$ are associated with the zero eigenvalues of the matrix $\boldsymbol{E}$, and are structured as follows:

$$\boldsymbol{N}_i = \begin{bmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & 1 \\ & & & & 0 \end{bmatrix}, \quad i = 1, \dots, n_1.$$

The dimension of each block $\boldsymbol{N}_i$ corresponds to the difference between the algebraic and geometric multiplicities of the respective eigenvalue. Conversely, the blocks $\boldsymbol{J}_i$ for $i = 1, \dots, n_2$ are classical Jordan blocks, structured as follows:

$$\boldsymbol{J}_k = \begin{bmatrix} \lambda_k & 1 & & & \\ & \lambda_k & 1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & 1 \\ & & & & \lambda_k \end{bmatrix}, \quad k = 1, \dots, n_2.$$

Next, we provide a constructive proof of the Weierstrass canonical form. This proof is based on a sequential multiplication of non-singular matrices. We will demonstrate only the sufficiency part; for the necessity, please refer to [9].

Proof. Let us consider the regular pencil pair $(\boldsymbol{E}, \boldsymbol{A})$. We can express this as:

$$(\boldsymbol{E}, \boldsymbol{A} + \lambda_0 \boldsymbol{E} - \lambda_0 \boldsymbol{E}).$$

Now, define the matrix $\boldsymbol{T}_0 = (\boldsymbol{A} - \lambda_0 \boldsymbol{E})$, which is non-singular due to the regular pencil assumption. This leads us to consider the expression:

$$\boldsymbol{T}_0^{-1}(\boldsymbol{E}, \boldsymbol{A} + \lambda_0 \boldsymbol{E} - \lambda_0 \boldsymbol{E}) = \left(\boldsymbol{T}_0^{-1}\boldsymbol{E}, \boldsymbol{I} + \lambda_0 \boldsymbol{T}_0^{-1}\boldsymbol{E}\right).$$

Next, let $\boldsymbol{T}_1$ be the non-singular transformation matrix such that $\boldsymbol{T}_1^{-1}\boldsymbol{T}_0^{-1}\boldsymbol{E}\boldsymbol{T}_1$ is in Jordan canonical form. Thus, the pair

$$\boldsymbol{T}_1^{-1}\left(\boldsymbol{T}_0^{-1}\boldsymbol{E}, \boldsymbol{I} + \lambda_0 \boldsymbol{T}_0^{-1}\boldsymbol{E}\right)\boldsymbol{T}_1$$

takes the following structure:

$$\left(\begin{bmatrix} \tilde{\boldsymbol{N}} & 0 \\ 0 & \tilde{\boldsymbol{J}} \end{bmatrix}, \begin{bmatrix} \boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}} & 0 \\ 0 & \boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{J}} \end{bmatrix}\right). \tag{7.4}$$

Here, the block $\tilde{\boldsymbol{N}}$ corresponds to the zero eigenvalues of the singular matrix $\boldsymbol{T}_0^{-1}\boldsymbol{E}$. Therefore, $\tilde{\boldsymbol{N}}$ is upper triangular, with zeros on the main diagonal. As a result, the matrix $(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})$ is invertible.

Next, we define the non-singular transformation matrix $\boldsymbol{T}_2$ as follows:

$$\boldsymbol{T}_2 = \begin{bmatrix} (\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}}) & 0 \\ 0 & \tilde{\boldsymbol{J}} \end{bmatrix}.$$

Multiplying the expression in Equation 7.4 by $\boldsymbol{T}_2^{-1}$ from the left, we obtain the following structure:

$$\boldsymbol{S} = \left(\begin{bmatrix} (\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}\tilde{\boldsymbol{N}} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix}, \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & \tilde{\boldsymbol{J}}^{-1}(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{J}}) \end{bmatrix}\right). \tag{7.5}$$

Finally, we need to consider matrices $\boldsymbol{T}_{31}$ and $\boldsymbol{T}_{32}$ such that:

- $\boldsymbol{T}_{31}^{-1}(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}\tilde{\boldsymbol{N}}\boldsymbol{T}_{31}$ is in Jordan canonical form.
- $\boldsymbol{T}_{32}^{-1}\tilde{\boldsymbol{J}}^{-1}(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{J}})\boldsymbol{T}_{32}$ is also in Jordan canonical form.

We construct the transformation matrix $\boldsymbol{T}_3$ as follows:

$$\boldsymbol{T}_3 = \begin{bmatrix} \boldsymbol{T}_{31} & 0 \\ 0 & \boldsymbol{T}_{32} \end{bmatrix}.$$

Thus, from Equation 7.5 the final change of coordinates yields:

$$\boldsymbol{T}_3^{-1}\boldsymbol{S}\boldsymbol{T}_3 = \left(\begin{bmatrix} \boldsymbol{N} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix}, \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & \boldsymbol{J} \end{bmatrix}\right).$$

It remains to show that the matrix $\boldsymbol{N}$ is nilpotent. To do this, we need to demonstrate that the matrix

$$(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}\tilde{\boldsymbol{N}}$$

is nilpotent. First, observe that the matrices $(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})$ and $\tilde{\boldsymbol{N}}$ commute since:

$$(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})\tilde{\boldsymbol{N}} = \tilde{\boldsymbol{N}} + \lambda_0 \tilde{\boldsymbol{N}}^2 = \tilde{\boldsymbol{N}}(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}}).$$

Furthermore, $(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}$ also commutes with $\tilde{\boldsymbol{N}}$:

$$(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}\tilde{\boldsymbol{N}} = (\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}\tilde{\boldsymbol{N}}(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1}$$

$$= \tilde{\boldsymbol{N}}(\boldsymbol{I} + \lambda_0 \tilde{\boldsymbol{N}})^{-1},$$

where we have used the fact that

$$\tilde{\boldsymbol{N}}(\boldsymbol{I} + \lambda_0\tilde{\boldsymbol{N}}) = (\boldsymbol{I} + \lambda_0\tilde{\boldsymbol{N}})\tilde{\boldsymbol{N}}$$

Given that these matrices commute, let ν be the index such that $\tilde{\boldsymbol{N}}^{\nu} = 0$. The following holds:

$$\left[(\boldsymbol{I} + \lambda_0\tilde{\boldsymbol{N}})\tilde{\boldsymbol{N}}\right]^{\nu} = (\boldsymbol{I} + \lambda_0\tilde{\boldsymbol{N}})^{\nu}\tilde{\boldsymbol{N}}^{\nu} = 0.$$

This completes the proof.

In summary, the overall transformation matrices $(\boldsymbol{P}, \boldsymbol{Q})$ are given by:

$$\boldsymbol{P} = (\boldsymbol{T}_0\boldsymbol{T}_1\boldsymbol{T}_2\boldsymbol{T}_3)^{-1}, \quad \boldsymbol{Q} = \boldsymbol{T}_1\boldsymbol{T}_3.$$

■

7.1.3 Differential–Algebraic Decomposition

The Weierstrass canonical form is particularly valuable because it allows us to rewrite the system described by Equation 7.1 using the transformation matrices $(\boldsymbol{P}, \boldsymbol{Q})$ defined earlier. This results in the following expression:

$$\boldsymbol{P}\boldsymbol{E}\boldsymbol{Q}\boldsymbol{Q}^{-1}\dot{\boldsymbol{x}}(t) = \boldsymbol{P}\boldsymbol{A}\boldsymbol{Q}\boldsymbol{Q}^{-1}\boldsymbol{x}(t) + \boldsymbol{P}\boldsymbol{f}(t).$$

Next, we apply the following change of coordinates:

$$\begin{aligned} \boldsymbol{z}(t) &= \boldsymbol{Q}^{-1}\boldsymbol{x}(t), \\ \boldsymbol{g}(t) &= \boldsymbol{P}\boldsymbol{f}(t), \end{aligned} \tag{7.6}$$

leading to the equivalent differential-algebraic equations (DAEs):

$$\begin{bmatrix} \boldsymbol{N} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{z}}_1(t) \\ \dot{\boldsymbol{z}}_2(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{z}_1(t) \\ \boldsymbol{z}_2(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{g}_1(t) \\ \boldsymbol{g}_2(t) \end{bmatrix}.$$

In this formulation, the vectors $\boldsymbol{z}(t)$ and $\boldsymbol{g}(t)$ are partitioned according to the dimensions of matrices $\boldsymbol{J}$ and $\boldsymbol{N}$. Specifically, we have:

$$\boldsymbol{z}(t) = \begin{pmatrix} \boldsymbol{z}_1(t) \\ \boldsymbol{z}_2(t) \end{pmatrix}, \quad \boldsymbol{g}(t) = \begin{pmatrix} \boldsymbol{g}_1(t) \\ \boldsymbol{g}_2(t) \end{pmatrix}.$$

Notably, the two sets of equations corresponding to $\boldsymbol{z}_1(t)$ and $\boldsymbol{z}_2(t)$ are decoupled, yielding:

$$\begin{aligned} \boldsymbol{N}\dot{\boldsymbol{z}}_1 &= \boldsymbol{z}_1 + \boldsymbol{g}_1(t), \\ \dot{\boldsymbol{z}}_2 &= \boldsymbol{J}\boldsymbol{z}_2 + \boldsymbol{g}_2(t). \end{aligned} \tag{7.7}$$

7.1.3.1 Differential Component

The equation associated with $\boldsymbol{z}_2(t)$ is straightforward; it represents an ordinary differential equation (ODE). Therefore, the unique solution given the initial condition $\boldsymbol{z}_2(0)$ is:

$$\boldsymbol{z}_2(t) = \mathrm{e}^{\boldsymbol{J}t}\boldsymbol{z}_2(0) + \int_0^t \mathrm{e}^{\boldsymbol{J}(t-\tau)}\boldsymbol{g}_2(\tau)\,\mathrm{d}\tau, \quad \forall t \geq 0.$$

where matrix exponential is

$$\exp \boldsymbol{A} = \boldsymbol{I} + \boldsymbol{A} + \frac{1}{2}\boldsymbol{A}^2 + \frac{1}{6}\boldsymbol{A}^3 + \cdots + \frac{1}{k!}\boldsymbol{A}^k + \cdots$$

For further details, refer to [11].

7.1.3.2 Algebraic Component

On the other hand, Equation 7.7 remains a DAE. However, since the matrix $\boldsymbol{N}$ is nilpotent, we can manipulate it effectively. To facilitate this, we can rewrite Equation 7.7 as:

$$z_1(t) = \boldsymbol{N}\dot{z}_1(t) - \boldsymbol{g}_1(t). \tag{7.8}$$

Substituting Equation 7.8 into itself and differentiating gives us:

$$\begin{aligned} z_1(t) &= \boldsymbol{N}\frac{\mathrm{d}}{\mathrm{d}t}\left[\boldsymbol{N}\dot{z}_1(t) - \boldsymbol{g}_1(t)\right] - \boldsymbol{g}_1(t) \\ &= \boldsymbol{N}^2\frac{\mathrm{d}^2 z_1(t)}{\mathrm{d}t^2} - \boldsymbol{N}\frac{\mathrm{d}\boldsymbol{g}_1(t)}{\mathrm{d}t} - \boldsymbol{g}_1(t) \\ &= \boldsymbol{N}^3\frac{\mathrm{d}^3 z_1(t)}{\mathrm{d}t^3} - \boldsymbol{N}^2\frac{\mathrm{d}^2\boldsymbol{g}_1(t)}{\mathrm{d}t^2} - \cdots \\ &\vdots \\ &= \boldsymbol{N}^\nu\frac{\mathrm{d}^\nu z_1(t)}{\mathrm{d}t^\nu} - \boldsymbol{N}^{\nu-1}\frac{\mathrm{d}^{\nu-1}\boldsymbol{g}_1(t)}{\mathrm{d}t^{\nu-1}} - \cdots - \boldsymbol{N}\frac{\mathrm{d}\boldsymbol{g}_1(t)}{\mathrm{d}t} - \boldsymbol{g}_1(t). \end{aligned}$$

This process establishes differential properties of the solutions. Note that the procedure terminates when we reach the index ν of matrix $\boldsymbol{N}$, since $\boldsymbol{N}^\nu = 0$. Consequently, we can express a unique solution as a finite summation of increasing derivatives of $\boldsymbol{g}(t)$:

$$z_1(t) = -\sum_{k=0}^{\nu-1} \boldsymbol{N}^k \frac{\mathrm{d}^k \boldsymbol{g}_1(t)}{\mathrm{d}t^k}.$$

R The nilpotency degree ν is often called the *Kronecker (differentiation) index* for a regular constant-coefficient DAE. The highest derivative of $\boldsymbol{g}_1$ appearing in the algebraic solution is of order $\nu - 1$, not ν. An invertible $\boldsymbol{E}$ corresponds to index zero (no nilpotent block).

With this understanding, we can now express the unique solution to the original problem stated in Equation 7.1. To do this, we simply invert the change of coordinates described in Equation 7.6. The solution is given by:

$$\boldsymbol{x}(t) = \boldsymbol{Q}\boldsymbol{z}(t) = \boldsymbol{Q}\begin{bmatrix} -\sum_{k=0}^{\nu-1} \boldsymbol{N}^k \frac{\mathrm{d}^k \boldsymbol{g}_1(t)}{\mathrm{d}t^k} \\ \mathrm{e}^{\boldsymbol{J}t}\boldsymbol{z}_2(0) + \int_0^t \mathrm{e}^{\boldsymbol{J}(t-\tau)}\boldsymbol{g}_2(\tau)\,\mathrm{d}\tau \end{bmatrix} \tag{7.9}$$

R It is crucial to observe that the initial condition $z_1(0)$ cannot be chosen arbitrarily; it must satisfy the following relationship:

$$z_1(0) = -\sum_{k=0}^{\nu-1} \boldsymbol{N}^k \frac{\mathrm{d}^k \boldsymbol{g}_1(0)}{\mathrm{d}t^k}.$$

As a result, the initial condition $\boldsymbol{x}_0 = \boldsymbol{x}(t=0)$ for Equation 7.1 is also not freely chosen. This reflects the requirement that the initial condition must be consistent with the constraints imposed by the matrix $\boldsymbol{E}$.

To summarize our discussion, we can outline the conditions for the existence of a solution to the problem stated in Equation 7.1:

Theorem 7.1.2 The problem in Equation 7.1 with initial condition $\boldsymbol{x}_0 = \boldsymbol{x}(t = 0)$ has a solution if the following conditions are satisfied:

- The differential forcing is locally integrable (piecewise continuity is sufficient), and the transformed algebraic forcing $\boldsymbol{g}_1$ has the derivatives through order $\nu - 1$ required by Equation 7.9 (with additional smoothness when a classical continuously differentiable solution is sought).
- The pair $(\boldsymbol{E}, \boldsymbol{A})$ forms a regular pencil.
- The initial condition $\boldsymbol{x}_0$ meets the requirement:

$$\boldsymbol{x}_0 = \boldsymbol{Q}\begin{bmatrix} -\sum_{k=0}^{\nu-1} \boldsymbol{N}^k \frac{\mathrm{d}^k \boldsymbol{g}_1(0)}{\mathrm{d}t^k} \\ \boldsymbol{z}_2(0) \end{bmatrix}$$

Furthermore, the corresponding solution is given by Equation 7.9.

7.1.4 Jordan form and the infinite eigenvalues of a pencil

Over $\mathbb{C}$, every square matrix $\boldsymbol{A}$ is similar to a block-diagonal Jordan matrix $\boldsymbol{J} = \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T}$. Its blocks are $\boldsymbol{J}_k(\lambda) = \lambda \boldsymbol{I}_k + \boldsymbol{N}_k$, where $\boldsymbol{N}_k$ has ones on the superdiagonal and $\boldsymbol{N}_k^k = \boldsymbol{0}$. The number of blocks associated with λ is its geometric multiplicity; a matrix is diagonalizable exactly when all its Jordan blocks have size one.

For a *regular* pencil, Weierstrass equivalence uses two matrices $\boldsymbol{P}, \boldsymbol{Q}$, rather than a similarity transformation. The Jordan block $\boldsymbol{J}$ describes finite generalized eigenvalues; the nilpotent block $\boldsymbol{N}$ describes eigenvalues at infinity. The largest nilpotent block size is the differentiation index. A regular pencil can therefore have singular $\boldsymbol{E}$ without loss of uniqueness for consistent initial data.

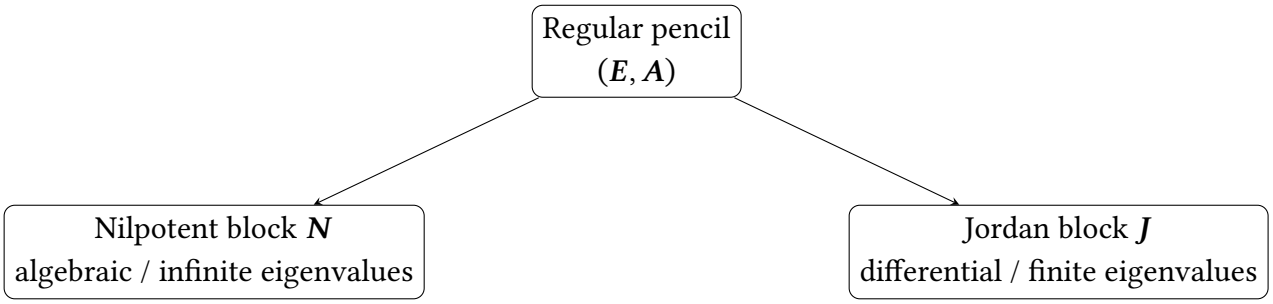


Figure 7.2: Weierstrass decomposition of a regular matrix pencil.

7.1.5 Worked examples: indices one, two, and three

Use the sign convention $\boldsymbol{N}\dot{\boldsymbol{z}} + \boldsymbol{z} = \boldsymbol{g}(t)$ in the following examples. This convention is different from the $\boldsymbol{N}\dot{\boldsymbol{z}} - \boldsymbol{z} = \boldsymbol{g}_1$ convention in Equation 7.8, so the alternating signs below are intentional:

$$\boldsymbol{z} = \sum_{k=0}^{\nu-1} (-1)^k \boldsymbol{N}^k \boldsymbol{g}^{(k)}.$$

Index 1. For $\boldsymbol{N} = [0]$, the constraint is $z = g$; no differentiation is required.

Index 2. For $\boldsymbol{N} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ and $\boldsymbol{g} = (t, t^2)^T$,

$$z = g - Ng' = (t - 2t, t^2)^T = (-t, t^2)^T.$$

Indeed $N\dot{z} + z = (2t - t, t^2)^T = g$.

Index 3. For $N = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ and $g = (0, 0, t^2)^T$,

$$z = g - Ng' + N^2 g'' = (2, -2t, t^2)^T.$$

Direct substitution verifies $N\dot{z} + z = g$. In each case the initial algebraic state is **fixed** by $g(0), \dots, g^{(\nu-1)}(0)$, whereas the initial state of any accompanying ODE block is free. Inconsistent algebraic initial values do not produce a classical solution satisfying both the equation and the prescribed initial condition.

Index	Nilpotency	Highest forcing derivative
0	No algebraic block	Not applicable
1	$N = 0$	0
2	$N^2 = 0$	1
3	$N^3 = 0$	2
ν	$N^\nu = 0$	$\nu - 1$

7.1.6 Further regular and singular pencils

For $E = \mathrm{diag}(1, 0)$ and $A = \mathrm{diag}(2, 1)$, $\det(A - \lambda E) = 2 - \lambda$: the pencil is regular although E is singular. The DAE $E\dot{x} = Ax$ gives $\dot{x}_1 = 2x_1$, $x_2 = 0$, so $x_2(0) = 0$ is mandatory. In contrast, $E = \mathrm{diag}(1, 0)$ and $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ have identically zero determinant and permit the arbitrary-function solutions derived above. Regularity is therefore the key structural hypothesis, not invertibility of E.

8. Index Reduction for Linear DAEs

The Weierstrass form of Chapter 7 is valuable theoretically but expensive to compute: it requires Jordan forms, which are not numerically stable. This chapter obtains the same structural information — the separation between differential and algebraic equations, the hidden constraints, and the index — using nothing but Gaussian elimination.

The procedure is a loop with three steps, repeated until the derivative coefficient matrix becomes nonsingular:

1. **separate** the algebraic equations, by row-reducing $M = [E \,|\, A \,|\, f]$ until the rows of E that vanish are grouped at the bottom;
2. **file** those algebraic equations as invariants, or hidden constraints, of the system that will result;
3. **differentiate** them, append the result to the differential block, and start again.

Each pass through the loop reduces the index by one. What the loop produces is an ODE together with a list of invariants, and the initial condition of that ODE is not free: it must satisfy every invariant collected along the way.

8.1 Separating the Algebraic Equations

We are now prepared to explore the utility of the Gauss-like method Theorem A.1.1 in determining the index of a Differential-Algebraic Equation (DAE) and transforming a generic linear DAE into an Ordinary Differential Equation (ODE).

8.1.1 DAE Formulation

Consider a generic linear DAE expressed in the following form:

$$E\boldsymbol{x}' + A\boldsymbol{x} = \boldsymbol{f}(t). \tag{8.1}$$

The matrices E and $A \in \mathbb{R}^{n\times n}$. Define the rank of E as:

$$\operatorname{rank} \boldsymbol{E} = r \leq n.$$

8.1.2 Change of Coordinates

Now, consider a pair of matrices $\boldsymbol{L}^{(0)} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{R}^{(0)} \in \mathbb{R}^{n\times n}$ such that:

$$\boldsymbol{L}^{(0)}\boldsymbol{E}\boldsymbol{R}^{(0)} = \boldsymbol{E}^{(0)} = \begin{bmatrix} \boldsymbol{\Lambda}^{(0)} & 0 \\ 0 & 0 \end{bmatrix}.$$

As in the previous section, the superscript $^{(0)}$ indicates the initial reduction stage; it does **not** by itself give the Kronecker index of the DAE. At this first iteration, $\boldsymbol{\Lambda}^{(0)} \in \mathbb{R}^{r\times r}$.

If $r = n$, we are done since the DAE has index $\nu = 0$, and the matrix $\boldsymbol{E}$ is full rank and invertible. If $r < n$, we introduce the following change of coordinates:

$$\boldsymbol{x} = \boldsymbol{R}^{(0)}\boldsymbol{x}^{(0)}, \qquad \boldsymbol{x}' = \boldsymbol{R}^{(0)}(\boldsymbol{x}^{(0)})',$$

and define the matrix $\boldsymbol{A}^{(0)}$ and vector $\boldsymbol{f}^{(0)}$ as follows:

$$\begin{aligned} \boldsymbol{A}^{(0)} &:= \boldsymbol{L}^{(0)}\boldsymbol{A}\boldsymbol{R}^{(0)}, \\ \boldsymbol{f}^{(0)} &:= \boldsymbol{L}^{(0)}\boldsymbol{f}. \end{aligned}$$

so that

$$\boldsymbol{E}^{(0)}(\boldsymbol{x}^{(0)})' + \boldsymbol{A}^{(0)}\boldsymbol{x}^{(0)} = \boldsymbol{f}^{(0)}(t). \tag{8.2}$$

obtained by multiply the left-hand side of Equation 8.1 by $\boldsymbol{L}^{(0)}$.

8.1.2.1 Matrix Structure

Notice that by partitioning

$$(\boldsymbol{x}')_1^{(0)}, \boldsymbol{x}_1^{(0)} \in \mathbb{R}^r, \quad (\boldsymbol{x}')_2^{(0)}, \boldsymbol{x}_2^{(0)} \in \mathbb{R}^{n-r}, \quad \text{and} \quad \boldsymbol{x}^{(0)} = \begin{bmatrix} \boldsymbol{x}_1^{(0)} \\ \boldsymbol{x}_2^{(0)} \end{bmatrix}.$$

then Equation 8.2 has the following special structure:

$$\begin{bmatrix} \boldsymbol{\Lambda}^{(0)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} (\boldsymbol{x}')_1^{(0)}(t) \\ (\boldsymbol{x}')_2^{(0)}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{A}_{11}^{(0)} & \boldsymbol{A}_{12}^{(0)} \\ \boldsymbol{A}_{21}^{(0)} & \boldsymbol{A}_{22}^{(0)} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1^{(0)} \\ \boldsymbol{x}_2^{(0)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_1^{(0)}(t) \\ \boldsymbol{f}_2^{(0)}(t) \end{bmatrix}. \tag{8.3}$$

Equation 8.3 can also be expressed as a pair of matrix equations:

$$\begin{aligned} \boldsymbol{\Lambda}^{(0)}(\boldsymbol{x}')_1^{(0)} + \boldsymbol{A}_{11}^{(0)}\boldsymbol{x}_1^{(0)} + \boldsymbol{A}_{12}^{(0)}\boldsymbol{x}_2^{(0)} &= \boldsymbol{f}_1^{(0)}(t), \qquad (a) \\ \boldsymbol{A}_{21}^{(0)}\boldsymbol{x}_1^{(0)} + \boldsymbol{A}_{22}^{(0)}\boldsymbol{x}_2^{(0)} &= \boldsymbol{f}_2^{(0)}(t), \qquad (b) \end{aligned} \tag{8.4}$$

8.1.2.2 Component Decoupling

Notice that

- Equation 8.4(a) is an ODE since $\boldsymbol{\Lambda}^{(0)}$ is invertible
- Equation 8.4(b) is a pure algebraic equation.

These equations are separated **by equation type**, but are not generally decoupled **by variable**: the blocks $\boldsymbol{A}_{12}^{(0)}$ and $\boldsymbol{A}_{21}^{(0)}$ still couple the two groups. Actual decoupling requires further reduction (or the Weierstrass form).

8.1.2.3 Constraint Differentiation

Now we differentiate Equation 8.4(b) with respect to time:

$$A_{21}^{(0)}(x')_1^{(0)} + A_{22}^{(0)}(x')_2^{(0)} = (f_2'(t))^{(0)}. \tag{8.5}$$

Writing Equation 8.5 with Equation 8.4(a) in matrix form yields:

$$\begin{bmatrix} \Lambda^{(0)} & \mathbf{0} \\ A_{21}^{(0)} & A_{22}^{(0)} \end{bmatrix} \begin{bmatrix} (x')_1^{(0)} \\ (x')_2^{(0)} \end{bmatrix} + \begin{bmatrix} A_{11}^{(0)} & A_{12}^{(0)} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \end{bmatrix} = \begin{bmatrix} v_1^{(0)} \\ (f_2'(t))^{(0)} \end{bmatrix}. \tag{8.6}$$

8.1.3 Rank Increase

As observed, the differentiation with respect to time of the algebraic constraints in the matrix representation produces merely a shift to the left of the two blocks $A_{21}^{(0)}$ and $A_{22}^{(0)}$ while preserving the original structure.

The rank of the derivative coefficient matrix **need not increase strictly** after differentiation. For example, if $A_{22}^{(0)} = \mathbf{0}$, the displayed derivative matrix remains rank-deficient. Further differentiation can reveal additional constraints, but the original equations must also be retained. For a regular pencil, the nilpotency degree in the Weierstrass form gives a rigorous reference definition of the index. Rank progression alone, without specifying the precise derivative array and its independent constraints, is not an index theorem.

Definition 8.1.1 For a **regular constant-coefficient pencil**, the Kronecker index is the largest size of a nilpotent Jordan block in its Weierstrass form (index zero if no nilpotent block is present). An iterative Gaussian method may recover this number only under explicitly stated regularity and rank assumptions; the dimension of a single evolving block is not, by itself, a general definition.

In this way, we are able to compute the index of a DAE systematically through a sequence of transformations, eventually leading to a formulation where the components are decoupled, thus facilitating the analysis and solution of the DAE.

8.2 The Reduction Algorithm

A Differential-Algebraic Equation (DAE) is a system of equations that consists of both differential and algebraic equations. Unlike ordinary differential equations (ODEs), DAEs may contain constraints that restrict the solution space. The **(differential) index** of a DAE measures how many times the system must be differentiated to convert it into an explicit ODE.

This tutorial explains how to compute the index of a linear DAE systematically, assuming constant coefficient matrices and avoiding the Kronecker canonical form.

8.2.1 Reduction Procedure

First recall that a linear DAE with constant coefficients is given by:

$$Ex' + Ax = f(t)$$

where:

- E and A are constant matrices,
- $x(t)$ is the state variable,

- $f(t)$ is an input function,
- x' is the time derivative of x.

The following steps provide a rough outline of the approach, offering a general understanding of the index reduction procedure.

Step 1: Check Singularity

Compute $\det(E)$:

- If $\det(E) \neq 0$, then E is invertible, and the system reduces to an ODE (**index 0**). Terminate procedure.
- If $\det(E) = 0$, then E is singular, indicating the presence of algebraic constraints. Proceed to the next step.

Step 2: Decompose the System

Determine the **rank** of $E \in \mathbb{R}^{n\times n}$, denoted as $r = \operatorname{rank}(E)$. Find:

- A **basis** $Q_a \in \mathbb{R}^{n\times(n-k)}$ for the **null space** $\mathcal{N}(E^T)$, which identifies the algebraic constraints.
  The basis satisfy $E^T Q_a = 0$ or transposing $Q_a^T E = 0$.
- A matrix $Q_d \in \mathbb{R}^{n\times k}$ that complete Q_a to a basis. In practice Q_d make the matrix $Q = (Q_d, Q_a)$ non singular.

Decomposing the system: $Ex' + Ax = f(t)$ by multiplying to the left with Q^T:

- **Differential part**: $E_d x' + A_d x = f_d(t)$ where

$$E_d = Q_d^T E, \quad A_d = Q_d^T A, \quad f_d(t) = Q_d^T f(t)$$

- **Algebraic part**: $A_a x = f_a(t)$ where

$$A_a = Q_a^T A, \qquad f_a(t) = Q_a^T f(t)$$

since $Q_a^T E = 0$.

R The decomposition used to separate differential and algebraic part seem quite cumbersome and difficult at first. However, in practice, it is straightforward to compute using standard tools from linear algebra. For instance, the LU decomposition can be applied to the matrix E to efficiently compute Q_A and Q_b. The detail will be discussed forward.

Step 3: Handle Constraints

The algebraic constraints from Step 2 are:

$$A_a x = f_a(t)).$$

Differentiate these constraints:

$$A_a x' = f_a'(t)$$

Now the DAE take the form

$$\begin{pmatrix} E_d \\ A_a \end{pmatrix} x' + \begin{pmatrix} A_d \\ 0 \end{pmatrix} x = \begin{pmatrix} f_d(t) \\ f_a'(t) \end{pmatrix}$$

or

$$\widetilde{E}x' + \widetilde{A}x = \widetilde{f}(t)$$

where

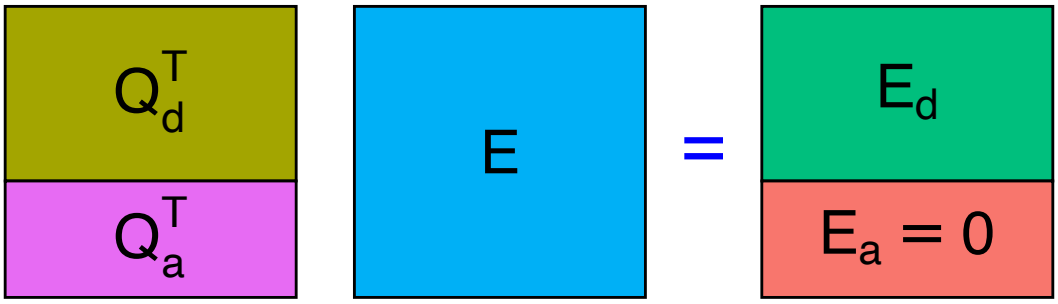


Figure 8.1: Effect of left multiplication by the transpose of the projector matrix.

$$\widetilde{E} = \begin{pmatrix} E_d \\ A_a \end{pmatrix} \quad \widetilde{A} = \begin{pmatrix} A_d \\ 0 \end{pmatrix} \quad \widetilde{f}(t) = \begin{pmatrix} f_d(t) \\ f_a'(t) \end{pmatrix}$$

The effect of multiply E by Q^T

Return to step 1. When termninate the procedure the index is the number of time Step 3 (differentiation) is performed.

8.2.2 Constructing the Subspaces

For the systematic construction of the matrices Q_a and Q_d, we leverage the (generalized) LU decomposition, which provides a structured way to separate the differential and algebraic components of a Differential-Algebraic Equation (DAE).

Theorem 8.2.1 See [10] and [23].

Let $A \in \mathbb{R}^{n\times m}$ of rank $k \leqslant \min(n, m)$, then there exists two permutation matrices P_L and P_R such that

$$P_L A P_R = \begin{pmatrix} L & 0 \\ M & I \end{pmatrix} \begin{pmatrix} U & N \\ 0 & 0 \end{pmatrix}$$

with

- $L \in \mathbb{R}^{k\times k}$ lower triangular with 1 on the main diagonal.
- $U \in \mathbb{R}^{k\times k}$ upper triangular with nonzero element on the main diagonal.
- $M \in \mathbb{R}^{n-k\times k}$
- $N \in \mathbb{R}^{k\times m-k}$

Using this theorem for E:

$$P_L\, E\, P_R = \begin{pmatrix} L & 0 \\ M & I \end{pmatrix} \begin{pmatrix} U & N \\ 0 & 0 \end{pmatrix}$$

and (using the fact that for a permutation matrix $P^{-1} = P^T$)

$$E = P_L^T \begin{pmatrix} L & 0 \\ M & I \end{pmatrix} \begin{pmatrix} U & N \\ 0 & 0 \end{pmatrix} P_R^T$$

so that defining

$$Q^T = \begin{pmatrix} L & 0 \\ M & I \end{pmatrix}^{-1} P_L = \begin{pmatrix} L^{-1} & 0 \\ -ML^{-1} & I \end{pmatrix} P_L$$

we have

$$Q^T E = \underbrace{\begin{pmatrix} L^{-1} & 0 \\ -ML^{-1} & I \end{pmatrix} P_L}_{Q^T} \quad \underbrace{P_L^T \begin{pmatrix} L & 0 \\ M & I \end{pmatrix} \begin{pmatrix} U & N \\ 0 & 0 \end{pmatrix} P_R^T}_{E}$$

$$= \begin{pmatrix} U & N \\ 0 & 0 \end{pmatrix} P_R^T$$

8.2.2.1 Simplification

we can discard the colum permutation P_R defining

$$\begin{pmatrix} E_d \\ 0 \end{pmatrix} = \begin{pmatrix} U & N \\ 0 & 0 \end{pmatrix} P_R^T$$

so that

$$E = P_L^T \begin{pmatrix} L & 0 \\ M & I \end{pmatrix} \begin{pmatrix} S \\ 0 \end{pmatrix}, \quad Q^T E = \begin{pmatrix} E_d \\ 0 \end{pmatrix},$$

and applying Q^T to the DAE $Ex' + Ax = f(t)$ results in

$$Q^T \left(E\,x' + A\,x \right) = Q^T f(t)$$

$$\begin{pmatrix} E_d \\ 0 \end{pmatrix} x' + Q^T A x = Q^T f(t)$$

and there is a practical separation of the algebraic equation from the DAE.

R [Matrix-Free Construction] The separation of differential and algebraic part is equivalent to the reduction to echelon form (see Appendix A). This method is a **systematic** approach to solving linear systems, computing ranks, and checking singularity conditions.

8.2.4 Rank-based algorithms: scope and verification

A practical constant-coefficient reduction proceeds by computing a rank-revealing factorization of E, applying the *same* row and state transformations to the entire pencil, identifying independent algebraic constraints, and differentiating them only as needed. After each step, retain the original constraints, check consistency, and recompute the ranks of the **specified** derivative array. Do not assume that every differentiation raises the rank. For a regular pencil, compare the inferred index against the largest nilpotent block in Weierstrass form, when that form is accessible symbolically.

8.3 Hidden Constraints as Invariants

The reduction of Section 8.2 replaces algebraic equations by their time derivatives. The resulting system is an ODE, but it is **not** the same problem: differentiation loses information, and what is lost must be recorded.

Consider the two-equation example

$$\dot{x} + y = 0, \qquad x = 1, \tag{8.7}$$

8.2.3 Reduction Algorithm

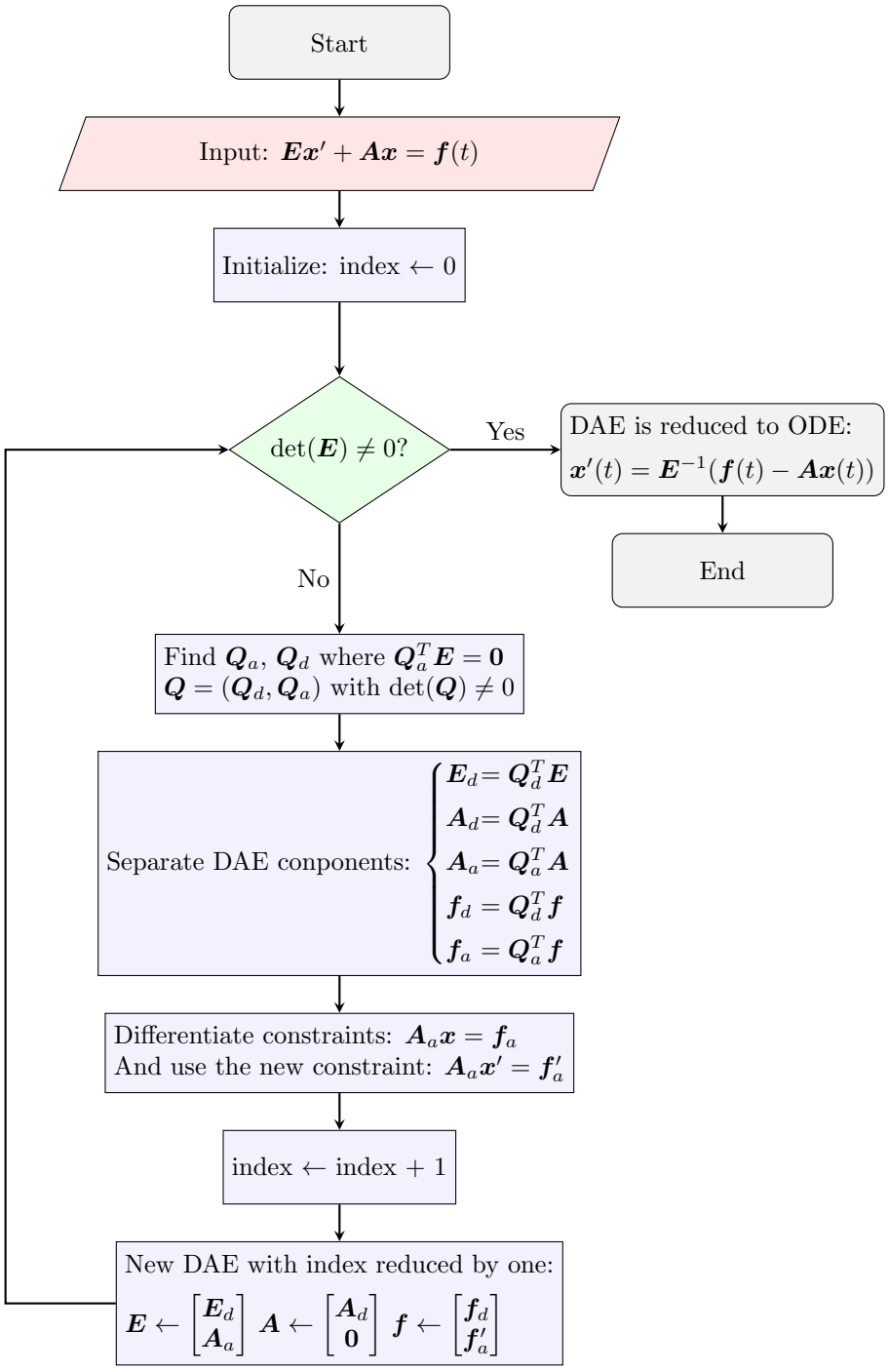


Figure 8.2

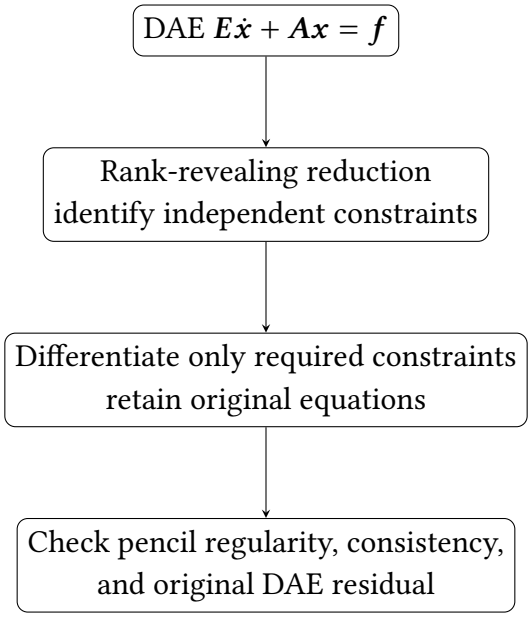


Figure 8.3: Safe logical sequence for Gaussian-style DAE analysis.

whose unique solution is $x \equiv 1$, $y \equiv 0$. Separating and differentiating the algebraic equation gives $\dot{x} = 0$, and after one further step the explicit system

$$\dot{x} = -\tfrac{1}{2}y, \qquad \dot{y} = 0. \tag{8.8}$$

Every solution of Equation 8.7 solves Equation 8.8, but the converse is false: Equation 8.8 admits $x(t) = c_1 - \frac{1}{2}c_2 t$, $y(t) = c_2$ for every pair (c_1, c_2), a two-parameter family, whereas the DAE has exactly one solution. The ODE has **more** solutions than the DAE it came from.

Definition 8.3.1 — Invariant of an ODE. A function $\Psi(\boldsymbol{x}, t)$ is an *invariant*, or *hidden constraint*, of the ODE $\dot{\boldsymbol{x}} = \boldsymbol{F}(\boldsymbol{x}, t)$ if

$$\frac{\mathrm{d}}{\mathrm{d}t}\Psi(\boldsymbol{x}(t), t) = \frac{\partial \Psi}{\partial \boldsymbol{x}}\boldsymbol{F}(\boldsymbol{x}, t) + \frac{\partial \Psi}{\partial t} \equiv \boldsymbol{0}$$

along every solution. Its value is therefore fixed by the initial condition.

The algebraic equations discarded during index reduction are exactly the invariants of the reduced ODE. In Equation 8.7 they are $x - 1 = 0$ and $y = 0$, and the extra solutions disappear once the initial condition is required to satisfy them. This is the operational rule:

> **Index reduction never discards the algebraic equations: it files them as invariants of the resulting ODE, and the initial condition must satisfy them.**

In the worked reductions of Section 8.4 the invariants determine the initial condition almost completely: in the three-variable example only one component of $\boldsymbol{x}(0)$ remains free, the other two being forced by the hidden constraints.

8.3.1 Invariants of a genuine ODE

Invariants are not a DAE phenomenon. Take

$$\dot{x} = y, \qquad \dot{y} = -x, \qquad x(0) = 1,\ y(0) = 0. \tag{8.9}$$

Multiplying the first equation by x, the second by y and adding gives

$$x\dot{x} + y\dot{y} = xy - xy = 0 \quad \Longleftrightarrow \quad \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\left(x^2 + y^2\right) = 0,$$

so $x^2 + y^2 = c$ is an invariant, with $c = x(0)^2 + y(0)^2 = 1$ fixed by the initial data. The exact trajectory is the unit circle.

8.3.2 Invariants are destroyed by the integrator

A numerical method applied to Equation 8.9 does not in general preserve $x^2 + y^2$. Explicit Euler gives

$$x_{k+1} = x_k + \Delta t\, y_k, \qquad y_{k+1} = y_k - \Delta t\, x_k.$$

Multiplying the first by x_k, the second by y_k and adding,

$$x_k x_{k+1} + y_k y_{k+1} = x_k^2 + \Delta t\, x_k y_k + y_k^2 - \Delta t\, x_k y_k = x_k^2 + y_k^2,$$

that is $\boldsymbol{v}_k \cdot \boldsymbol{v}_{k+1} = \|\boldsymbol{v}_k\|^2$ with $\boldsymbol{v}_k = (x_k, y_k)^T$. By the Cauchy–Schwarz inequality $|\boldsymbol{v}_k \cdot \boldsymbol{v}_{k+1}| \le \|\boldsymbol{v}_k\|\,\|\boldsymbol{v}_{k+1}\|$, hence

$$\|\boldsymbol{v}_k\|^2 \le \|\boldsymbol{v}_k\|\,\|\boldsymbol{v}_{k+1}\| \quad \Longrightarrow \quad \boxed{\|\boldsymbol{v}_{k+1}\| \ge \|\boldsymbol{v}_k\|} \tag{8.10}$$

The discrete trajectory spirals **outwards**, monotonically, for every step size: the invariant is not merely perturbed, it is violated in a fixed direction.

Implicit Euler,

$$x_{k+1} = x_k + \Delta t\, y_{k+1}, \qquad y_{k+1} = y_k - \Delta t\, x_{k+1} \quad \Longleftrightarrow \quad \begin{bmatrix} 1 & -\Delta t \\ \Delta t & 1 \end{bmatrix} \boldsymbol{v}_{k+1} = \boldsymbol{v}_k,$$

reverses the inequality. Multiplying the first equation by x_{k+1}, the second by y_{k+1} and adding,

$$\|\boldsymbol{v}_{k+1}\|^2 = \boldsymbol{v}_k \cdot \boldsymbol{v}_{k+1} \le \|\boldsymbol{v}_k\|\,\|\boldsymbol{v}_{k+1}\| \quad \Longrightarrow \quad \boxed{\|\boldsymbol{v}_{k+1}\| \le \|\boldsymbol{v}_k\|} \tag{8.11}$$

so the trajectory spirals **inwards**. Both methods converge as $\Delta t \to 0$ on a fixed interval, and both destroy the invariant at fixed Δt — one by inflation, one by deflation.

The remedy is not a smaller step but a method that respects the geometry: projection onto the constraint manifold, Baumgarte stabilization, or a structure-preserving integrator. These are the subject of Numerical Methods for DAEs, and the reason the hidden constraints produced by index reduction must be carried along with the reduced ODE rather than discarded.

8.4 Examples

The three examples below apply the algorithm of Section 8.2 in increasing order of difficulty: a system that becomes an ODE after two reductions, one that requires a row exchange, and one whose pencil is singular so that a variable can never be recovered.

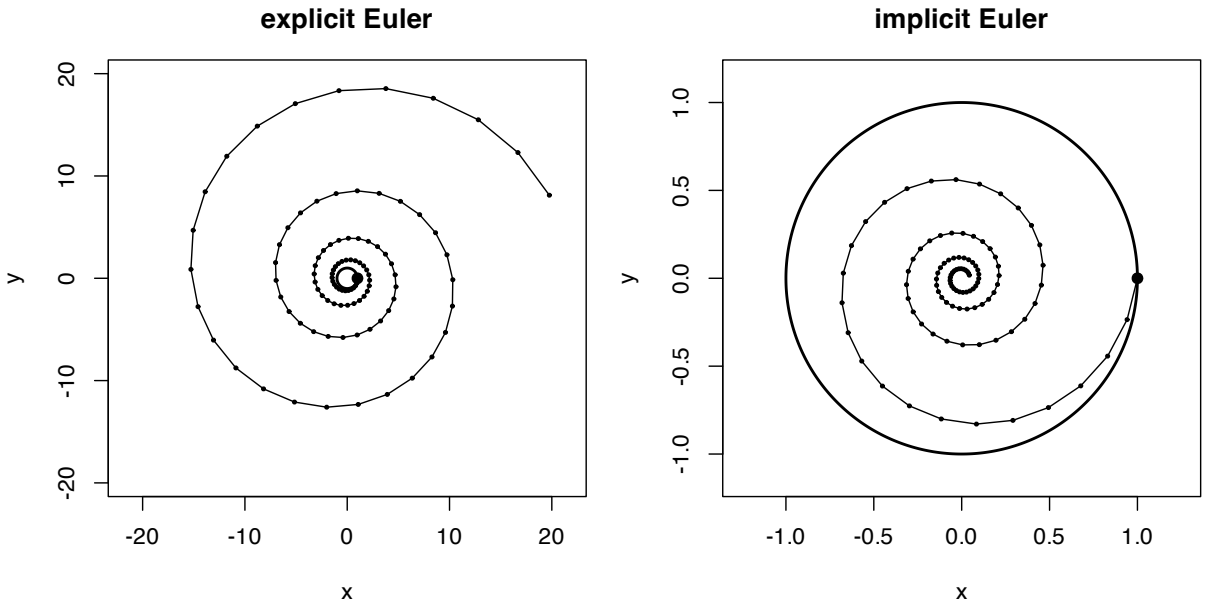


Figure 8.4: Explicit and implicit Euler applied to $\dot{x} = y$, $\dot{y} = -x$ with $\Delta t = 0.25$ over four turns. The exact trajectory is the unit circle; explicit Euler grows monotonically, implicit Euler decays monotonically, in agreement with Equation 8.10 and Equation 8.11.

Example 1: two reductions to an ODE

8.4.0.1 Linear DAE Reduction

Consider the following linear DAE:

$$x_1' + x_2 = 0,$$
$$x_1 = 1,$$

That written in matrix form become:

$$\boldsymbol{E}\boldsymbol{x}' + \boldsymbol{A}\boldsymbol{x} = \boldsymbol{f}(t),$$

with the following matrices and vectors:

$$\boldsymbol{E} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \boldsymbol{f}(t) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

8.4.0.2 First reduction

Step 1: Check if E is Singular

We compute:

$$\det(\boldsymbol{E}) = 0$$

Since $\boldsymbol{E}$ is singular, the system is a DAE.

Step 2: Decompose the System

The basis for the null space of $\boldsymbol{E}^T$ is: $\boldsymbol{Q}_a = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

The complementary vector is: $\boldsymbol{Q}_d = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

We construct the matrix $\boldsymbol{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Multiplying the system by $\boldsymbol{Q}^T$, we get the new matrices:

$$
\begin{aligned}
\boldsymbol{E}_d = \boldsymbol{Q}_d^T \boldsymbol{E} &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \\
\boldsymbol{A}_d = \boldsymbol{Q}_d^T \boldsymbol{A} &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \end{bmatrix}, \\
f_d(t) = \boldsymbol{Q}_d^T f(t) &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} &= \begin{bmatrix} 0 \end{bmatrix}, \\
\boldsymbol{A}_a = \boldsymbol{Q}_a^T \boldsymbol{A} &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \\
f_a(t) = \boldsymbol{Q}_a^T f(t) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} &= \begin{bmatrix} 1 \end{bmatrix}.
\end{aligned}
$$

Step 3: Solve the System

The system becomes:

1. **Differential part:** $x_1' + x_2 = 0$.
2. **Algebraic part:** $x_1 = 1$.

Differentiating the algebraic constraint $x_1 = 1$:

$$x_1' = 0.$$

At this point the DAE become:

$$\widetilde{\boldsymbol{E}}\boldsymbol{x}' + \widetilde{\boldsymbol{A}}\boldsymbol{x} = \widetilde{\boldsymbol{f}}(t)$$

with the following matrices and vectors:

$$\widetilde{\boldsymbol{E}} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad \widetilde{\boldsymbol{A}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \widetilde{\boldsymbol{f}}(t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

8.4.0.3 Second Reduction

Now reapply the algorithm:

Step 1: Check if $\widetilde{E}$ is Singular

We compute:

$$\det(\widetilde{\boldsymbol{E}}) = 0$$

Since $\widetilde{\boldsymbol{E}}$ is singular, the system is still a DAE.

Step 2: Decompose the System

The basis for the null space of $\widetilde{\boldsymbol{E}}^T$ is: $\widetilde{\boldsymbol{Q}}_a = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$.

The complementary vector is: $\widetilde{\boldsymbol{Q}}_d = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$.

We construct the matrix $\widetilde{\boldsymbol{Q}} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Multiplying the system by $\widetilde{\boldsymbol{Q}}^T$, we get the new matrices:

$$
\begin{aligned}
\widetilde{\boldsymbol{E}}_d = \widetilde{\boldsymbol{Q}}_d^T \widetilde{\boldsymbol{E}} &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} &&= \begin{bmatrix} 2 & 0 \end{bmatrix}, \\
\widetilde{\boldsymbol{A}}_d = \widetilde{\boldsymbol{Q}}_d^T \widetilde{\boldsymbol{A}} &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} &&= \begin{bmatrix} 0 & 1 \end{bmatrix}, \\
\widetilde{\boldsymbol{f}}_d(t) = \widetilde{\boldsymbol{Q}}_d^T \widetilde{\boldsymbol{f}}(t) &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} &&= \begin{bmatrix} 0 \end{bmatrix}, \\
\widetilde{\boldsymbol{A}}_a = \widetilde{\boldsymbol{Q}}_a^T \widetilde{\boldsymbol{A}} &= \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} &&= \begin{bmatrix} 0 & 1 \end{bmatrix}, \\
\widetilde{\boldsymbol{f}}_a(t) = \widetilde{\boldsymbol{Q}}_a^T \widetilde{\boldsymbol{f}}(t) &= \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} &&= \begin{bmatrix} 0 \end{bmatrix}.
\end{aligned}
$$

Step 3: Solve the System

The system becomes:

1. **Differential part:** $2x_1' + x_2 = 0$.
2. **Algebraic part:** $x_2 = 0$.

Differentiating the algebraic constraint $x_2 = 0$:

$$x_2' = 0.$$

At this point the DAE become:

$$\widetilde{\widetilde{\boldsymbol{E}}}\boldsymbol{x}' + \widetilde{\widetilde{\boldsymbol{A}}}\boldsymbol{x} = \widetilde{\widetilde{\boldsymbol{f}}}(t)$$

with the following matrices and vectors:

$$\widetilde{\widetilde{\boldsymbol{E}}} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, \quad \widetilde{\widetilde{\boldsymbol{A}}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \widetilde{\widetilde{\boldsymbol{f}}}(t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Now reapply the algorithm. At the step 1 $\det(\widetilde{\widetilde{\boldsymbol{E}}}) = 2 \neq 0$ so now we have an ODE:

$$
\begin{aligned}
2x_1' + x_2 &= 0, \\
x_2' &= 0.
\end{aligned}
$$

we have done two derivation, thus the **index of original DAE is 2**.

Example 2: reduction with a row exchange

Consider the DAE

$$
\begin{cases}
x'(t) = z(t) \\
z'(t) = y(t) - t \\
x'(t) = z(t) + x(t) + t + 1
\end{cases}
$$

define

$$\boldsymbol{E} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \qquad \boldsymbol{A} = \begin{pmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ -1 & 0 & -1 \end{pmatrix}, \qquad \boldsymbol{f}(t) = \begin{pmatrix} 0 \\ -t \\ t+1 \end{pmatrix}$$

so that the DAE can be written as

$$\boldsymbol{E}\,\boldsymbol{x}'(t) + \boldsymbol{A}\,\boldsymbol{x}(t) = \boldsymbol{f}(t)$$

The table

$$T = \left[\begin{array}{c|c|c} E & A & f(t) \end{array}\right] = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ 1 & 0 & 0 & -1 & 0 & -1 & t+1 \end{array}\right]$$

is used to memorize the operation done on the DAE. Let index $\leftarrow 0$.

1. Multiply table T on the left by L_1 where

$$L_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

so that

$$T_1 = L_1 T = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ \hline 0 & 0 & 0 & -1 & 0 & 0 & t+1 \end{array}\right]$$

The last row contain the algebraic equation

$$-x(t) = t + 1$$

2. First block now is in echelon form. Apply derivation of the last row (reduce index by one). Let index $\leftarrow 1$.

$$T_2 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ \hline -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

3. Multiply table T_2 on the left by L_2 where

$$L_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

so that

$$T_3 = L_2 T_2 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ \hline 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{array}\right]$$

The last row contain the algebraic equation

$$-z(t) = 1$$

4. First block again is in echelon form. Apply derivation of the last row (reduce index by one). Let index $\leftarrow 2$.

$$T_4 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ \hline 0 & 0 & -1 & 0 & 0 & 0 & 0 \end{array}\right]$$

5. Multiply table T_4 on the left by L_4 where

$$L_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

so that

$$T_5 = L_4 T_4 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ \hline 0 & 0 & 0 & 0 & -1 & 0 & -t \end{array}\right]$$

The last row contain the algebraic equation

$$-y(t) = -t$$

6. First block again is in echelon form. Apply derivation of the last row (reduce index by one). Let index $\leftarrow 3$.

$$T_6 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \\ \hline 0 & -1 & 0 & 0 & 0 & 0 & -1 \end{array}\right]$$

7. Exchange last two rows and multily by -1 second row

$$P_6 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}$$

multilying T_6 by P_6:

$$P_6 T_6 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & -t \end{array}\right]$$

- The ODE after index reduction

$$\begin{cases} x'(t) = z(t) \\ y'(t) = 1 \\ z'(t) = y(t) - t \end{cases}$$

- The algebraic (hidden) constraints

$$\begin{aligned} -x(t) &= t + 1 \\ -z(t) &= 1 \\ -y(t) &= -t \end{aligned} \quad \Longrightarrow \quad \begin{cases} x(t) = -t - 1 \\ z(t) = -1 \\ y(t) = t \end{cases}$$

- The **differential index** is 3

Notice that the hiddent constaints determine the complete solution and the initial condition of the ODE $(x(0), y(0), z(0)) = (-1, -1, 0)$.

Example 3: a singular pencil

Consider the DAE

$$\begin{cases} x'(t) = z(t) \\ x'(t) = y(t) - \sin t \\ x'(t) = z(t) + y(t) + \cos t \end{cases}$$

define

$$\boldsymbol{E} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \qquad \boldsymbol{A} = \begin{pmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ 0 & -1 & -1 \end{pmatrix}, \qquad \boldsymbol{f}(t) = \begin{pmatrix} 0 \\ -\sin t \\ \cos t \end{pmatrix}$$

so that the DAE can be written as

$$\boldsymbol{E}\,\boldsymbol{x}'(t) + \boldsymbol{A}\,\boldsymbol{x}(t) = \boldsymbol{f}(t)$$

The table

$$\boldsymbol{T} = \left[\begin{array}{c|c|c} \boldsymbol{E} & \boldsymbol{A} & \boldsymbol{f}(t) \end{array}\right] = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & -\sin t \\ 1 & 0 & 0 & 0 & -1 & -1 & \cos t \end{array}\right]$$

is used to memorize the operation done on the DAE. Let index $\leftarrow 0$.

1. Multiply table $\boldsymbol{T}$ on the left by $\boldsymbol{L}_1$ where

$$\boldsymbol{L}_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

so that

$$\boldsymbol{T}_1 = \boldsymbol{L}_1\boldsymbol{T} = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ \hline 0 & 0 & 0 & 0 & -1 & +1 & -\sin t \\ 0 & 0 & 0 & 0 & -1 & 0 & \cos t \end{array}\right]$$

The last two rows contains the algebraic equations

$$-y(t) + z(t) = -\sin t, \qquad -y(t) = \cos(t)$$

2. First block now is in echelon form. Apply derivation of the last rows (reduce index by one). Let index $\leftarrow 1$.

$$\boldsymbol{T}_2 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ \hline 0 & -1 & +1 & 0 & 0 & 0 & -\cos t \\ 0 & -1 & 0 & 0 & 0 & 0 & -\sin t \end{array}\right]$$

3. Multiply table $\boldsymbol{T}_2$ on the left by $\boldsymbol{L}_2$ where

$$\boldsymbol{L}_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$$

so that

$$\boldsymbol{T}_3 = \boldsymbol{L}_2\boldsymbol{T}_2 = \left[\begin{array}{ccc|ccc|c} 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & +1 & 0 & 0 & 0 & -\cos t \\ 0 & 0 & -1 & 0 & 0 & 0 & \cos t - \sin t \end{array}\right]$$

- The ODE after index reduction

$$\begin{aligned} x'(t) &= z(t) \\ -y'(t) + z'(t) &= -\cos t \\ -z'(t) &= \cos t - \sin t \end{aligned} \quad \Longrightarrow \quad \begin{cases} x'(t) = z(t) \\ y'(t) = \sin t \\ z'(t) = \sin t - \cos t \end{cases}$$

- The algebraic (hidden) constraints

$$\begin{aligned} -y(t) + z(t) &= -\sin t, \\ -y(t) &= \cos t \end{aligned} \quad \Longrightarrow \quad \begin{cases} y(t) = -\cos t \\ z(t) = \cos t - \sin t \end{cases}$$

- The **differential index** is 2

Notice that the hiddent constaints determine **part** of complete solution and the initial condition of the ODE $(x(0), y(0), z(0)) = (\text{free}, -1, 0)$. Thus, only $z(0)$ must be set to determine the solution.

Linear DAE Example

Consider the following set of linear differential equation,

$$\begin{aligned} x_1'(t) + x_2'(t) + x_3'(t) &= 2x_1(t) + x_3(t) + t, \\ x_1'(t) + x_2'(t) + x_3'(t) &= x_1(t) + x_2(t) + t^3, \\ 0 &= x_1(t) + x_2(t) + x_3(t) + \sin(t), \end{aligned}$$

that we can write in a more compact matrix form as follows,

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1' \\ x_2' \\ x_3' \end{bmatrix} - \begin{bmatrix} 2 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} t \\ t^3 \\ \sin(t) \end{bmatrix}.$$

Which corresponds to the usual form $\boldsymbol{E}\boldsymbol{x}'(t) + \boldsymbol{A}\boldsymbol{x}(t) = \boldsymbol{f}(t)$ obvious meaning for the symbols. Now consider to form the table

$$\begin{bmatrix} \boldsymbol{E} & \boldsymbol{A} & \boldsymbol{f}(t) \end{bmatrix} \in \mathbb{R}^{n \times 2n+1}$$

in such a way that,

$$\begin{bmatrix} \boldsymbol{E} & \boldsymbol{A} & \boldsymbol{f}(t) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{x} \\ -1 \end{bmatrix} = \boldsymbol{0}$$

First it is necessary to check if the pair $(\boldsymbol{E}, \boldsymbol{A})$ is a regular pencil,

$$\left| \boldsymbol{A} - \lambda \boldsymbol{E} \right| = \begin{vmatrix} -2-\lambda & -\lambda & -1-\lambda \\ -1-\lambda & -1-\lambda & -\lambda \\ -1 & -1 & -1 \end{vmatrix} = -2$$

the determinant is different from zero, so the pair $(\boldsymbol{E}, \boldsymbol{A})$ is a *regular pencil.*
Now it is easy to manipulate the following table

$$\begin{bmatrix} 1 & 1 & 1 & -2 & 0 & -1 & t \\ 1 & 1 & 1 & -1 & -1 & 0 & t^3 \\ 0 & 0 & 0 & -1 & -1 & -1 & \sin(t) \end{bmatrix}$$

subtract first row to the second row

$$\begin{bmatrix} 1 & 1 & 1 & -2 & 0 & -1 & t \\ 0 & 0 & 0 & 1 & -1 & 1 & t^3 - t \\ 0 & 0 & 0 & -1 & -1 & -1 & \sin(t) \end{bmatrix}$$

constraints:

$$x_1 - x_2 + x_3 - t^3 + t = 0$$
$$x_1 + x_2 + x_3 + \sin(t) = 0$$

---

differentiate second and third row (reduce index by one)

$$\begin{bmatrix} 1 & 1 & 1 & 2 & 0 & 1 & t \\ -1 & 1 & -1 & 0 & 0 & 0 & 3t^2 - 1 \\ -1 & -1 & -1 & 0 & 0 & 0 & \cos(t) \end{bmatrix}$$

add first row to the second and the third row

$$\begin{bmatrix} 1 & 1 & 1 & 2 & 0 & 1 & t \\ 0 & 2 & 0 & 2 & 0 & 1 & 3t^2 + t - 1 \\ 0 & 0 & 0 & 2 & 0 & 1 & \cos(t) + t \end{bmatrix}$$

constraint:

$$2x_1 + x_3 - \cos(t) - t = 0$$

---

derive third row (reduce index by one)

$$\begin{bmatrix} 1 & 1 & 1 & 2 & 0 & 1 & t \\ 0 & 2 & 0 & 2 & 0 & 1 & 3t^2 + t - 1 \\ 2 & 0 & 1 & 0 & 0 & 0 & -\sin(t) + 1 \end{bmatrix}$$

subtract 2 times first row to third

$$\begin{bmatrix} 1 & 1 & 1 & 2 & 0 & 1 & t \\ 0 & 2 & 0 & 2 & 0 & 1 & 3t^2 + t - 1 \\ 0 & -2 & -1 & -4 & 0 & -2 & -\sin(t) + 1 - 2t \end{bmatrix}$$

add second row to third

$$\begin{bmatrix} 1 & 1 & 1 & 2 & 0 & 1 & t \\ 0 & 2 & 0 & 2 & 0 & 1 & 3t^2 + t - 1 \\ 0 & 0 & -1 & -2 & 0 & -1 & 3t^2 - t - \sin(t) \end{bmatrix}$$

add third row to first and then multiply third row by (-1) and second row by $(1/2)$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 3t^2 - \sin(t) \\ 0 & 1 & 0 & 1 & 0 & 1 & \frac{1}{2}(3t^2 + t - 1) \\ 0 & 0 & 1 & 2 & 0 & 1 & t + \sin(t) - 3t^2 \end{bmatrix}$$

subtract second row to first

$$\left[\begin{array}{cccccc|c} 1 & 0 & 0 & -1 & 0 & -1 & \frac{3}{2}t^2+\frac{1}{2}(1-t)-\sin(t) \\ 0 & 1 & 0 & 1 & 0 & 1 & \frac{1}{2}(3t^2+t-1) \\ 0 & 0 & 1 & 2 & 0 & 1 & 3t^2-t-\sin(t) \end{array}\right]$$

then we obtain the ODE

$$\begin{cases} x_1' = x_1 + x_3 + \dfrac{3}{2}t^2 + \dfrac{1}{2}(1-t) - \sin(t) \\ x_2' = -x_1 - x_3 + \dfrac{1}{2}(3t^2 + t - 1) \\ x_3' = -2x_1 - x_3 + 3t^2 - t - \sin(t) \end{cases}$$

with the hidden invariants (or constraints)

$$x_1 - x_2 + x_3 - t^3 + t = 0$$
$$x_1 + x_2 + x_3 + \sin(t) = 0$$
$$2x_1 + x_3 - \cos(t) - t = 0$$

8.4.1 Worked reduction: a regular index-two pencil

Take

$$E = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad A = I_2, \qquad f(t) = \begin{bmatrix} t \\ t^2 \end{bmatrix}, \qquad E\dot{x} + Ax = f.$$

Since $\det(A + \lambda E) = 1$, the pencil is regular. The two scalar equations are

$$\dot{x}_2 + x_1 = t, \qquad x_2 = t^2.$$

Differentiating the algebraic constraint gives $\dot{x}_2 = 2t$, hence $x_1 = -t$. The solution is uniquely determined:

$$x(t) = (-t, t^2)^T, \qquad E\dot{x} + Ax - f = 0.$$

The consistent initial condition is $x(0) = 0$; prescribing $x_1(0) = 1$ would make the IVP inconsistent. The nilpotent block is E itself and has degree two, so the index is two.

8.4.2 Worked reduction: a singular pencil and a missing variable

Consider the three-dimensional example

$$E = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad f(t) = \begin{bmatrix} t \\ 1 \\ t^2 \end{bmatrix}, \quad E\dot{x} + Ax = f.$$

The middle column of both E and A is zero, so x_2 is absent from every equation. Moreover, the second and third equations yield

$$2\dot{x}_1 = 1, \qquad \dot{x}_1 + x_3 = t^2, \quad \Longrightarrow \quad \dot{x}_1 = \frac{1}{2}, \quad x_3 = t^2 - \frac{1}{2}.$$

The first equation then requires

$$\dot{x}_3 + x_1 + x_3 = t - \frac{1}{2} \quad \Longrightarrow \quad x_1 = -t^2 - t,$$

which contradicts $\dot{x}_1 = 1/2$. Thus this particular forcing is **incompatible**: no classical solution exists. Even for a compatible forcing, $x_2(t)$ would remain arbitrary. This illustrates two separate pathologies of singular pencils: nonexistence and nonuniqueness.

Indeed

$$\det(\boldsymbol{A} - \lambda \boldsymbol{E}) \equiv 0,$$

because its second column is identically zero. Differentiating constraints does not restore a missing equation. This is the key lesson of the final handwritten example: a formally obtained ODE for some variables need not determine the full state.

8.4.3 Exercises with solutions

Exercise 1 (regularity). Determine the regularity of $\boldsymbol{E} = \operatorname{diag}(1, 0)$ and $\boldsymbol{A} = \operatorname{diag}(0, 1)$. **Solution:** $\det(\boldsymbol{A} - \lambda \boldsymbol{E}) = -\lambda$ is not identically zero, so the pencil is regular.

Exercise 2 (consistency). For $\dot{x}_2 + x_1 = \sin t$, $x_2 = t$, determine the consistent initial state. **Solution:** $x_1 = \sin t - 1$, hence $(x_1(0), x_2(0)) = (-1, 0)$.

Exercise 3 (index). Let $\boldsymbol{N}$ be the 3×3 nilpotent Jordan block. Determine the index and maximum derivative order of $\boldsymbol{g}$. **Solution:** $\boldsymbol{N}^3 = 0$ but $\boldsymbol{N}^2 \neq 0$, so the index is three and the highest derivative is $\boldsymbol{g}''$.

Exercise 4 (singularity). If $\boldsymbol{E}$ and $\boldsymbol{A}$ have a common zero column, can their pencil be regular? **Solution:** No: $\boldsymbol{A} - \lambda \boldsymbol{E}$ has a zero column for every λ, hence its determinant vanishes identically.

9. The Differential Index

The index of a DAE measures how far it is from an ODE: it is the number of times the equations must be differentiated before an explicit $\dot{\boldsymbol{x}} = \boldsymbol{F}(\boldsymbol{x}, t)$ can be extracted. For the constant-coefficient case Chapter 7 gave this number as the nilpotency degree of the Weierstrass algebraic block; this chapter gives the definition that applies to any DAE, linear or not, and relates it to the other notions of index in use.

9.1 DAE Index

The **differential index** describes how many differentiations of the DAE equations are needed before the derivatives of all state variables can be determined locally as functions of the state and time. It is a property of a specified formulation, together with the regularity and rank conditions used in the calculation.

9.1.1 Differential index

Consider an implicit system

$$\boldsymbol{F}\left(\boldsymbol{x}(t), \boldsymbol{x}'(t), t\right) = \boldsymbol{0}, \qquad \boldsymbol{x}(t) \in \mathbb{R}^n.$$

When the Jacobian $\boldsymbol{F}_{\boldsymbol{x}'}$ is singular, the system cannot be solved for $\boldsymbol{x}'$ directly: some equations, or some combinations of equations, do not contain $\boldsymbol{x}'$ at all and act as algebraic constraints. The missing information is hidden in the time derivatives of the equations.

9.1.1.1 The sequence of derivatives (derivative array)

Assume $\boldsymbol{F}$ is sufficiently smooth and let $\boldsymbol{x}(t)$ be a solution. Since $\boldsymbol{F}\left(\boldsymbol{x}(t), \boldsymbol{x}'(t), t\right) = \boldsymbol{0}$ for every t, all its time derivatives vanish as well:

$$
\begin{aligned}
\mathbf{0} &= \boldsymbol{F}(\boldsymbol{x}, \boldsymbol{x}', t),\\
\mathbf{0} &= \frac{d}{dt}\boldsymbol{F} = \boldsymbol{F}_{\boldsymbol{x}'}\,\boldsymbol{x}'' + \boldsymbol{F}_{\boldsymbol{x}}\,\boldsymbol{x}' + \boldsymbol{F}_t,\\
\mathbf{0} &= \frac{d^2}{dt^2}\boldsymbol{F} = \boldsymbol{F}_{\boldsymbol{x}'}\,\boldsymbol{x}''' + \boldsymbol{\Phi}_2(\boldsymbol{x}, \boldsymbol{x}', \boldsymbol{x}'', t),\\
&\;\;\vdots\\
\mathbf{0} &= \frac{d^k}{dt^k}\boldsymbol{F} = \boldsymbol{F}_{\boldsymbol{x}'}\,\boldsymbol{x}^{(k+1)} + \boldsymbol{\Phi}_k\big(\boldsymbol{x}, \boldsymbol{x}', \dots, \boldsymbol{x}^{(k)}, t\big),
\end{aligned}
$$

where $\boldsymbol{\Phi}_k$ collects all chain-rule terms that do not contain the highest derivative $\boldsymbol{x}^{(k+1)}$. Stacking the original equations and their first q derivatives gives the **derivative array** of order q:

$$
\boldsymbol{\mathcal{F}}_q\big(\boldsymbol{x}, \boldsymbol{x}', \boldsymbol{x}'', \dots, \boldsymbol{x}^{(q+1)}, t\big) = \begin{pmatrix} \boldsymbol{F} \\ \dfrac{d}{dt}\boldsymbol{F} \\ \vdots \\ \dfrac{d^q}{dt^q}\boldsymbol{F} \end{pmatrix} = \mathbf{0}. \tag{9.1}
$$

For fixed $(\boldsymbol{x}, t)$, Equation 9.1 is a system of $(q+1)n$ algebraic equations in which $\boldsymbol{x}', \boldsymbol{x}'', \dots, \boldsymbol{x}^{(q+1)}$ are treated as independent unknowns.

9.1.1.2 Extracting an ODE from the array

The derivative array need not determine all its unknowns: the highest derivatives usually remain free. This is not needed. Only $\boldsymbol{x}'$ matters. By algebraic manipulations of the array (solving some rows for some unknowns, substituting, taking linear combinations of rows) one eliminates $\boldsymbol{x}'', \dots, \boldsymbol{x}^{(q+1)}$. If, for some q, these manipulations fix $\boldsymbol{x}'$ uniquely as a function of $\boldsymbol{x}$ and t, the result is the **underlying ODE**

$$
\boldsymbol{x}'(t) = \boldsymbol{G}\big(\boldsymbol{x}(t), t\big).
$$

During the extraction the equations of the array fall into three groups:

1. **Equations used to determine $\boldsymbol{x}'$.** They produce the underlying ODE.
2. **Equations that become invariants.** After elimination they give relations between $\boldsymbol{x}$ and t only: the original algebraic constraints and the *hidden constraints* produced by differentiation. They are not part of the ODE, but every solution of the DAE satisfies them. A solution of $\boldsymbol{x}' = \boldsymbol{G}(\boldsymbol{x}, t)$ that satisfies them at the initial time satisfies them for all t, so they are invariants of the underlying ODE and characterize the consistent initial values.
3. **Equations that are discarded.** The remaining rows are not needed to determine $\boldsymbol{x}'$. Along a solution of the DAE they hold automatically, because they are time derivatives of the DAE equations.

The DAE is therefore equivalent to the underlying ODE *together with* its invariants, not to the ODE alone. Integrating only $\boldsymbol{x}' = \boldsymbol{G}(\boldsymbol{x}, t)$ numerically lets the invariants drift away (see Constraint Drift).

The differentiation index counts how far the array must be extended before this extraction succeeds. The definition presumes that the necessary Jacobians have the ranks required for

the stated local solution; without those conditions the index may not be defined or may vary along a trajectory.

Definition 9.1.1 — Definition (differentiation index). The differentiation index of a regular DAE is the least nonnegative integer q such that the derivative array $\boldsymbol{\mathcal{F}}_q = \mathbf{0}$ of Equation 9.1 determines $\boldsymbol{x}'$ locally and uniquely as a function of $(\boldsymbol{x}, t)$, so that algebraic manipulations of the array yield the underlying ODE $\boldsymbol{x}' = \boldsymbol{G}(\boldsymbol{x}, t)$ for every component of the state. An ODE already in explicit form has index zero.

9.1.1.3 Colour convention for the examples

In the examples that follow, the derivatives of **all** the equations of the DAE are written out, row by row, up to the order q of the index. Row k contains the k-th time derivative of each equation, and column (i) refers to the i-th equation of the DAE in the order written:

- uncoloured equations are used to build the underlying ODE (or to express an invariant in the state variables);
- red equations are discarded;
- green equations become invariants.

9.1.1.4 A direct calculation: the semi-explicit index-one case

For

$$\begin{aligned} \boldsymbol{x}' &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{z}, t), \\ \mathbf{0} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{z}, t), \end{aligned}$$

the derivative array of order one is

k	eq. (1)	eq. (2)
0	$\boldsymbol{x}' - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{z}, t) = \mathbf{0}$	$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{z}, t) = \mathbf{0}$
1	$\boldsymbol{x}'' - \boldsymbol{f}_{\boldsymbol{x}}\boldsymbol{x}' - \boldsymbol{f}_{\boldsymbol{z}}\boldsymbol{z}' - \boldsymbol{f}_t = \mathbf{0}$	$\boldsymbol{g}_{\boldsymbol{x}}\boldsymbol{x}' + \boldsymbol{g}_{\boldsymbol{z}}\boldsymbol{z}' + \boldsymbol{g}_t = \mathbf{0}$

The row $k = 0$ alone does not contain $\boldsymbol{z}'$, so $q = 0$ is not enough. The first row gives $\boldsymbol{x}' = \boldsymbol{f}$. If $\boldsymbol{g}_{\boldsymbol{z}}$ is nonsingular, the differentiated constraint gives

$$\boldsymbol{z}' = -\boldsymbol{g}_{\boldsymbol{z}}^{-1}\left(\boldsymbol{g}_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{z}, t) + \boldsymbol{g}_t\right),$$

so one differentiation determines the derivatives of both $\boldsymbol{x}$ and $\boldsymbol{z}$. The differentiated first equation only determines $\boldsymbol{x}''$ and is discarded; the constraint $\boldsymbol{g} = \mathbf{0}$ becomes an invariant. This is the regular index-one case introduced in Section 10.1.

9.1.2 Worked exercises using the definition

The following calculations use only differentiation and algebraic elimination. In each case, count the differentiations needed to determine the derivatives of **all** state components.

9.1.2.1 Exercise 1: an explicit ODE (index zero)

Consider

$$x'(t) = -x(t).$$

The derivative is already explicit, so no differentiation is required. The array of order zero is the equation itself, nothing is discarded and there is no invariant. The index is $q = 0$.

9.1.2.2 Exercise 2: a regular semi-explicit DAE (index one)

Consider

$$x'(t) = -y(t),$$
$$0 = x(t) + y(t) - \sin t.$$

The derivatives of both equations up to order one are

k	eq. (1)	eq. (2)
0	$x' + y = 0$	$x + y - \sin t = 0$
1	$x'' + y' = 0$	$x' + y' - \cos t = 0$

At order zero y' does not appear, so $q \geq 1$. The first equation gives $x' = -y$; the differentiated constraint then gives $y' = \cos t - x' = y + \cos t$. Therefore

$$x' = -y, \qquad y' = y + \cos t,$$

and one differentiation suffices: $q = 1$. Equivalently, the constraint Jacobian is $g_y = 1$. The equation $x'' + y' = 0$ only determines x'' and is discarded. The invariant is $x + y - \sin t = 0$; indeed, along the ODE, $\frac{d}{dt}(x + y - \sin t) = -y + (y + \cos t) - \cos t = 0$.

9.1.2.3 Exercise 3: correcting the prescribed-trajectory example (index two)

Consider

$$0 = x(t) - \sin t,$$
$$0 = x'(t) + y(t).$$

The derivatives of both equations up to order two are

k	eq. (1)	eq. (2)
0	$x - \sin t = 0$	$x' + y = 0$
1	$x' - \cos t = 0$	$x'' + y' = 0$
2	$x'' + \sin t = 0$	$x''' + y'' = 0$

With the array of order one, $x' = \cos t$ is determined, but $y' = -x''$ is not, because x'' is still free. The second derivative of the first equation fixes $x'' = -\sin t$, hence $y' = \sin t$. Thus the explicit system is

$$x' = \cos t, \qquad y' = \sin t,$$

and the maximum differentiation order needed is two: $q = 2$. The equation $x''' + y'' = 0$ is discarded. The invariants are $x - \sin t = 0$ and, using $x' = \cos t$, $y + \cos t = 0$. They fix the state completely, so the only consistent initial value at t_0 is $x = \sin t_0$, $y = -\cos t_0$. The earlier calculation of this example stopped before deriving y', so it did not establish the claimed index.

9.1.2.4 Exercise 4: a simple index-two chain

Consider the semi-explicit DAE

$$x'(t) = y(t),$$
$$0 = x(t).$$

The derivatives of both equations up to order two are

k	eq. (1)	eq. (2)
0	$x' - y = 0$	$x = 0$
1	$x'' - y' = 0$	$x' = 0$
2	$x''' - y'' = 0$	$x'' = 0$

The first equation gives $x' = y$. At order one, $y' = x''$ is not yet determined; the second derivative of the constraint gives $x'' = 0$, hence $y' = 0$. The resulting explicit ODE is

$$x' = y, \qquad y' = 0,$$

so $q = 2$. The equation $x''' - y'' = 0$ is discarded. The invariants are the constraint $x = 0$ and the hidden constraint $x' = 0$, which in the state variables reads $y = 0$.

9.1.2.5 Exercise 5: a simple index-three chain

Consider

$$\begin{aligned} x'(t) &= y(t), \\ y'(t) &= z(t), \\ 0 &= x(t). \end{aligned}$$

The derivatives of all three equations up to order three are

k	eq. (1)	eq. (2)	eq. (3)
0	$x' - y = 0$	$y' - z = 0$	$x = 0$
1	$x'' - y' = 0$	$y'' - z' = 0$	$x' = 0$
2	$x''' - y'' = 0$	$y''' - z'' = 0$	$x'' = 0$
3	$x^{(4)} - y''' = 0$	$y^{(4)} - z''' = 0$	$x''' = 0$

The first two equations give $x' = y$ and $y' = z$. For z' the chain $z' = y'' = x'''$ uses the rows $y'' - z' = 0$ and $x''' - y'' = 0$, and only the third derivative of the constraint fixes $x''' = 0$. Only after three differentiations is the derivative of every component determined. The equivalent explicit ODE on the consistent solution set is $x' = y$, $y' = z$, $z' = 0$, and the differentiation index is $q = 3$.

The red rows are discarded. The invariants are $x = 0$ and the hidden constraints $x' = 0$ and $x'' = 0$; with $x' = y$ and $x'' = y' = z$ (the row $x'' - y' = 0$) they read $y = 0$ and $z = 0$.

9.1.2.6 An analogy: Taylor series for an ODE

The repeated differentiation of Definition 9.1.1 is less exotic than it looks: it is the same operation that produces a Taylor method for an ODE. For

$$\dot{y} = t\,y^2 + t^2, \qquad y(0) = 1,$$

the derivatives of the solution are obtained by differentiating the equation itself and substituting back,

$$\begin{aligned} \dot{y} &= t\,y^2 + t^2, \\ \ddot{y} &= y^2 + 2t\,y\dot{y} + 2t = y^2 + 2t\,y\,(t\,y^2 + t^2) + 2t, \\ \dddot{y} &= 4y\dot{y} + 2t\,\dot{y}^2 + 2t\,y\ddot{y} + 2, \end{aligned}$$

and the local expansion $y_{k+1} = y_k + \dot{y}_k\Delta t + \ddot{y}_k\frac{\Delta t^2}{2} + \dddot{y}_k\frac{\Delta t^3}{6} + \mathcal{O}(\Delta t^4)$ is a numerical method of arbitrary order, built from nothing but the equation and the chain rule.

The DAE case differs in what the differentiations are *for*. For an ODE they produce higher-order accuracy and may be stopped at any point; for a DAE they produce the missing differential equations, and stopping early leaves a system that cannot be integrated at all. The index is precisely the number of differentiations that cannot be skipped. In both cases, however, the practical obstacle is the same — the expressions grow quickly — and it is the reason both subjects turn to structured algorithms rather than to the definition.

9.1.3 Why direct differentiation becomes difficult

These small examples can be handled by hand. For a general nonlinear DAE, successive differentiation creates many product and chain-rule terms, may introduce higher derivatives, and requires careful elimination while retaining the original constraints. Rank loss, redundant equations, or inconsistent initial data can make a seemingly valid differentiation step misleading. Consequently, direct application of the definition becomes difficult even for moderate systems. The following chapters develop structured algorithms for linear and nonlinear DAEs; these methods organize the required differentiation, expose hidden constraints, and support rank and consistency checks.

9.1.4 Other notions of index

The word *index* is used for several related but distinct quantities. Their values can agree for important regular classes, but there is no general equivalence without assumptions on the formulation and its ranks.

9.1.4.1 Perturbation index

The **perturbation index** measures how derivatives of a residual perturbation enter estimates for the solution error. It is tied to stability and error analysis, and therefore depends on the perturbation model and the norms used. It is not defined simply by counting differentiated constraints.

9.1.4.2 Tractability and strangeness indices

For linear or suitably structured systems, the **tractability index** uses a sequence of projectors to separate differential and algebraic directions. The **strangeness index** uses derivative arrays and rank-revealing transformations to expose hidden constraints and remove redundant equations; it is particularly useful for time-varying and over- or underdetermined systems. These constructions differ from direct differentiation of the full state equations, and their numerical values need not match the differentiation index for arbitrary systems.

9.1.4.3 Differentiation index and linear-pencil indices

The differentiation index used in this chapter is based on the number of time differentiations needed to obtain an explicit ODE. For a regular constant-coefficient linear pencil in Weierstrass form, it is related to the nilpotency degree of the algebraic block (the Kronecker or nilpotency index), with conventions depending on whether an ODE is assigned index zero or one. Under the standard convention that an ODE has index zero, these indices agree for regular pencils. The correspondence should not be transferred to singular pencils, general nonlinear DAEs, or arbitrary differentiation algorithms without a separate theorem. A useful corollary of the agreement is that the highest derivative of the forcing term required by the algebraic solution Equation 7.9 is one less than the index.

10. Semi-Explicit Form and Its Classification

Any DAE $F(x, \dot{x}, t) = 0$ can be brought to the semi-explicit form

$$\dot{x} = y, \qquad 0 = F(x, y, t),$$

by the single substitution $y = \dot{x}$. The transformation is free, and it is worth making because the semi-explicit form exposes immediately which equations are differential and which are algebraic — and, as this chapter shows, allows the index to be determined by inspecting which variables each algebraic equation depends on, without performing any reduction.

10.1 From an Implicit DAE to Semi-Explicit Form

An implicit differential-algebraic equation (DAE) is written as

$$F\big(x(t), x'(t), t\big) = 0, \qquad x(0) = x_0,$$

where $x(t) \in \mathbb{R}^n$. If the Jacobian $F_{x'}$ is nonsingular at a solution point, the implicit function theorem gives, locally,

$$x'(t) = f\big(x(t), t\big).$$

The system is then locally an ODE. When $F_{x'}$ is singular, the equation cannot in general be solved for the full derivative; the remaining relations constrain the state and lead to a DAE. Singular Jacobians alone do not determine an index: the derivative constraints and their rank conditions also matter.

10.1.1 Semi-explicit form

A common structure separates differential variables $x(t) \in \mathbb{R}^n$ from algebraic variables $z(t) \in \mathbb{R}^m$:

$$\begin{aligned} x'(t) &= f\big(x(t), z(t), t\big), \\ 0 &= g\big(x(t), z(t), t\big). \end{aligned}$$

The first equation supplies the differential dynamics; the second imposes an algebraic constraint. This is called **semi-explicit form**. The variables z need not be constant: their values are determined at each time by the constraint and the evolving differential state.

10.1.2 From an implicit DAE to semi-explicit form

If the equations can be partitioned so that some equations explicitly define the derivatives of selected variables and the remaining equations contain no derivatives, they can be arranged in semi-explicit form by reordering variables and equations. For a general equation

$$F\big(x(t), x'(t), t\big) = 0,$$

one can also introduce an auxiliary variable $v = x'$ and write

$$x' = v,$$
$$0 = F(x, v, t).$$

This is a semi-explicit representation in the broad sense, but it may not be the usual differential/algebraic split with an invertible algebraic Jacobian. Under the regularity assumptions for this order-lifting construction, it typically raises the differentiation index by one. It is therefore not an index-preserving simplification in general.

10.1.3 The index-one Jacobian condition

For the semi-explicit system, suppose g is continuously differentiable and the square Jacobian

$$g_z(x, z, t) = \frac{\partial g}{\partial z}$$

is nonsingular at a point satisfying the constraint. The implicit function theorem then gives a local function $z = \phi(x, t)$ such that $g(x, \phi(x, t), t) = 0$. Substitution reduces the differential equation locally to

$$x'(t) = f\big(x(t), \phi(x(t), t), t\big).$$

This regular semi-explicit case is called **index one** under the differentiation-index convention used in the next chapter. If g_z is singular, this argument does not apply; one may need to differentiate constraints to reveal additional relations. The definition and worked calculations are given in Section 9.1.

Theorem 10.1.1 — Implicit Function Theorem. Let $g : U \subset \mathbb{R}^{n+m+1} \to \mathbb{R}^m$ be continuously differentiable, and let $(x_0, z_0, t_0) \in U$ satisfy $g(x_0, z_0, t_0) = 0$. If $g_z(x_0, z_0, t_0)$ is nonsingular, then there are neighborhoods of (x_0, t_0) and z_0 and a unique continuously differentiable function $z = \phi(x, t)$ such that, locally,

$$g(x, z, t) = 0 \quad \Longleftrightarrow \quad z = \phi(x, t).$$

Its derivatives satisfy

$$\phi_x = -g_z^{-1} g_x, \qquad \phi_t = -g_z^{-1} g_t,$$

with Jacobians evaluated along $z = \phi(x, t)$.

10.1.4 Example: an index-one system

Consider

$$x'(t) = -y(t),$$
$$0 = x(t) + y(t) - \sin t.$$

Here $g_y = 1$, so the constraint gives $y = \sin t - x$. The system reduces to

$$x'(t) = x(t) - \sin t, \qquad y(t) = \sin t - x(t).$$

Equivalently, differentiating the constraint once gives $x' + y' - \cos t = 0$, and using $x' = -y$ yields $y' = y + \cos t$. Thus one differentiation supplies an explicit ODE for (x, y), in agreement with the index-one Jacobian test.

10.2 Classification of Semi-Explicit DAEs

Once a DAE has been written in the semi-explicit form of Section 10.1, its index can be read off from **which variables the algebraic equations depend on**, without performing the reduction. The mechanism is always the same: differentiate the constraint, look for the algebraic variable in the result, and if it is absent differentiate again.

Throughout this section $\boldsymbol{x}$ denotes the differential variables and $\boldsymbol{y}$ the algebraic ones, and all Jacobians are assumed to exist and to be continuous.

10.2.1 Index one

The constraint contains the algebraic variable:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, t), \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, t). \end{aligned} \tag{10.1}$$

Differentiating the constraint once and substituting $\dot{\boldsymbol{x}} = \boldsymbol{f}$,

$$\boldsymbol{0} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\dot{\boldsymbol{x}} + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{y}}\dot{\boldsymbol{y}} + \frac{\partial \boldsymbol{g}}{\partial t} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\boldsymbol{f} + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{y}}\dot{\boldsymbol{y}} + \frac{\partial \boldsymbol{g}}{\partial t}.$$

Definition 10.2.1 — Index-one condition. Equation 10.1 is a DAE of index one if

$$\boxed{\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{y}}(\boldsymbol{x}, \boldsymbol{y}, t) \quad \text{is nonsingular.}} \tag{10.2}$$

Under Equation 10.2 the derivative of the constraint can be solved for $\dot{\boldsymbol{y}}$,

$$\dot{\boldsymbol{y}} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}, t) := -\left(\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{y}}\right)^{-1}\left(\frac{\partial \boldsymbol{g}}{\partial t} + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\boldsymbol{f}\right),$$

and one differentiation suffices:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, t), \\ \dot{\boldsymbol{y}} &= \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}, t), \end{aligned} \qquad \oplus \qquad \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, t) = \boldsymbol{0} \ \text{(invariant)}.$$

Index one and coordinate partitioning

By the implicit function theorem (Theorem 10.1.1), condition Equation 10.2 also guarantees the existence of $\boldsymbol{y} = \boldsymbol{\varphi}(\boldsymbol{x}, t)$ with $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\varphi}(\boldsymbol{x}, t), t) \equiv \boldsymbol{0}$, so that Equation 10.1 becomes the genuine ODE $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\varphi}(\boldsymbol{x}, t), t)$ on the constraint manifold. This is the trivial case of the **coordinate partitioning** technique of Coordinate Partitioning. It is of limited practical use, because $\boldsymbol{\varphi}$ is rarely available in closed form; the derivative formula above is what an integrator actually uses.

10.2.2 Index two

The algebraic variable disappears from the constraint:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, t), \\ \mathbf{0} &= \boldsymbol{g}(\boldsymbol{x}, t). \end{aligned} \tag{10.3}$$

Now $\partial \boldsymbol{g}/\partial \boldsymbol{y} = \mathbf{0}$ identically, so Equation 10.2 fails and one differentiation cannot produce $\dot{\boldsymbol{y}}$. It produces instead a **new algebraic equation**,

$$\mathbf{0} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{x}, t)\, \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, t) + \frac{\partial \boldsymbol{g}}{\partial t}(\boldsymbol{x}, t) =: \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}, t), \tag{10.4}$$

in which $\boldsymbol{y}$ **does** appear, through $\boldsymbol{f}$. The pair $\{\dot{\boldsymbol{x}} = \boldsymbol{f},\ \mathbf{0} = \boldsymbol{h}\}$ has the shape of Equation 10.1, and is therefore of index one exactly when $\partial \boldsymbol{h}/\partial \boldsymbol{y}$ is nonsingular. Differentiating Equation 10.4 with respect to $\boldsymbol{y}$ gives

$$\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{y}} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}},$$

the first instance of the pattern that governs the whole classification.

Definition 10.2.2 — Index-two condition. Equation 10.3 is a DAE of index two if

$$\boxed{\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{x}, t) \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}}(\boldsymbol{x}, \boldsymbol{y}, t) \quad \text{is nonsingular.}} \tag{10.5}$$

Two differentiations are then needed, and the reduced ODE carries **two** invariants: the original constraint $\boldsymbol{g} = \mathbf{0}$ and its first derivative $\boldsymbol{h} = \mathbf{0}$. Note that each is weaker than the previous one: a numerical trajectory can satisfy $\boldsymbol{h} = \mathbf{0}$ to machine precision and still drift away from $\boldsymbol{g} = \mathbf{0}$.

10.2.3 Index three

The constraint reaches the algebraic variable only through two intermediate differential equations:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, t), \\ \dot{\boldsymbol{y}} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, t), \\ \mathbf{0} &= \boldsymbol{h}(\boldsymbol{y}, t), \end{aligned} \tag{10.6}$$

with $\boldsymbol{z}$ algebraic. Differentiating the constraint,

$$\mathbf{0} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{y}} \dot{\boldsymbol{y}} + \frac{\partial \boldsymbol{h}}{\partial t} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{y}} \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, t) + \frac{\partial \boldsymbol{h}}{\partial t} =: \boldsymbol{a}(\boldsymbol{x}, \boldsymbol{y}, t), \tag{10.7}$$

which still does not contain $\boldsymbol{z}$. Collecting $\boldsymbol{w} = (\boldsymbol{x}, \boldsymbol{y})$ and $\boldsymbol{F}(\boldsymbol{w}, \boldsymbol{z}, t) = \big(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, t),\ \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, t)\big)$, the system

$$\dot{\boldsymbol{w}} = \boldsymbol{F}(\boldsymbol{w}, \boldsymbol{z}, t), \qquad \mathbf{0} = \boldsymbol{a}(\boldsymbol{w}, t)$$

is precisely Equation 10.3, hence of index two when $(\partial \boldsymbol{a}/\partial \boldsymbol{w})(\partial \boldsymbol{F}/\partial \boldsymbol{z})$ is nonsingular. Since

$$\frac{\partial \boldsymbol{a}}{\partial \boldsymbol{w}} = \left(\frac{\partial \boldsymbol{a}}{\partial \boldsymbol{x}}, \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{y}} \right), \qquad \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{z}} = \begin{pmatrix} \dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{z}} \\ \mathbf{0} \end{pmatrix}, \qquad \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}},$$

the product collapses to a chain of three Jacobians.

Definition 10.2.3 — Index-three condition. Equation 10.6 is a DAE of index three if

$$\boxed{\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{y}}(\boldsymbol{y}, t)\,\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{y}, t)\,\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{z}}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, t) \quad \text{is nonsingular.}} \tag{10.8}$$

10.2.4 Hessenberg form

The three conditions Equation 10.2, Equation 10.5 and Equation 10.8 are the first terms of one chain. The systems for which this is exactly true have a name.

Definition 10.2.4 — Hessenberg form of size r. A semi-explicit DAE is in **Hessenberg form of size** r if it can be written

$$\begin{aligned} \dot{\boldsymbol{x}}_i &= \boldsymbol{f}_i(\boldsymbol{x}_1, \dots, \boldsymbol{x}_{i+1}, t), \quad i = 1, \dots, r-1, \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}_1, t), \end{aligned} \tag{10.9}$$

that is, each differential equation may depend only on the variables up to the next block, the constraint depends only on $\boldsymbol{x}_1$, and the chain of Jacobians

$$\boxed{\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}_1}\frac{\partial \boldsymbol{f}_1}{\partial \boldsymbol{x}_2}\frac{\partial \boldsymbol{f}_2}{\partial \boldsymbol{x}_3}\cdots\frac{\partial \boldsymbol{f}_{r-1}}{\partial \boldsymbol{x}_r} \quad \text{is nonsingular.}} \tag{10.10}$$

Theorem 10.2.1 — Index of a Hessenberg DAE. A DAE in Hessenberg form of size r has differentiation index exactly r.

The name refers to the staircase sparsity of the Jacobian of the whole system with respect to the derivatives: each block couples only to its neighbour, exactly as in a Hessenberg matrix. Setting $r = 2$ in Equation 10.10 reproduces Equation 10.5 and $r = 3$ reproduces Equation 10.8, with the identification $\boldsymbol{x}_1 = \boldsymbol{y}$, $\boldsymbol{x}_2 = \boldsymbol{x}$, $\boldsymbol{x}_3 = \boldsymbol{z}$ in the latter. Index one is the degenerate case in which the constraint already contains the algebraic variable, $\boldsymbol{0} = \boldsymbol{g}(\boldsymbol{x}_1, \boldsymbol{x}_2, t)$ with $\partial \boldsymbol{g}/\partial \boldsymbol{x}_2$ nonsingular.

! Why Hessenberg form matters in practice

Hessenberg form is not a curiosity: it is the structure of constrained mechanics, and therefore of most of the DAEs that arise in this course. For a mechanical system with generalized coordinates $\boldsymbol{q}$, mass matrix $\boldsymbol{M}(\boldsymbol{q})$, applied forces $\boldsymbol{F}$, holonomic constraints $\boldsymbol{\Phi}(\boldsymbol{q}) = \boldsymbol{0}$ and constraint Jacobian $\boldsymbol{G} = \partial \boldsymbol{\Phi}/\partial \boldsymbol{q}$, the equations of motion are

$$\dot{\boldsymbol{q}} = \boldsymbol{v}, \qquad \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{v}} = \boldsymbol{F}(\boldsymbol{q}, \boldsymbol{v}, t) - \boldsymbol{G}(\boldsymbol{q})^T \boldsymbol{\lambda}, \qquad \boldsymbol{0} = \boldsymbol{\Phi}(\boldsymbol{q}),$$

which is Equation 10.9 with $r = 3$, $\boldsymbol{x}_1 = \boldsymbol{q}$, $\boldsymbol{x}_2 = \boldsymbol{v}$, $\boldsymbol{x}_3 = \boldsymbol{\lambda}$. The chain Equation 10.10 becomes

$$\boldsymbol{G}\,\boldsymbol{I}\left(-\boldsymbol{M}^{-1}\boldsymbol{G}^T\right) = -\boldsymbol{G}\boldsymbol{M}^{-1}\boldsymbol{G}^T,$$

nonsingular precisely when $\boldsymbol{M}$ is positive definite and $\boldsymbol{G}$ has full row rank, that is when the mass matrix is physical and the constraints are independent. **Every holonomically constrained mechanical system is an index-three DAE**, and the two conditions that

make it so are the two conditions a modeller checks anyway.

Knowing r in advance is worth the effort: it tells how many differentiations the reduction of Chapter 11 will need, how many invariants the reduced ODE will carry, and — as Order Reduction in DAE Integration shows — how much accuracy a Runge–Kutta method loses on the algebraic components.

10.3 Can a Runge–Kutta Method Be Applied Directly?

Before developing index reduction it is worth asking whether it is needed at all. A Runge–Kutta method can be written for a fully implicit ODE, and the fully implicit DAE $\boldsymbol{F}(\boldsymbol{x}, \dot{\boldsymbol{x}}, t) = \boldsymbol{0}$ looks like one. The answer is instructive: the formulas exist, they can be implemented, and they do not solve the problem.

10.3.1 Implicit form of a Runge–Kutta method

For the explicit ODE $\dot{\boldsymbol{x}} = \boldsymbol{g}(\boldsymbol{x}, t)$, an s-stage Runge–Kutta method advances as

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \sum_{j=1}^{s} b_j \boldsymbol{K}_j, \qquad \boldsymbol{K}_j = \Delta t\, \boldsymbol{g}\Big(\boldsymbol{x}_k + \sum_{\ell=1}^{s} a_{j\ell} \boldsymbol{K}_\ell,\ t_k + c_j \Delta t\Big),$$

so that each stage satisfies $\boldsymbol{K}_j / \Delta t = \boldsymbol{g}(\,\cdot\,)$. The quantity $\boldsymbol{K}_j/\Delta t$ is therefore an approximation of the derivative at the stage point. If the equation is given implicitly as $\boldsymbol{F}(\boldsymbol{x}, \dot{\boldsymbol{x}}, t) = \boldsymbol{0}$, the same stage conditions read

$$\boldsymbol{F}\Big(\boldsymbol{x}_k + \sum_{\ell=1}^{s} a_{j\ell} \boldsymbol{K}_\ell,\ \frac{\boldsymbol{K}_j}{\Delta t},\ t_k + c_j \Delta t\Big) = \boldsymbol{0}, \qquad j = 1, \dots, s, \tag{10.11}$$

a nonlinear system of $s\,n$ equations in the $s\,n$ unknowns $\boldsymbol{K}_1, \dots, \boldsymbol{K}_s$, to be solved with the Newton iteration of Solving the Stage Equations. Nothing in Equation 10.11 requires $\partial \boldsymbol{F} / \partial \dot{\boldsymbol{x}}$ to be invertible.

10.3.2 Why this is not enough

Equation 10.11 can be written down for any $\boldsymbol{F}$, which makes it tempting to conclude that any DAE can be integrated by an implicit Runge–Kutta method. The conclusion is false, for three separate reasons.

- **The problem may have no solution at all.** A DAE can be inconsistent, as the singular-pencil example of Section 8.4 shows: there, no differentiable $\boldsymbol{x}$ satisfies the equations for the given forcing. A convergent iteration on Equation 10.11 cannot produce one.
- **The initial condition may be inconsistent.** Even for a solvable DAE, an arbitrary $\boldsymbol{x}_k$ need not satisfy the hidden constraints, and then Equation 10.11 has no solution near $\boldsymbol{K}_j = \boldsymbol{0}$. This is why consistent initialization (Consistent Initialization) is a separate step and not a detail.
- **The accuracy is not the one advertised.** Even when everything is consistent, the classical order of a Runge–Kutta method does not carry over to the algebraic components of a high-index DAE. The loss depends on the index and on the method (is the subject of Order Reduction in DAE Integration).

The practical conclusion is the one that organizes the rest of this part:

Rather than applying a general-purpose solver to $\boldsymbol{F}(\boldsymbol{x}, \dot{\boldsymbol{x}}, t) = \boldsymbol{0}$ and hoping, one first determines the index, reduces it to one or zero, records the hidden constraints as invariants, and only then integrates — with a method chosen in view of the remaining structure.

10.4 Examples

An index-three nonlinear system

Example from [15] equation (3.1)–(3.3)

$$\begin{cases} y' = t - x \\ y' + z' = 1 - (\alpha x + \eta) y \\ 0 = y(2 - y) + 2z \end{cases}$$

10.4.0.1 Semi-Explicit Form

For the previous DAE system introduce dummy variable x_{dot} and the system decomposes into differential (blue) and algebraic (red) components:

$$\begin{cases} x' = x_{\text{dot}} \\ y' = t - x \\ z' = 1 - t + x - (\alpha x + \eta) y \end{cases} \bigoplus \quad 0 = y(1 - y/2) + z \qquad \text{(Constraint 1)}$$

For this DAE we have:

- **Differential variables**: x, y, z.
- **Algebraic variables**: x_{dot}.

10.4.0.2 Reduction Procedure

We attempt to solve for x_{dot}, but the system is not yet solvable explicitly. We proceed by **differentiating**.

Step 1: Differentiate Constraints

Differentiate the constraint

$$\begin{aligned} (\text{Constraint 1})' &= 1 - yy' + z' \\ &= 1 - y(t - x) + 1 - t + x - (\alpha x + \eta) y \\ &= 2 + (x - t)(y + 1) - (\alpha x + \eta) y \end{aligned}$$

so that the DAE become:

$$\begin{cases} x' = x_{\text{dot}} \\ y' = t - x \\ z' = 1 - t + x - (\alpha x + \eta) y \end{cases} \bigoplus \begin{cases} 0 = 2 + (x - t)(y + 1) - (\alpha x + \eta) y & (\text{Constraint 1})' \\ 0 = y(1 - y/2) + z & (\text{Constraint 1}) \end{cases}$$

The red algebraic equations are the new algebraic equation after the derivation. The green algebraic equation is the hidden constraints that is removed from the DAE system but that is satisfied by choosing the right initial conditions.

The red part cannot determine x_{dot} the algebraic variable must be differentiated again.

Step 2: Differentiate Again

$$\begin{aligned}(\text{Constraint 1})'' &= (x'-1)(y+1)+(x-t)y'-\alpha(x'y+xy')+\eta y' \\ &= (x_{\text{dot}}-1)(y+1)+(x-t)(t-x)-\alpha(x_{\text{dot}}y+x(t-x))+\eta(t-x) \\ &= x_{\text{dot}}(y(1-\alpha)+1)-y-1-(x-t)^2+(\alpha x+\eta)(t-x)=0\end{aligned}$$

The dummy variable x_{dot} can be solved:

$$x_{\text{dot}}=\frac{y+1+(x-t+\alpha x+\eta)(x-t)}{y(1-\alpha)+1}$$

and the DAE is reduced to an index-1 DAE. Having done two derivation of the constraint the original DAE was of **index 3**. Removing the dummy variable we can obtain an ODE with invariants:

$$\begin{cases} x'=\dfrac{y+1+(x-t+\alpha x+\eta)(x-t)}{y(1-\alpha)+1} \\ y'=t-x \\ z'=1-t+x-(\alpha x+\eta)y \end{cases} \oplus \begin{cases} 0=2+(x-t)(y+1)-(\alpha x+\eta)y & (\text{Constraint 1})' \\ 0=y(1-y/2)+z & (\text{Constraint 1}) \end{cases}$$

10.4.0.3 Baumgarte Stabilization

As usual we want to stabilize the constraints to avoid the drift on the hidden constraints:

$$(\text{Constraints 1})''+2\eta\omega(\text{Constraints 1})'+\omega^2(\text{Constraints 1})=0$$

and substituting the expression

$$\begin{aligned}&x_{\text{dot}}(y(1-\alpha)+1)-y-1-(x-t)^2+(\alpha x+\eta)(t-x) \\ &+2\eta\omega(2+(x-t)(y+1)-(\alpha x+\eta)y) \\ &+\omega^2(y(1-y/2)+z)=0\end{aligned}$$

so that

$$\begin{aligned}x_{\text{dot}} &= f(x,y,z,t) \\ f(x,y,z,t) &= \frac{a(x,y,z,t)-2\eta\omega b(x,y,z,t)-\omega^2 c(x,y,z,t)}{y(1-\alpha)+1} \\ a(x,y,z,t) &= y+1+(x-t+\alpha x+\eta)(x-t) \\ b(x,y,z,t) &= 2+(x-t)(y+1)-(\alpha x+\eta)y \\ c(x,y,z,t) &= y(1-y/2)+z\end{aligned}$$

and the stabilized ODE become

$$\begin{cases} x'=f(x,y,z,t) \\ y'=t-x \\ z'=1-t+x-(\alpha x+\eta)y \end{cases} \oplus \begin{cases} 0=2+(x-t)(y+1)-(\alpha x+\eta)y & (\text{Constraint 1})' \\ 0=y(1-y/2)+z & (\text{Constraint 1}) \end{cases}$$

11. General Index Reduction

This chapter treats the general case. The object is a DAE written as

$$E(x, t)\,\dot{x} = G(x, t),$$

which covers every semi-explicit system of Chapter 10 and the constant-coefficient systems of Chapter 7, and the goal is the one stated in Chapter 8, now for a state-dependent E:

$$\text{DAE} \xrightarrow{\text{differentiation}} \text{ODE or index-1 DAE} \oplus \text{invariants}.$$

The algorithm is the same loop — separate, file, differentiate — but the separation now requires a state-dependent transformation, and constructing it is where the linear algebra enters.

11.1 The General Form $E(x, t)\dot{x} = G(x, t)$

The semi-explicit transformation of Section 10.1 always produces a system of the shape

$$E(x, t)\,\dot{x} = G(x, t), \qquad E : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{n\times n}, \tag{11.1}$$

with $E = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ in the semi-explicit case. There is no reason to keep that particular E: the reduction algorithm below works for an arbitrary state- and time-dependent E, and after one step the identity block is destroyed anyway. Taking Equation 11.1 as the object of study therefore costs nothing and covers all the semi-explicit cases of Section 10.2 at once, as well as the constant-coefficient DAE $E\dot{x} = Ax + f(t)$ of Chapter 7.

Three cases, exactly parallel to the linear ones.

$E(x, t)$ **nonsingular.** Then

$$\dot{x} = E(x, t)^{-1} G(x, t)$$

is an ODE, or equivalently a DAE of index zero. In a code the inverse is never formed: one factorizes E and solves at each evaluation.

$E(x,t) \equiv 0$. The system degenerates to the purely algebraic equation $0 = G(x,t)$. Whether this can be turned into an ODE depends on G: if $\partial G/\partial x$ is nonsingular, differentiating

$$\frac{\mathrm{d}}{\mathrm{d}t} G(x,t) = \frac{\partial G}{\partial x}(x,t)\,\dot{x} + \frac{\partial G}{\partial t}(x,t) = 0$$

gives the ODE

$$\dot{x} = -\left(\frac{\partial G}{\partial x}(x,t)\right)^{-1} \frac{\partial G}{\partial t}(x,t), \tag{11.2}$$

whose solutions include those of $G = 0$. As always, the original equation survives as an invariant and fixes the initial condition: $x(0)$ is not free but must solve $G(x(0),0) = 0$.

$E(x,t)$ **singular but not zero.** This is the genuine DAE case and the subject of the rest of the chapter. Note that the rank of E may depend on x and t: a system can change index along a trajectory, and every statement below is local.

11.2 Separating the Algebraic Part with a Generalized LU

The reduction algorithm of Section 11.3 needs, at every step, a nonsingular $T(x,t)$ that moves all the zero rows of $E(x,t)$ to the bottom. This section constructs it, and the construction is nothing more than Gaussian elimination with pivoting applied to a possibly rectangular, possibly singular matrix.

11.2.1 LU with partial and full pivoting

For a nonsingular $A \in \mathbb{R}^{n\times n}$, partial pivoting produces

$$PA = LU,$$

with P a row permutation, L unit lower triangular and U upper triangular. Allowing column interchanges as well gives full pivoting,

$$PAQ = LU, \tag{11.3}$$

with Q a column permutation. The point for our purposes is that **Equation 11.3 does not require A to be square or nonsingular**: the elimination of Appendix A stops when no nonzero pivot remains, and what it leaves is

$$PAQ = LU, \qquad U = \begin{pmatrix} U_1 & N \\ 0 & 0 \end{pmatrix},$$

with $U_1 \in \mathbb{R}^{k\times k}$ upper triangular and nonsingular, $k = \operatorname{rank} A$. The precise statement, with the block factorization of both factors, is Theorem 8.2.1.

11.2.2 The form actually used: $PA = LM$

For index reduction the column permutation is a nuisance: permuting columns permutes the unknowns, and we want to touch only the equations. It can be absorbed.

Theorem 11.2.1 — Row-only rank-revealing factorization. Let $A \in \mathbb{R}^{m\times n}$ have rank k. Then there exist a permutation P, a unit lower triangular L and a matrix M with

$$\boxed{\boldsymbol{P}\boldsymbol{A} = \boldsymbol{L}\boldsymbol{M}, \qquad \boldsymbol{M} = \begin{pmatrix} \boldsymbol{M}_1 \\ \boldsymbol{0} \end{pmatrix}, \qquad \boldsymbol{M}_1 \in \mathbb{R}^{k \times n} \text{ of full row rank.}} \tag{11.4}$$

Proof. Apply Equation 11.3 to obtain $\boldsymbol{P}\boldsymbol{A}\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{U}$. Since $\boldsymbol{Q}$ is a permutation, $\boldsymbol{Q}^{-1} = \boldsymbol{Q}^T$ and

$$\boldsymbol{P}\boldsymbol{A} = \boldsymbol{L}\boldsymbol{U}\boldsymbol{Q}^T = \boldsymbol{L}\big(\boldsymbol{U}\boldsymbol{Q}^T\big).$$

Right-multiplication by $\boldsymbol{Q}^T$ permutes the columns of $\boldsymbol{U}$ and therefore cannot create a nonzero entry in a zero row: the last $m - k$ rows of $\boldsymbol{U}\boldsymbol{Q}^T$ are still zero, and its first k rows have full row rank because $\boldsymbol{U}_1$ is nonsingular. Set $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{Q}^T$. □ ■

11.2.3 The separator $\boldsymbol{L}^{-1}\boldsymbol{P}$

Apply Equation 11.4 to the matrix $\boldsymbol{E}(\boldsymbol{x}, t)$ of Equation 11.1:

$$\boldsymbol{P}\boldsymbol{E} = \boldsymbol{L}\boldsymbol{M} \qquad \Longrightarrow \qquad \boldsymbol{L}^{-1}\boldsymbol{P}\,\boldsymbol{E} = \begin{pmatrix} \boldsymbol{M}_1 \\ \boldsymbol{0} \end{pmatrix}. \tag{11.5}$$

Multiplying the DAE on the left by $\boldsymbol{L}^{-1}\boldsymbol{P}$ therefore splits it into a differential and an algebraic block without changing its solution set:

$$\boxed{\boldsymbol{T}(\boldsymbol{x}, t) = \boldsymbol{L}(\boldsymbol{x}, t)^{-1}\boldsymbol{P}(\boldsymbol{x}, t) \quad \text{is the matrix that separates the algebraic part of a DAE.}} \tag{11.6}$$

This answers the question left open in Section 11.3. The factorization is recomputed at each reduction step, because $\boldsymbol{E}$ changes; $\boldsymbol{L}^{-1}$ is of course never formed, the product $\boldsymbol{L}^{-1}\boldsymbol{P}\boldsymbol{G}$ being obtained by a forward substitution on the permuted right-hand side.

> ⚠ The permutation is not cosmetic
>
> $\boldsymbol{P}$ depends on $\boldsymbol{x}$ and t through the pivot choice. A pivot that is nonzero but small makes $\boldsymbol{L}$ badly conditioned, and the "zero" rows of $\boldsymbol{M}$ are zero only to within a tolerance. The separation between differential and algebraic equations is therefore a **numerical rank decision**, with all that this implies: see Section 8.2.4. Near a point where the rank of $\boldsymbol{E}$ changes, no factorization can make the choice for you.

11.3 Nonlinear Index Reduction

This tutorial explains how to compute the index of a non-linear DAE systematically.

11.3.1 General DAE

A general **Differential-Algebraic equation (DAE)** takes the form:

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{x}', t) = \boldsymbol{0},$$

where

- $\boldsymbol{x}$ represents the state variables
- $\boldsymbol{x}'$ denotes their derivatives with respect to time t
- $\boldsymbol{f}$ is a system of equations involving both algebraic and differential components.

The defining characteristic of a DAE is that the Jacobian matrix

$$\frac{\partial f(x, x', t)}{\partial x'}$$

is **singular**, meaning that the system does not explicitly define all derivatives in terms of the state variables. This is in contrast to **ordinary differential equations (ODEs)**, where the system can be solved explicitly for x'.

11.3.2 Semi-Explicit Form

Introducing the auxiliary variable $y = x'$, the DAE can be rewritten as the system:

$$\begin{cases} x' = y, \\ 0 = f(x, y, t). \end{cases}$$

This is a special case of the **semi-explicit DAE** of the form:

$$\begin{cases} x' = g(x, y, t), \\ 0 = f(x, y, t), \end{cases}$$

where g defines an explicit relationship for the differential variables.

11.3.3 Matrix Form

The semi-explicit DAE system can be expressed in a compact **matrix form**:

$$\underbrace{\begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}}_{\mathcal{E}(z,t)} \underbrace{\begin{pmatrix} x' \\ y' \end{pmatrix}}_{\boldsymbol{\varkappa}(t)} = \underbrace{\begin{pmatrix} g(x, y, t) \\ f(x, y, t) \end{pmatrix}}_{\mathcal{G}(z,t)}.$$

This formulation aligns with the **general form of a DAE system**:

$$\mathcal{E}(z, t)\, z(t) = \mathcal{G}(z, t),$$

where $\mathcal{E}(z, t)$ is a singular matrix, reinforcing the differential-algebraic nature of the system.

11.3.4 Reduction Algorithm

The **differential index** of a differential-algebraic equation (DAE) provides insight into the complexity of solving the system. The following steps outline an approach to determine the index and perform index reduction when necessary.

Step 1: Build the Matrix Form

A nonlinear DAE can be written as:

$$E(x, t)x' = g(x, t), \tag{11.7}$$

where:

- $E(x, t)$ is an $n \times n$ matrix,
- $x(t)$ represents the state variables,
- x' is the time derivative of x,
- $g(x, t)$ is a given function.

This formulation allows us to analyze the structure of the system and determine if it contains purely algebraic constraints.

Step 2: Check Singularity

Compute the determinant:

- If $\det(\boldsymbol{E}(\boldsymbol{x}, t)) \neq 0$, then $\boldsymbol{E}(\boldsymbol{x}, t)$ is invertible, and the system reduces to an **ODE (index 0)**. **Terminate the procedure.**
- If $\det(\boldsymbol{E}(\boldsymbol{x}, t)) = 0$, then $\boldsymbol{E}(\boldsymbol{x}, t)$ is singular, indicating that algebraic constraints are present. **Proceed to the next step.**

R [Constant-Rank Assumption] We assume that $\boldsymbol{E}(\boldsymbol{x}, t)$ has constant rank in the domain of interest $(\boldsymbol{x}, t)$. If the rank varies, numerical methods must adapt accordingly, and the system's index may change dynamically.

Step 3: Decompose the System

Determine the **rank** of $\boldsymbol{E}(\boldsymbol{x}, t)$, denoted as $r = \operatorname{rank}(\boldsymbol{E})$. Define:

- A **basis** $\boldsymbol{Q}_a(\boldsymbol{x}, t) \in \mathbb{R}^{n\times(n-k)}$ for the **null space** $\mathcal{N}(\boldsymbol{E}(\boldsymbol{x}, t)^T)$, which identifies the algebraic constraints. This basis satisfies:

$$\boldsymbol{E}(\boldsymbol{x}, t)^T \boldsymbol{Q}_a(\boldsymbol{x}, t) = \boldsymbol{0}, \quad \text{or equivalently,} \quad \boldsymbol{Q}_a(\boldsymbol{x}, t)^T \boldsymbol{E}(\boldsymbol{x}, t) = \boldsymbol{0}.$$

- A matrix $\boldsymbol{Q}_d(\boldsymbol{x}, t) \in \mathbb{R}^{n\times k}$ that completes $\boldsymbol{Q}_a(\boldsymbol{x}, t)$ to a full basis, ensuring that:

$$\boldsymbol{Q}(\boldsymbol{x}, t) = \big(\boldsymbol{Q}_d(\boldsymbol{x}, t), \boldsymbol{Q}_a(\boldsymbol{x}, t)\big)$$

  is non-singular.

By multiplying the original system by $\boldsymbol{Q}(\boldsymbol{x}, t)^T$, we obtain:

- **Differential part:**

$$\boldsymbol{E}_d(\boldsymbol{x}, t)\boldsymbol{x}' = \boldsymbol{g}_d(\boldsymbol{x}, t),$$

  where:

$$\boldsymbol{E}_d(\boldsymbol{x}, t) = \boldsymbol{Q}_d(\boldsymbol{x}, t)^T \boldsymbol{E}(\boldsymbol{x}, t), \quad \boldsymbol{g}_d(\boldsymbol{x}, t) = \boldsymbol{Q}_d^T(\boldsymbol{x}, t)\boldsymbol{g}(\boldsymbol{x}, t).$$

- **Algebraic part:**

$$\boldsymbol{g}_a(\boldsymbol{x}, t) = \boldsymbol{0},$$

  where:

$$\boldsymbol{g}_a(\boldsymbol{x}, t) = \boldsymbol{Q}_a(\boldsymbol{x}, t)^T \boldsymbol{g}(\boldsymbol{x}, t).$$

  Since $\boldsymbol{Q}_a(\boldsymbol{x}, t)^T \boldsymbol{E}(\boldsymbol{x}, t) = \boldsymbol{0}$, no derivatives appear in this equation.

Step 4: Differentiate Constraints

The algebraic constraints must be differentiated to reveal hidden differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{g}_a(\boldsymbol{x}, t) = \boldsymbol{g}_a^t(\boldsymbol{x}, t) + \boldsymbol{g}_a^x(\boldsymbol{x}, t)\boldsymbol{x}'(t) = \boldsymbol{0}.$$

Rewriting, the system becomes:

$$\begin{pmatrix} \boldsymbol{E}_d(\boldsymbol{x}, t) \\ -\boldsymbol{g}_a^x(\boldsymbol{x}, t) \end{pmatrix} \boldsymbol{x}' = \begin{pmatrix} \boldsymbol{g}_d(\boldsymbol{x}, t) \\ \boldsymbol{g}_a^t(\boldsymbol{x}, t) \end{pmatrix}.$$

Defining:

$$\widetilde{\boldsymbol{E}}(\boldsymbol{x}, t) = \begin{pmatrix} \boldsymbol{E}_d(\boldsymbol{x}, t) \\ -\boldsymbol{g}_a^x(\boldsymbol{x}, t) \end{pmatrix}, \quad \widetilde{\boldsymbol{g}}(\boldsymbol{x}, t) = \begin{pmatrix} \boldsymbol{g}_d(\boldsymbol{x}, t) \\ \boldsymbol{g}_a^t(\boldsymbol{x}, t) \end{pmatrix},$$

we obtain a new DAE system:

$$\widetilde{\boldsymbol{E}}(\boldsymbol{x}, t)\boldsymbol{x}' = \widetilde{\boldsymbol{g}}(\boldsymbol{x}, t).$$

Step 5: Repeat to Nonsingularity

Repeat **Steps 2–4** until $\widetilde{E}(x, t)$ is invertible. The **differentiation index** is the number of times Step 4 is performed before reaching an ODE system.

11.4 Alternative Constructions and Which to Use

In the index reduction process, to separate the differential and algebraic components, we need to compute the kernel (null space) of E, as it represents the algebraic constraints. There are different approaches available to compute it:

- **LU decomposition/factorization**: a decomposition that factors a matrix A as the product of two matrices, $A = LU$, where L is a lower triangular matrix and U is an upper triangular matrix;
- **Singular Value Decomposition (SVD)**: a factorization that factors a matrix as the product of three matrices $A = UDV^T$ where the columns of U and V are orthonormal and the matrix D is diagonal with positive real entries, to analyze matrix rank and separate the space into differential and algebraic components.

11.4.1 Kernel-Based Reduction

When the matrix E is singular, there exists a nonzero vector $v \neq 0$ such that

$$Ev = 0.$$

All such vectors form a linear subspace called the *kernel* of E, denoted as Kernel(E). This subspace can be spanned by a set of basis vectors $\{v_i\}$. Similarly, there exists a nonzero vector $w \neq 0$ such that

$$w^T E = 0^T,$$

which defines the kernel of the transpose of E, denoted Kernel(E^T). This subspace is spanned by basis vectors $\{w_i\}$.

We focus on the kernel of the transpose, cokernel,

$$\text{Kernel}(E^T) = \{w \mid w^T E = 0\} = \text{Span}\{w_1, w_2, \dots, w_p\} = K.$$

Denoting $K = (w_1, w_2, \dots, w_p)$, it follows that

$$K^T E = 0.$$

If we can compute this kernel matrix $K(t, y)$ for the time- and state-dependent matrix $E(t, y)$, we can identify an invariant condition for the DAE.

Consider a general the DAE system Equation 11.7, and define

$$K(t, y) = \text{Kernel}(E(t, y)^T) = \text{Cokernel}(E).$$

Multiplying the original DAE by $K(t, y)^T$, we get:

$$K(t, y)^T E(t, y) y' = K(t, y)^T G(t, y).$$

Since $K^T E = 0$, the left side vanishes, yielding

$$0 = K(t, y)^T G(t, y).$$

This relation defines an invariant that the solution must satisfy.

Next, we extend the kernel matrix $K(t, y)$ by finding a complementary matrix $N(t, y)$ such that the combined matrix

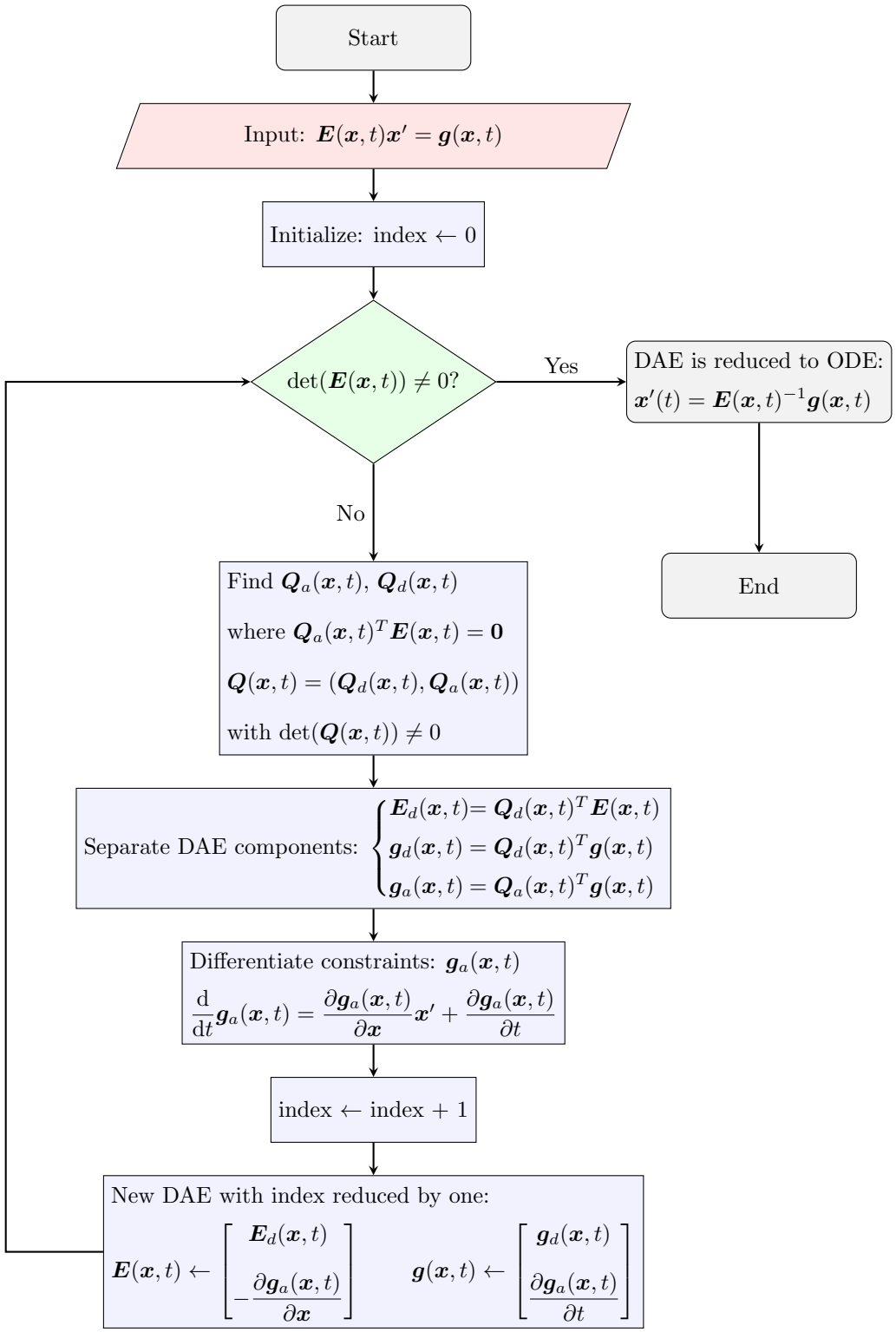


Figure 11.1: Index Reduction Algorithm for Nonlinear DAEs

$$M(t,y) = \begin{bmatrix} K(t,y) & N(t,y) \end{bmatrix}$$

is nonsingular (this process is often called the *completion of the kernel*).

Multiplying the original DAE by $M(t,y)^T$ separates the system into algebraic and differential parts:

$$M(t,y)^T E(t,y) y' = M(t,y)^T G(t,y),$$

which can be rewritten as

$$\begin{bmatrix} K(t,y)^T \\ N(t,y)^T \end{bmatrix} E(t,y) y' = \begin{bmatrix} K(t,y)^T \\ N(t,y)^T \end{bmatrix} G(t,y).$$

Manipulating the left side using $K^T E = 0$ gives

$$\begin{bmatrix} 0 \\ N(t,y)^T E(t,y) \end{bmatrix} y' = \begin{bmatrix} K(t,y)^T G(t,y) \\ N(t,y)^T G(t,y) \end{bmatrix}.$$

Define

$$\tilde{G}_1 = N(t,y)^T G(t,y), \quad \tilde{G}_2 = K(t,y)^T G(t,y).$$

Differentiating $\tilde{G}_2$ with respect to time,

$$\frac{d}{dt}\tilde{G}_2(t,y) = \frac{\partial \tilde{G}_2(t,y)}{\partial t} + \frac{\partial \tilde{G}_2(t,y)}{\partial y} y'.$$

The reduced DAE system becomes

$$\begin{bmatrix} M_1(t,y) \\ -\frac{\partial \tilde{G}_2(t,y)}{\partial y} \end{bmatrix} y' = \begin{bmatrix} \tilde{G}_1(t,y) \\ \frac{\partial \tilde{G}_2(t,y)}{\partial t} \end{bmatrix},$$

where $M_1(t,y) = N(t,y)^T E(t,y)$.

Thus, the DAE can be expressed as the system

$$\begin{cases} E_2(t,y) y' = G_2(t,y), \\ \tilde{G}_2(t,y) = 0, \end{cases}$$

with

$$E_2(t,y) = \begin{bmatrix} M_1(t,y) \\ -\frac{\partial \tilde{G}_2(t,y)}{\partial y} \end{bmatrix}, \quad G_2(t,y) = \begin{bmatrix} \tilde{G}_1(t,y) \\ \frac{\partial \tilde{G}_2(t,y)}{\partial t} \end{bmatrix}.$$

This reduction process decreases the index of the DAE by one.

In summary, the algorithm proceeds as follows:

- Compute the kernel of E^T.
- Complete the kernel to obtain a nonsingular matrix N.
- Separate the algebraic and differential parts.
- Differentiate the algebraic constraints.
- Repeat as needed to reduce the index.

11.4.1.1 Kernel Methods

The goal is to find a matrix $\mathbf{K}$ whose columns form a basis of Kernel($\mathbf{E}^T$), i.e., they satisfy

$$\mathbf{K}^T \mathbf{E} = \mathbf{0},$$

and such that the augmented matrix $(\mathbf{K}, \mathbf{N})$ is nonsingular (invertible), where $\mathbf{N}$ complements $\mathbf{K}$ to form a full-rank basis of the space.

Two classical methods to compute $\mathbf{K}$ are described below.

LU Decomposition

The LU decomposition factorizes the matrix $\mathbf{E}$ into a product of a lower triangular matrix $\mathbf{L}$ and an upper trapezoidal matrix $\mathbf{U}$, up to row and column permutations given by permutation matrices $\mathbf{P}$ and $\mathbf{Q}$:

$$\mathbf{PEQ} = \mathbf{LU}.$$

Here, $\mathbf{P}$ and $\mathbf{Q}$ are chosen to ensure numerical stability and reveal the rank structure of $\mathbf{E}$.

Rearranging, the original matrix $\mathbf{E}$ can be expressed as

$$\mathbf{E} = \mathbf{P}^T\mathbf{LUQ}^T.$$

Define

$$\mathbf{M} = \mathbf{UQ}^T,$$

so that

$$\mathbf{E} = \mathbf{P}^T\mathbf{LM}.$$

Because $\mathbf{U}$ is upper trapezoidal, $\mathbf{M}$ has a block structure of the form

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{M}_1$ is full rank.

Multiplying both sides by $\mathbf{L}^{-1}\mathbf{P}$, we obtain

$$\mathbf{L}^{-1}\mathbf{PE} = \mathbf{M} = \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{0} \end{bmatrix}.$$

Now, by pre-multiplying with the block row selectors $(\mathbf{I}, \mathbf{0})$ and $(\mathbf{0}, \mathbf{I})$, where $\mathbf{I}$ is the identity matrix and $\mathbf{0}$ the zero matrix, we isolate the nonzero and zero parts respectively:

$$(\mathbf{I}, \mathbf{0})\mathbf{L}^{-1}\mathbf{PE} = \mathbf{M}_1,$$

$$(\mathbf{0}, \mathbf{I})\mathbf{L}^{-1}\mathbf{PE} = \mathbf{0}.$$

From this, we define the matrices $\mathbf{K}$ and $\mathbf{N}$ as

$$(\mathbf{K}, \mathbf{N}) := \mathbf{P}^T\mathbf{L}^{-T} \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix},$$

so that

$$\mathbf{N}^T = (\mathbf{I}, \mathbf{0})\mathbf{L}^{-1}\mathbf{P}, \quad \text{and} \quad \mathbf{K}^T = (\mathbf{0}, \mathbf{I})\mathbf{L}^{-1}\mathbf{P}.$$

By construction, $\mathbf{K}^T\mathbf{E} = \mathbf{0}$, and $(\mathbf{K}, \mathbf{N})$ is nonsingular. Thus, the LU decomposition provides a practical way to compute the *kernel* of $\mathbf{E}$.

Singular Value Decomposition

Another standard approach is to use the Singular Value Decomposition (SVD), which factorizes $\mathbf{E}$ into orthogonal matrices $\mathbf{U}$, $\mathbf{V}$, and a diagonal matrix Σ:

$$\mathbf{E} = \mathbf{U}\Sigma\mathbf{V}^T,$$

where

$$\mathbf{U}^T\mathbf{U} = \mathbf{I}, \quad \mathbf{V}^T\mathbf{V} = \mathbf{I}, \quad \text{and} \quad \Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_r, 0, \ldots, 0).$$

Partitioning the matrices according to the rank r of $\mathbf{E}$, we write

$$\mathbf{E} = \left[\begin{array}{c|c} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \hline \mathbf{U}_{21} & \mathbf{U}_{22} \end{array}\right] \left[\begin{array}{c|c} \Sigma_r & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c|c} \mathbf{V}_{11}^T & \mathbf{V}_{12}^T \\ \hline \mathbf{V}_{21}^T & \mathbf{V}_{22}^T \end{array}\right].$$

Multiplying both sides by $\mathbf{U}^T$ and then selecting the lower block with $(\mathbf{0}, \mathbf{I})$, we get

$$(\mathbf{0}, \mathbf{I})\mathbf{U}^T\mathbf{E} = (\mathbf{0}, \mathbf{I}) \left[\begin{array}{c|c} \Sigma_r & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c|c} \mathbf{V}_{11}^T & \mathbf{V}_{12}^T \\ \hline \mathbf{V}_{21}^T & \mathbf{V}_{22}^T \end{array}\right] = \mathbf{0}.$$

This directly shows that the columns of $(\mathbf{0}, \mathbf{I})\mathbf{U}^T$ form a basis of the *kernel* of $\mathbf{E}$.
Defining

$$(\mathbf{K}, \mathbf{N}) = \mathbf{U} \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix},$$

we have

$$\mathbf{N}^T = (\mathbf{I}, \mathbf{0})\mathbf{U}^T, \quad \text{and} \quad \mathbf{K}^T = (\mathbf{0}, \mathbf{I})\mathbf{U}^T,$$

and thus $\mathbf{K}^T\mathbf{E} = \mathbf{0}$.

While the SVD method is robust and numerically stable, it is computationally more expensive than the LU approach and therefore less suited for large-scale or real-time applications. Also for symbolic computations, it is preferable to use LU decomposition.

11.4.2 A kernel-based separator

The factorization Equation 11.4 is not the only way to build the separator Equation 11.6. An equivalent construction starts from the left null space of $\boldsymbol{E}$,

$$\ker(\boldsymbol{E}^T) = \{\boldsymbol{v} \,:\, \boldsymbol{v}^T\boldsymbol{E} = \mathbf{0}\},$$

whose dimension is $n - \operatorname{rank}\boldsymbol{E}$. Let the columns of $\boldsymbol{K}$ be a basis of $\ker(\boldsymbol{E}^T)$, and let $\boldsymbol{L}$ complete that basis to $\mathbb{R}^n$, so that

$$\boldsymbol{S} = \begin{pmatrix} \boldsymbol{L}^T \\ \boldsymbol{K}^T \end{pmatrix} \quad \text{is nonsingular.}$$

Then

$$\boldsymbol{S}\boldsymbol{E} = \begin{pmatrix} \boldsymbol{L}^T\boldsymbol{E} \\ \boldsymbol{K}^T\boldsymbol{E} \end{pmatrix} = \begin{pmatrix} \boldsymbol{L}^T\boldsymbol{E} \\ \mathbf{0} \end{pmatrix},$$

which is again the separation of Equation 11.5, with the algebraic equations in the bottom block.

> ⚠ Correct, but not recommended
>
> The kernel construction is mathematically transparent and computationally awkward: it requires a basis of a null space and a completion of that basis to the whole space, both of which are more expensive and less stable than the elimination that produces Equation 11.4, and both of which must be recomputed at every reduction step and at every point $(\boldsymbol{x}, t)$. Use it to understand what the separator does, not to compute it.

11.4.3 Which construction to use

Four routes to the same separation, in decreasing order of practicality.

Construction	Cost	When it is the right choice
Elimination, $\boldsymbol{P}\boldsymbol{E} = \boldsymbol{L}\boldsymbol{M}$	$\mathcal{O}(n^3)$, cheapest	the default for a numerical implementation

Construction	Cost	When it is the right choice
QR with column pivoting	$\mathcal{O}(n^3)$, larger constant	when the rank decision is delicate and backward stability matters
SVD	$\mathcal{O}(n^3)$, largest constant	when the numerical rank is genuinely ambiguous and must be decided by singular values
Kernel basis $\boldsymbol{S} = (\boldsymbol{L}^T; \boldsymbol{K}^T)$	highest	exposition only

! For hand computation, do not use matrices at all

All of the above assume that the reduction is performed on the matrix $\boldsymbol{E}(\boldsymbol{x}, t)$. For a system small enough to be treated by hand — which includes every example in this book and most textbook models — that is the wrong approach. It is faster and far less error-prone to write the system in the semi-explicit form of Section 10.1, identify the algebraic equations by inspection, differentiate them, and substitute the differential equations into the result. The pendulum reduction of Section 11.5 is carried out that way in four lines, whereas the matrix route requires assembling and factorizing a 5×5 state-dependent matrix at each of three steps.

The matrix formulation earns its keep when the system is large, when the reduction must be automated, or when the rank of $\boldsymbol{E}$ is not obvious — in other words, exactly when a human cannot do it by inspection.

11.5 Examples

A high-index chain

Consider the following DAE system:

$$\begin{cases} x_6'(t) + x_5(t) = \sin t \\ x_5'(t) + x_4(t) = \sin t \\ x_4'(t) + x_3(t) = \sin t \\ x_3'(t) + x_2(t) = \sin t \\ x_2'(t) + x_1(t) = \sin t \\ \qquad\quad x_6(t) = \cos t \end{cases}$$

We define the variable vector and matrices:

$$Z = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix}, \quad \boldsymbol{E}(Z, t) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad G(Z, t) = \begin{pmatrix} \sin t - x_5(t) \\ \sin t - x_4(t) \\ \sin t - x_3(t) \\ \sin t - x_2(t) \\ \sin t - x_1(t) \\ \cos t - x_6(t) \end{pmatrix}$$

So that the DAE reads:

$$E(Z,t)\,Z'(t) = G(Z,t)$$

11.5.0.1 Index Reduction

The table

$$T = \left[\begin{array}{c|c} E(Z,t) & G(Z,t) \end{array}\right] = \left[\begin{array}{cccccc|c} 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos t - x_6(t) \end{array}\right]$$

is used to memorize the operation done on the DAE.

First Reduction

Reduce index by one

1. Let

$$[\text{alg1}] = \cos t - x_6$$

2. Derive the constraint [alg1]

$$\frac{\mathrm{d}[\text{alg1}]}{\mathrm{d}t} = -\sin t - x_6'$$

3. rewrite the DAE

$$T_1 = \left[\begin{array}{cccccc|c} 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & -\sin t \end{array}\right]$$

Second Reduction

1. Reorder the equation

$$T_2 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & -\sin t \end{array}\right]$$

2. Subtract row 5 to row 6

$$T_3 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 0 & 0 & x_5(t) - 2\sin t \end{array}\right]$$

3. found algebraic constraint

$$[\text{alg2}] = x_5 - 2\sin t$$

Derive constraint [alg2]

$$\frac{\mathrm{d}[\text{alg2}]}{\mathrm{d}t} = x_5' - 2\cos t$$

so that

$$T_4 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & 2\cos t \end{array}\right]$$

Third Reduction

1. Subtract row 4 to row 6

$$T_5 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 0 & 0 & x_4(t) + 2\cos t - \sin t \end{array}\right]$$

2. found algebraic constraint

$$[\text{alg3}] = x_4 + 2\cos t - \sin t$$

3. Reduce index by one by deriving the constraint [alg3]

$$\frac{\mathrm{d}[\text{alg3}]}{\mathrm{d}t} = x_4' - \cos t - 2\sin t$$

so that

$$T_6 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \cos t + 2\sin t \end{array}\right]$$

Fourth Reduction

1. Subtract row 3 to row 6

$$T_7 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos t + \sin t + x_3(t) \end{array}\right]$$

   found algebraic constraint

$$[\text{alg4}] = x_3 + \sin t + \cos t$$

2. Reduce index by one by deriving the constraint [alg4]

$$\frac{\mathrm{d}[\text{alg4}]}{\mathrm{d}t} = x_3' + \cos t - \sin t$$

   so that

$$T_8 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - \cos t \end{array}\right]$$

Fifth Reduction

1. Subtract row 2 to row 6

$$T_9 = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 0 & 0 & x_2(t) - \cos t \end{array}\right]$$

2. found algebraic constraint

$$[\text{alg5}] = x_2 - \cos t$$

3. Reduce index by one by deriving the constraint [alg5]

$$\frac{\mathrm{d}[\text{alg5}]}{\mathrm{d}t} = x_2' + \sin t$$

so that

$$T_{10} = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 1 & 0 & 0 & 0 & 0 & -\sin t \end{array}\right]$$

Sixth Reduction

1. Subtract row 1 to row 6

$$T_{11} = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 0 & 0 & 0 & 0 & 0 & 0 & x_1(t) - 2\sin t \end{array}\right]$$

2. found algebraic constraint

$$[\text{alg6}] = x_1 - 2\sin t$$

3. Reduce index y one by deriving the constraint [alg6]

$$\frac{\mathrm{d}[\text{alg6}]}{\mathrm{d}t} = x_1' - 2\cos t$$

so that

$$T_{12} = \left[\begin{array}{cccccc|c} 0 & 1 & 0 & 0 & 0 & 0 & \sin t - x_1(t) \\ 0 & 0 & 1 & 0 & 0 & 0 & \sin t - x_2(t) \\ 0 & 0 & 0 & 1 & 0 & 0 & \sin t - x_3(t) \\ 0 & 0 & 0 & 0 & 1 & 0 & \sin t - x_4(t) \\ 0 & 0 & 0 & 0 & 0 & 1 & \sin t - x_5(t) \\ 1 & 0 & 0 & 0 & 0 & 0 & 2\cos t \end{array}\right]$$

ODE Form

Now matrix is non-singular ans the resulting ODE is

$$\begin{cases} x_1'(t) = 2\cos t \\ x_2'(t) = \sin t - x_1(t) \\ x_3'(t) = \sin t - x_2(t) \\ x_4'(t) = \sin t - x_3(t) \\ x_5'(t) = \sin t - x_4(t) \\ x_6'(t) = \sin t - x_5(t) \end{cases}$$

with invariants (or hidden constraints)

$$\begin{aligned} [\text{alg1}] &= \cos t - x_6 \\ [\text{alg2}] &= x_5 - 2\sin t \\ [\text{alg3}] &= x_4 + 2\cos t - \sin t \\ [\text{alg4}] &= x_3 + \sin t + \cos t \\ [\text{alg5}] &= x_2 - \cos t \\ [\text{alg6}] &= x_1 - 2\sin t \end{aligned}$$

Notice that from the invariants we can deduce

$$\begin{aligned} x_1(t) &= 2\sin t, \\ x_2(t) &= \cos t, \\ x_3(t) &= -\sin t - \cos t \\ x_4(t) &= \sin t - 2\cos t, \\ x_5(t) &= 2\sin t, \\ x_6(t) &= \cos t \end{aligned}$$

and the general solution in determined without fixing any initial value.

Numerical solution witb $\Delta t = 0.2$. Notice the drift for components x_6 and x_5.

11.5.0.2 Dummy-Variable Reduction

Variable Classification

- **Differential variables**: x_1, x_2, x_3, x_4. x_5
- **Algebraic variables**: x_6

We attempt to solve for u_6 and v_{dot}, but due to the presence of λ, the system is not yet solvable explicitly. We proceed by **differentiating**.

Step 1: Differentiate Constraints

Differentiate (Constraint 1) $= \cos t - x_6$:

$$\frac{\mathrm{d}(\text{Constraint 1})}{\mathrm{d}t} = -\sin t - x_6'$$

Substitute known differential variables:

$$(\text{Constraint 2}) = x_5 - 2\sin t$$

x_1' not found, continue index reduction.

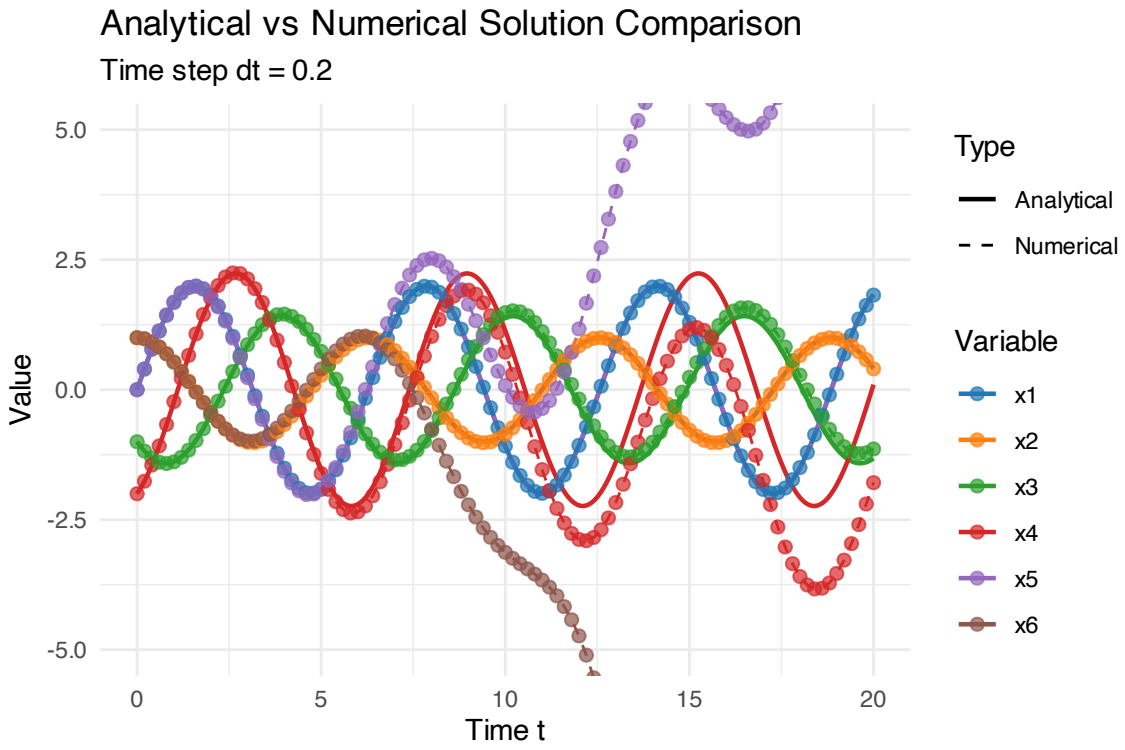


Figure 11.2: Constraint drift in the direct numerical solution of the index-6 DAE.

Step 2: Differentiate Constraints

Differentiate (Constraint 2)

$$\frac{\mathrm{d}(\text{Constraint 2})}{\mathrm{d}t} = x_5' - 2\cos t \quad \Longrightarrow \quad (\text{Constraint 3}) = \sin t - 2\cos t - x_4$$

x_1' not found, continue index reduction.

Step 3: Differentiate Constraints

Differentiate (Constraint 3):

$$\frac{\mathrm{d}(\text{Constraint 3})}{\mathrm{d}t} = \cos t + 2\sin t - x_4' \quad \Longrightarrow \quad (\text{Constraint 4}) = \cos t + \sin t + x_3$$

x_1' not found, continue index reduction.

Step 4: Differentiate Constraints

Differentiate (Constraint 4):

$$\frac{\mathrm{d}(\text{Constraint 4})}{\mathrm{d}t} = -\sin t + \cos t + x_3' \quad \Longrightarrow \quad (\text{Constraint 5}) = \cos t - x_2$$

x_1' not found, continue index reduction.

Step 5: Differentiate Constraints

Differentiate (Constraint 5):

$$\frac{\mathrm{d}(\text{Constraint 5})}{\mathrm{d}t} = -\sin t - x_2' \quad \Longrightarrow \quad (\text{Constraint 6}) = -2\sin t + x_1$$

x_1' not found, continue index reduction.

Step 6: Differentiate Constraints

Differentiate (Constraint 6):

$$\frac{\mathrm{d}(\text{Constraint 6})}{\mathrm{d}t} = -2\cos t - 2\sin t + x_1'$$

x_1' found, index reduction complete

$$\begin{cases} x_6' = \sin t - x_5 \\ x_5' = \sin t - x_4 \\ x_4' = \sin t - x_3 \\ x_3' = \sin t - x_2 \\ x_2' = \sin t - x_1 \\ x_1' = 2\sin t \end{cases} \quad \begin{cases} (\text{Constraint 1}) = \cos t - x_6 = 0 \\ (\text{Constraint 2}) = x_5 - 2\sin t = 0 \\ (\text{Constraint 3}) = \sin t - 2\cos t - x_4 = 0 \\ (\text{Constraint 4}) = \sin t + \cos t + x_3 = 0 \\ (\text{Constraint 5}) = \cos t - x_2 = 0 \\ (\text{Constraint 6}) = x_1 - 2\sin t = 0 \end{cases}$$

Applied 6 time differentiation, the **index is 6**.

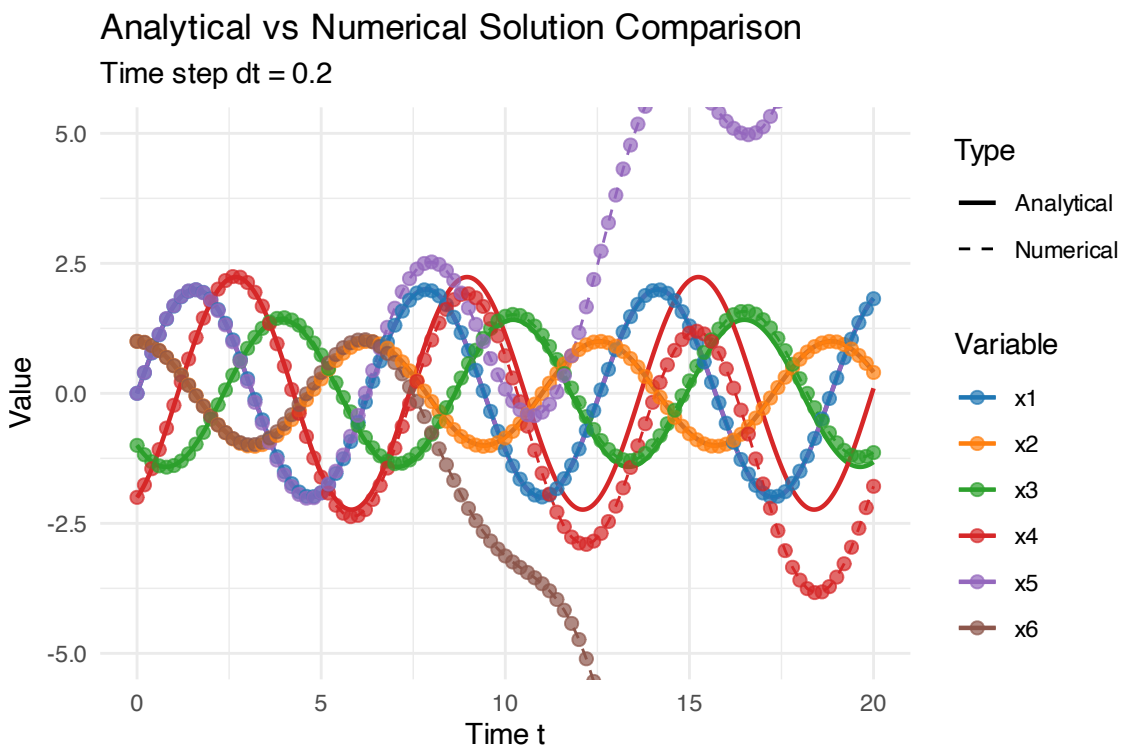


Figure 11.3: Constraint drift after differentiating the hidden constraints.

Numerical solution witb $\Delta t = 0.2$. Notice the drift for components x_6 and x_5.

12. Dummy Variables

Repeated index reduction has a cost that the algorithms of Chapter 11 do not hide: every differentiation enlarges the system and the expressions grow, so that a high-index model can become unmanageable well before it becomes an ODE.

The dummy-derivative method of [17] reduces that cost. Instead of differentiating constraints and accumulating equations, it introduces new algebraic unknowns — *dummy variables* — that stand for the derivatives already computed, so that the differentiated constraints can replace the original ones rather than being added to them. The size of the system stays bounded, and the invariants are retained explicitly.

12.1 Dummy-Variable Reduction

12.1.1 Overview

The **dummy variable method** provides a systematic approach for:

- Computing the differential index of nonlinear DAEs
- Transforming high-index problems into solvable forms
- Handling algebraic constraints through variable substitution

12.1.2 Requirements

Initial Form

Start with a general implicit DAE:

$$F(\boldsymbol{x}, \boldsymbol{x}', t) = \mathbf{0}, \quad \boldsymbol{x} \in \mathbb{R}^n$$

Required Semi-Explicit Form

Restructure to semi-explicit form (see Section 10.1):

$$\begin{cases} \boldsymbol{x}_d' = \boldsymbol{f}(\boldsymbol{x}_d, \boldsymbol{x}_a, t) \\ \mathbf{0} = \boldsymbol{g}(\boldsymbol{x}_d, \boldsymbol{x}_a, t) \end{cases} \tag{12.1}$$

Variables:

- $\boldsymbol{x}_d \in \mathbb{R}^{n_d}$: Differential variables
- $\boldsymbol{x}_a \in \mathbb{R}^{n_a}$: Algebraic variables ($n_d + n_a = n$)

> **Regularity Condition**
>
> The algebraic constraints must be **linear** in $\boldsymbol{x}_a$:
>
> $$\boldsymbol{f}(\boldsymbol{x}_d, \boldsymbol{x}_a, t) \equiv \boldsymbol{f}_0(\boldsymbol{x}_d, t) + \boldsymbol{A}(\boldsymbol{x}_d, t)\boldsymbol{x}_a$$
> $$\boldsymbol{g}(\boldsymbol{x}_d, \boldsymbol{x}_a, t) \equiv \boldsymbol{g}_0(\boldsymbol{x}_d, t) + \boldsymbol{B}(\boldsymbol{x}_d, t)\boldsymbol{x}_a$$

Variable Classification

Partition algebraic variables $\boldsymbol{x}_a$ of Equation 12.1

Type	Notation	Description
Linear	$\boldsymbol{x}_l$	Appears linearly/absent in constraints
Nonlinear	$\boldsymbol{x}_n$	Appears nonlinearly in constraints

12.1.3 Preprocessing

For $\boldsymbol{x}_n \neq \emptyset$:

1. Introduce dummy derivatives:
$$\boldsymbol{x}_n' = \boldsymbol{v}_n$$
2. Reclassify variables:
   - **New differential**: $(\boldsymbol{x}_d, \boldsymbol{x}_n)$
   - **New algebraic**: $(\boldsymbol{x}_l, \boldsymbol{v}_n)$

Now all the algebraic variables appears linearly. In fact consider the DAE before the preprocessing

$$\begin{cases} \boldsymbol{x}_d' = \boldsymbol{f}_0(\boldsymbol{x}_d, \boldsymbol{x}_n, t) + \boldsymbol{A}(\boldsymbol{x}_d, \boldsymbol{x}_n, t)\boldsymbol{x}_l \\ \boldsymbol{0} = \boldsymbol{g}_0(\boldsymbol{x}_d, \boldsymbol{x}_n, t) + \boldsymbol{B}(\boldsymbol{x}_d, \boldsymbol{x}_n, t)\boldsymbol{x}_l \end{cases}$$

introducing the dummy variables $\boldsymbol{x}_n' = \boldsymbol{v}_n$ the DAE become

$$\begin{cases} \begin{pmatrix} \boldsymbol{x}_d' \\ \boldsymbol{x}_n' \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_0(\boldsymbol{x}_d, \boldsymbol{x}_n, t) \\ \boldsymbol{0} \end{pmatrix} + \begin{pmatrix} \boldsymbol{A}(\boldsymbol{x}_d, \boldsymbol{x}_n, t) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_l \\ \boldsymbol{v}_n \end{pmatrix} \\ \boldsymbol{0} = \boldsymbol{g}_0(\boldsymbol{x}_d, \boldsymbol{x}_n, t) + \boldsymbol{B}(\boldsymbol{x}_d, \boldsymbol{x}_n, t)\boldsymbol{x}_l \end{cases}$$

and defining

- **New differential**: $\widetilde{\boldsymbol{x}}_d = (\boldsymbol{x}_d, \boldsymbol{x}_n)$
- **New algebraic**: $\widetilde{\boldsymbol{x}}_a = (\boldsymbol{x}_l, \boldsymbol{v}_n)$
- $\widetilde{\boldsymbol{f}}_0(\widetilde{\boldsymbol{x}}_d, t) = \begin{pmatrix} \boldsymbol{f}_0(\boldsymbol{x}_d, \boldsymbol{x}_n, t) \\ \boldsymbol{0} \end{pmatrix}$
- $\widetilde{\boldsymbol{A}}(\widetilde{\boldsymbol{x}}_d, t) = \begin{pmatrix} \boldsymbol{A}(\boldsymbol{x}_d, \boldsymbol{x}_n, t) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{pmatrix}$
- $\widetilde{\boldsymbol{g}}_0(\widetilde{\boldsymbol{x}}_d, t) = \boldsymbol{g}_0(\boldsymbol{x}_d, \boldsymbol{x}_n, t)$
- $\widetilde{\boldsymbol{B}}(\widetilde{\boldsymbol{x}}_d, t) = \begin{pmatrix} \boldsymbol{B}(\boldsymbol{x}_d, \boldsymbol{x}_n, t) & \boldsymbol{0} \end{pmatrix}$

the DAE become

$$\begin{cases} \widetilde{\boldsymbol{x}}_d' = \widetilde{\boldsymbol{f}}_0(\widetilde{\boldsymbol{x}}_d, t) + \widetilde{\boldsymbol{A}}(\widetilde{\boldsymbol{x}}_d, t)\widetilde{\boldsymbol{x}}_a \\ \boldsymbol{0} = \widetilde{\boldsymbol{g}}(\widetilde{\boldsymbol{x}}_d, t) + \widetilde{\boldsymbol{B}}(\widetilde{\boldsymbol{x}}_d, t)\widetilde{\boldsymbol{x}}_a \end{cases}$$

▪ **Example 12.1** Consider the following DAE system:

$$\begin{aligned} x' &= u \\ y' &= v \\ 2u' + v' + 2tx\lambda &= t \\ u' + 2v' - \lambda &= 1 - t \\ tx^2 - y^2 &= 1 \end{aligned}$$

Here:

- $\{x, y, u, v\}$ are **differential variables**
- λ is an **algebraic variable** that appears only **linearly**
- $\boldsymbol{x}_n = \emptyset$, $\boldsymbol{x}_l = \{\lambda\}$

This system satisfies the requirements of the dummy variable method.

▪

After preprocessing we can assume that DAE can be written as

$$\begin{cases} \boldsymbol{x}_d' = \boldsymbol{f}(\boldsymbol{x}_d, t) + \boldsymbol{A}(\boldsymbol{x}_d, t)\boldsymbol{x}_a \\ \boldsymbol{0} = \boldsymbol{g}(\boldsymbol{x}_d, t) + \boldsymbol{B}(\boldsymbol{x}_d, t)\boldsymbol{x}_a \end{cases}$$

where we have separation of differential and algebraic variables and algebraic variables appears linearly.

12.1.4 Reduction Algorithm

The index reduction algorithm now work as for the index reduction for nonlinear DAE (see Section 11.3.4) but working with the matrix $\boldsymbol{B}(\boldsymbol{x}_d, t)$.

Step 1: Check Singularity

The constraints are written as:

$$\boldsymbol{B}(\boldsymbol{x}_d, t)\boldsymbol{x}_a = -\boldsymbol{g}(\boldsymbol{x}_d, t),$$

Compute the determinant:

- If $\det(\boldsymbol{B}(\boldsymbol{x}_d, t)) \neq 0$, then $\boldsymbol{B}(\boldsymbol{x}_d, t)$ is invertible
  - the DAE is of **index 1**: $\boldsymbol{x}_a = -\boldsymbol{B}(\boldsymbol{x}_d, t)^{-1}\boldsymbol{g}(\boldsymbol{x}_d, t)$
  - **Terminate the procedure.**
- If $\det(\boldsymbol{B}(\boldsymbol{x}_d, t)) = 0$
  - $\boldsymbol{B}(\boldsymbol{x}_d, t)$ is singular
  - the DAE is of index > 1.
  - **Proceed to the next step.**

Step 2: Decompose the System

Determine the **rank** of $\boldsymbol{B}(\boldsymbol{x}_d, t)$, denoted as $r = \operatorname{rank}(\boldsymbol{B})$ and define:

- A **basis** $\boldsymbol{Q}_a(\boldsymbol{x}, t) \in \mathbb{R}^{n_a \times (n_a - k)}$ for the **null space** $\mathcal{N}(\boldsymbol{B}(\boldsymbol{x}, t)^T)$, which identifies the algebraic constraints that can be differentiated. This basis satisfies:
  - $\boldsymbol{B}(\boldsymbol{x}_d, t)^T \boldsymbol{Q}_a(\boldsymbol{x}_d, t) = \boldsymbol{0}$ or equivalently

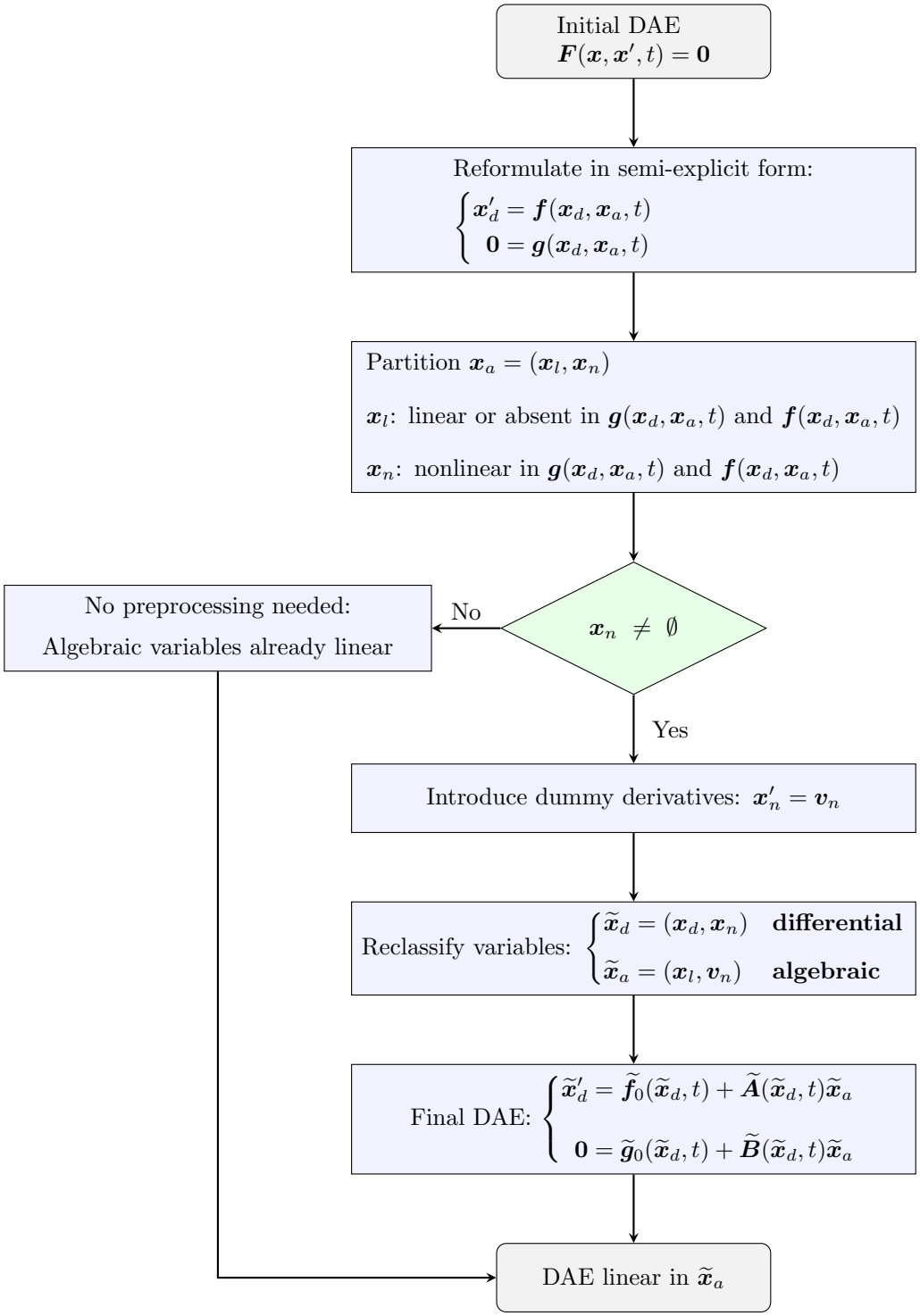


Figure 12.1: Preprocessing of the DAEs

– $\boldsymbol{Q}_a(\boldsymbol{x}_d, t)^T \boldsymbol{B}(\boldsymbol{x}_d, t) = \boldsymbol{0}$.

- A matrix $\boldsymbol{Q}_d(\boldsymbol{x}_d, t) \in \mathbb{R}^{n_a \times k}$ that completes $\boldsymbol{Q}_a(\boldsymbol{x}_d, t)$ to a full basis, ensuring that:

$$\boldsymbol{Q}(\boldsymbol{x}_d, t) = \big(\boldsymbol{Q}_d(\boldsymbol{x}_d, t), \boldsymbol{Q}_a(\boldsymbol{x}_d, t)\big)$$

is non-singular.

By multiplying the original constraints by $\boldsymbol{Q}(\boldsymbol{x}_d, t)^T$, we obtain:

- **Part that contains $\boldsymbol{x}_a$:**

$$\boldsymbol{B}_d(\boldsymbol{x}_d, t)\boldsymbol{x}_a = -\boldsymbol{g}_d(\boldsymbol{x}_d, t),$$

where:

$$\boldsymbol{B}_d(\boldsymbol{x}_d, t) = \boldsymbol{Q}_d(\boldsymbol{x}_d, t)^T \boldsymbol{B}(\boldsymbol{x}_d, t),$$
$$\boldsymbol{g}_d(\boldsymbol{x}_d, t) = \boldsymbol{Q}_d(\boldsymbol{x}_d, t)^T \boldsymbol{g}(\boldsymbol{x}_d, t).$$

- **Part free of $\boldsymbol{x}_a$:**

$$\boldsymbol{g}_a(\boldsymbol{x}, t) = \boldsymbol{0},$$

where:

$$\boldsymbol{g}_a(\boldsymbol{x}_d, t) = \boldsymbol{Q}_a(\boldsymbol{x}_d, t)^T \boldsymbol{g}(\boldsymbol{x}_d, t).$$

Since $\boldsymbol{Q}_a(\boldsymbol{x}_d, t)^T \boldsymbol{B}(\boldsymbol{x}_d, t) = \boldsymbol{0}$, no algebraic variables appear in this equation.

Step 3: Differentiate Constraints

The algebraic constraints must be differentiated to reveal hidden differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{g}_a(\boldsymbol{x}_d, t) = \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial t} + \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d}\boldsymbol{x}_d'(t) = \boldsymbol{0}.$$

Rewriting, the constraints becomes:

$$\begin{pmatrix} \boldsymbol{B}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d} \end{pmatrix} \boldsymbol{x}_d' = -\begin{pmatrix} \boldsymbol{g}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial t} \end{pmatrix}.$$

and using $\boldsymbol{x}_d' = \boldsymbol{f}(\boldsymbol{x}_d, t) + \boldsymbol{A}(\boldsymbol{x}_d, t)\boldsymbol{x}_a$ can be rewritten as

$$\begin{pmatrix} \boldsymbol{B}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d} \end{pmatrix} \boldsymbol{A}(\boldsymbol{x}_d, t)\boldsymbol{x}_a = -\begin{pmatrix} \boldsymbol{g}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial t} \end{pmatrix} - \begin{pmatrix} \boldsymbol{B}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d} \end{pmatrix} \boldsymbol{f}(\boldsymbol{x}_d, t)$$

or better

$$\begin{pmatrix} \boldsymbol{B}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d} \end{pmatrix} \boldsymbol{A}(\boldsymbol{x}_d, t)\boldsymbol{x}_a = -\begin{pmatrix} \boldsymbol{g}_d(\boldsymbol{x}_d, t) + \boldsymbol{B}_d(\boldsymbol{x}_d, t)\boldsymbol{f}(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial t} + \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d}\boldsymbol{f}(\boldsymbol{x}_d, t) \end{pmatrix}$$

Defining:

$$\widetilde{\boldsymbol{B}}(\boldsymbol{x}_d, t) = \begin{pmatrix} \boldsymbol{B}_d(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d} \end{pmatrix} \boldsymbol{A}(\boldsymbol{x}_d, t),$$

$$\widetilde{\boldsymbol{g}}(\boldsymbol{x}_d, t) = \begin{pmatrix} \boldsymbol{g}_d(\boldsymbol{x}_d, t) + \boldsymbol{B}_d(\boldsymbol{x}_d, t)\boldsymbol{f}(\boldsymbol{x}_d, t) \\ \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial t} + \frac{\partial \boldsymbol{g}_a(\boldsymbol{x}_d, t)}{\partial \boldsymbol{x}_d}\boldsymbol{f}(\boldsymbol{x}_d, t) \end{pmatrix}$$

we obtain a new constraint part of the DAE:

$$\widetilde{\boldsymbol{B}}(\boldsymbol{x}_d, t)\boldsymbol{x}_a = -\widetilde{\boldsymbol{g}}(\boldsymbol{x}_d, t).$$

notice that the DAE is still in the form with linear algebraic constraints.

Go to step 1.

The **differentiation index** is the number of times Step 4 is performed plus one before reaching an index 1 DAE.

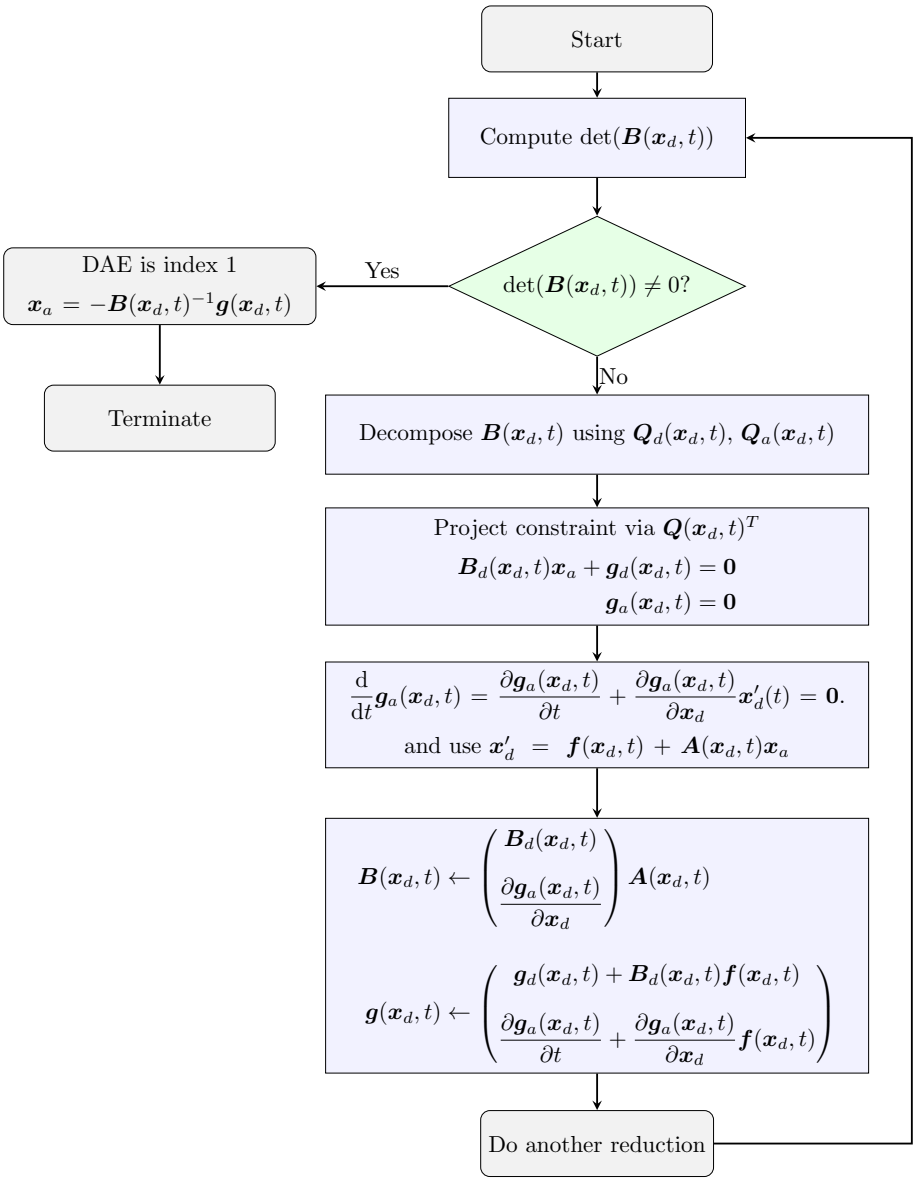


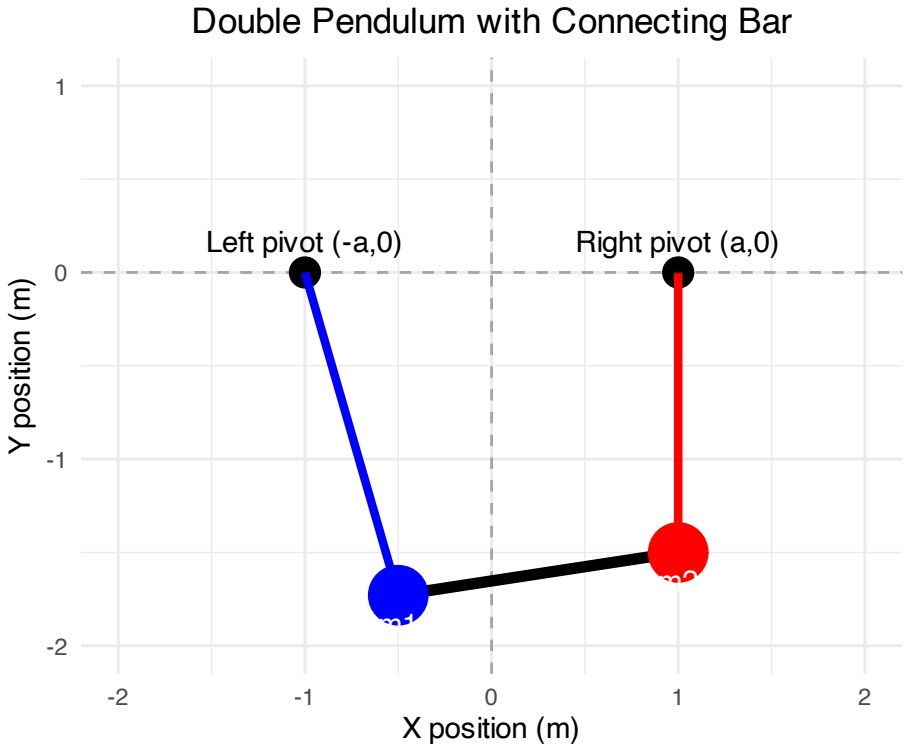
Figure 12.2: Index reduction by dummy variables

12.2 Examples

Coupled pendulums

A constrained mechanical system consisting of two rigid pendulums connected by a fixed-length bar. In details

1. Left Pendulum Assembly:
   - Pivot: Fixed at point $(-a, 0)$
   - Mass m_1 located at (x_1, y_1)
   - Constraint: Rigid rod maintains length $\ell_1^2 = (x_1 + a)^2 + y_1^2$
2. Right Pendulum Assembly:
   - Pivot: Fixed at point $(a, 0)$
   - Mass m_2 located at (x_2, y_2)
   - Length $\ell_2^2 = (x_2 - a)^2 + y_2^2$ fixed
3. Coupling Mechanism:
   - Connecting bar: Rigid link between m_1 and m_2
   - Constraint: Rigid rod maintains length $\ell_3^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$



DAE Derivation

Now the Differential-Algebraic Equations (DAE) for the double pendulum system with a connecting bar is derived.

We employ Cartesian coordinates with the following state variables:

- Position states:
  - (x_1, y_1): Coordinates of left mass m_1
  - (x_2, y_2): Coordinates of right mass m_2
- Velocity states:
  - (u_1, v_1): Velocity components of m_1
  - (u_2, v_2): Velocity components of m_2

- Constraint forces:
  - λ_1: Multiplier for left pendulum constraint
  - λ_2: Multiplier for right pendulum constraint
  - λ_3: Multiplier for connecting bar constraint

Applying Lagrange multiplier method with constraints yields:

Equations of Motion

Mass m_1

$$\begin{cases} m_1 u_1' = -\lambda_1(x_1 + a) - \lambda_3(x_1 - x_2) \\ m_1 v_1' = -\lambda_1 y_1 - \lambda_3(y_1 - y_2) - m_1 g \end{cases}$$

Mass m_2

$$\begin{cases} m_2 u_2' = -\lambda_2(x_2 - a) + \lambda_3(x_1 - x_2) \\ m_2 v_2' = -\lambda_2 y_2 + \lambda_3(y_1 - y_2) - m_2 g \end{cases}$$

Algebraic Constraints

$$\begin{cases} (x_1 + a)^2 + y_1^2 = \ell_1^2 \\ (x_2 - a)^2 + y_2^2 = \ell_2^2 \\ (x_1 - x_2)^2 + (y_1 - y_2)^2 = \ell_3^2 \end{cases}$$

12.2.0.1 Complete DAE

Introducing

$$\delta x = x_1 - x_2, \quad \delta y = y_1 - y_2, \quad \delta u = u_1 - u_2, \quad \delta v = v_1 - v_2,$$

the complete DAE system is:

Differential Part

$$\begin{cases} x_1' = u_1 \\ y_1' = v_1 \\ x_2' = u_2 \\ y_2' = v_2 \\ m_1 u_1' = -(x_1 + a)\lambda_1 - \delta x \lambda_3 \\ m_1 v_1' = -y_1 \lambda_1 - \delta y \lambda_3 - m_1 g \\ m_2 u_2' = -(x_2 - a)\lambda_2 + \delta x \lambda_3 \\ m_2 v_2' = -y_2 \lambda_2 + \delta y \lambda_3 - m_2 g \end{cases}$$

Algebraic Part

$$\begin{cases} (x_1 + a)^2 + y_1^2 - \ell_1^2 = 0, & \text{(Constraint 1)} \\ (x_2 - a)^2 + y_2^2 - \ell_2^2 = 0, & \text{(Constraint 2)} \\ \delta x^2 + \delta y^2 - \ell_3^2 = 0, & \text{(Constraint 3)} \end{cases}$$

12.2.0.2 Semi-Explicit Form

Decomposes previous DAE into differential (blue) and algebraic (red) components:

$$
\begin{cases}
x_1' = u_1 \\
y_1' = v_1 \\
x_2' = u_2 \\
y_2' = v_2 \\
u_1' = \dfrac{-(x_1 + a)\lambda_1 - \delta x \lambda_3}{m_1} \\
v_1' = \dfrac{-y_1\lambda_1 - \delta y \lambda_3}{m_1} - g \\
u_2' = \dfrac{-(x_2 - a)\lambda_2 + \delta x \lambda_3}{m_2} \\
v_2' = \dfrac{-y_2\lambda_2 + \delta y \lambda_3}{m_2} - g
\end{cases}
\bigoplus
\begin{cases}
(x_1 + a)^2 + y_1^2 - \ell_1^2 = 0, & \text{(Constraint 1)} \\
(x_2 - a)^2 + y_2^2 - \ell_2^2 = 0, & \text{(Constraint 2)} \\
\delta x^2 + \delta y^2 - \ell_3^2 = 0, & \text{(Constraint 3)}
\end{cases}
$$

For this DAE we have:

- **Differential variables**: x_1, y_1, x_2, y_2, u_1, v_1, u_2, v_2.
- **Algebraic variables**: λ_1, λ_2, λ_3.

no dummy variable are required.

12.2.0.3 Reduction Procedure

We attempt to solve for λ_1, λ_2 and λ_3 but the system is not yet solvable explicitly. We proceed by **differentiating**.

Step 1: Differentiate Algebraic Constraints

We now differentiate the constraints to reduce the index. Here we want to differentiate **only** the algebraic equation that can reduce the index and do not introduce new differential equation. Here we want to avoid to introduce derivatove of the algebraic variable unless is strictly necessary.

We construct a dependency table to identify which constraints to differentiate:

	λ_1	λ_2	λ_3
(Constraint 1)	0	0	0
(Constraint 2)	0	0	0
(Constraint 3)	0	0	0

a 0 means the the function do not depend on the variable on the column. A 1 means dependence. All constraint are candidate for differentiation as it doesn't introduce new algebraic variables. Differentiate thus the constraints and substitute known variable in differential form using the ODE part

$$
\begin{aligned}
\text{(Constraint 1)}' &= 2(x_1 + a)x_1' + 2y_1y_1' = 2(x_1 + a)u_1 + 2y_1v_1 \\
\text{(Constraint 2)}' &= 2(x_2 - a)x_2' + 2y_2y_2' = 2(x_2 - a)u_2 + 2y_2v_2 \\
\text{(Constraint 3)}' &= 2\delta x \delta x' + 2\delta y \delta y' = 2\delta x \delta u + 2\delta y \delta v
\end{aligned}
$$

so that the DAE become:

$$
\begin{cases}
x_1' = u_1 \\
y_1' = v_1 \\
x_2' = u_2 \\
y_2' = v_2 \\
u_1' = \dfrac{-(x_1 + a)\lambda_1 - \delta x \lambda_3}{m_1} \\
v_1' = \dfrac{-y_1\lambda_1 - \delta y \lambda_3}{m_1} - g \\
u_2' = \dfrac{-(x_2 - a)\lambda_2 + \delta x \lambda_3}{m_2} \\
v_2' = \dfrac{-y_2\lambda_2 + \delta y \lambda_3}{m_2} - g
\end{cases}
\oplus
\begin{cases}
2(x_1 + a)u_1 + 2y_1v_1 = 0, & \text{(Constraint 1)}' \\
2(x_2 - a)u_2 + 2y_2v_2 = 0, & \text{(Constraint 2)}' \\
2\delta x \delta u + 2\delta y \delta v = 0, & \text{(Constraint 3)}' \\
\\
(x_1 + a)^2 + y_1^2 - \ell_1^2 = 0, & \text{(Constraint 1)} \\
(x_2 - a)^2 + y_2^2 - \ell_2^2 = 0, & \text{(Constraint 2)} \\
\delta x^2 + \delta y^2 - \ell_3^2 = 0, & \text{(Constraint 3)}
\end{cases}
$$

The red algebraic equations are the new algebraic equation after the derivation. The green algebraic equation is the hidden constraints that is removed from the DAE system but that is satisfied by choosing the right initial conditions.

The red part cannot determine the algebraic variable and must be differentiated again.

Step 2: Differentiate Again

To find the right candidate(s) here we write a table of the dependencies

	λ_1	λ_2	λ_3
(Constraint 1)$'$	0	0	0
(Constraint 2)$'$	0	0	0
(Constraint 3)$'$	0	0	0

Again all the constraints can be differentiated

$$
\begin{aligned}
\text{(Constraint 1)}'' &= 2x_1'u_1 + 2y_1'v_1 + 2(x_1 + a)u_1' + 2y_1v_1' \\
&= 2(u_1^2 + v_1^2) + 2(x_1 + a)u_1' + 2y_1v_1' \\
\text{(Constraint 2)}'' &= 2x_2'u_2 + 2y_2'v_2 + 2(x_2 - a)u_2' + 2y_2v_2' \\
&= 2(u_2^2 + v_2^2) + 2(x_2 - a)u_2' + 2y_2v_2' \\
\text{(Constraint 3)}'' &= 2\delta x' \delta u + 2\delta x \delta u' + 2\delta y' \delta v + 2\delta y \delta v' \\
&= 2(\delta u^2 + \delta v^2) + 2\delta x \delta u' + 2\delta y \delta v'
\end{aligned}
$$

the substitution of the derivative make the constraint complex. Thus, dummy variable are introduced to reduce complexity:

$$
\begin{aligned}
\text{(Constraint 1)}'' &= 2(u_1^2 + v_1^2) + 2(x_1 + a)u_{1,\text{dot}} + 2y_1v_{1,\text{dot}} \\
\text{(Constraint 2)}'' &= 2(u_2^2 + v_2^2) + 2(x_2 - a)u_{2,\text{dot}} + 2y_2v_{2,\text{dot}} \\
\text{(Constraint 3)}'' &= 2(\delta u^2 + \delta v^2) + 2\delta x(u_{1,\text{dot}} - u_{2,\text{dot}}) + 2\delta y(v_{1,\text{dot}} - v_{2,\text{dot}})
\end{aligned}
$$

and the DAE, introducing

$$\delta u_{\text{dot}} = u_{1,\text{dot}} - u_{2,\text{dot}}, \qquad \delta v_{\text{dot}} = v_{1,\text{dot}} - v_{2,\text{dot}}$$

become:

$$\begin{cases} x_1' = u_1 \\ y_1' = v_1 \\ x_2' = u_2 \\ y_2' = v_2 \\ u_1' = u_{1,\text{dot}} \\ v_1' = v_{1,\text{dot}} \\ u_2' = u_{2,\text{dot}} \\ v_2' = v_{2,\text{dot}} \end{cases} \oplus \begin{cases} 2(u_1^2 + v_1^2) + 2(x_1 + a)u_{1,\text{dot}} + 2y_1 v_{1,\text{dot}} = 0, & \text{(Constraint 1)}'' \\ 2(u_2^2 + v_2^2) + 2(x_2 - a)u_{2,\text{dot}} + 2y_2 v_{2,\text{dot}} = 0, & \text{(Constraint 2)}'' \\ 2(\delta u^2 + \delta v^2) + 2\delta x \delta u_{\text{dot}} + 2\delta y \delta v_{\text{dot}} = 0, & \text{(Constraint 3)}'' \\ m_1 u_{1,\text{dot}} + (x_1 + a)\lambda_1 + \delta x \lambda_3 = 0, & \text{(Constraint 4)} \\ m_1 v_{1,\text{dot}} + y_1 \lambda_1 + \delta y \lambda_3 + m_1 g = 0, & \text{(Constraint 5)} \\ m_2 u_{2,\text{dot}} + (x_2 - a)\lambda_2 - \delta x \lambda_3 = 0, & \text{(Constraint 6)} \\ m_2 v_{2,\text{dot}} + y_2 \lambda_2 - \delta y \lambda_3 + m_2 g = 0, & \text{(Constraint 7)} \\ \\ 2(x_1 + a)u_1 + 2y_1 v_1 = 0, & \text{(Constraint 1)}' \\ 2(x_2 - a)u_2 + 2y_2 v_2 = 0, & \text{(Constraint 2)}' \\ 2\delta x \delta u + 2\delta y \delta v = 0, & \text{(Constraint 3)}' \\ \\ (x_1 + a)^2 + y_1^2 - \ell_1^2 = 0, & \text{(Constraint 1)} \\ (x_2 - a)^2 + y_2^2 - \ell_2^2 = 0, & \text{(Constraint 2)} \\ \delta x^2 + \delta y^2 - \ell_3^2 = 0, & \text{(Constraint 3)} \end{cases}$$

The red algebraic equations are the new algebraic equation after the derivation. Notice the four new algebraic constraints added from the introduction of dummy variables. The green algebraic equation is the hidden constraints that is removed from the DAE system but that is satisfied by choosing the right initial conditions.

Step 3: Final Differentiation

To find the right candidate(s) here we write a table of the dependencies. Notice that now the table is larger due to the introduction of new dummy variables and four new constraints

	λ_1	λ_2	λ_3	$u_{1,\text{dot}}$	$v_{1,\text{dot}}$	$u_{2,\text{dot}}$	$v_{2,\text{dot}}$
(Constraint 1)$''$	0	0	0	1	1	0	0
(Constraint 2)$''$	0	0	0	0	0	1	1
(Constraint 3)$''$	0	0	0	1	1	1	1
(Constraint 4)	1	0	1	1	0	0	0
(Constraint 5)	1	0	1	0	1	0	0
(Constraint 6)	0	1	1	0	0	1	0
(Constraint 7)	0	1	1	0	0	0	1

Again all the constraints **cannot** be differentiated. At this point we must check if the jacobian of the constraints is **non-singular**. If is non-singular formally we can solve for λ_1, λ_2, λ_3, $u_{1,\text{dot}}$, $v_{1,\text{dot}}$, $u_{2,\text{dot}}$ and $v_{2,\text{dot}}$ and the DAE is of index-1 and we can stop index reduction. To check if jacobian is singular before to try LU decomposion we analize the **nonzero** structure of the jacobian S

$$S = \begin{pmatrix} \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 \\ \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 \\ 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot \\ 1 & \cdot & 1 & \cdot & 1 & \cdot & \cdot \\ \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot \\ \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 \end{pmatrix}$$

In this form is difficult to understand if a matrix with nonzero at the 1 position can be non-singular.

We use sprank of MATLAB to check if the matrix can be non singular.

```
sparsity_pattern = [
  0 0 0 1 1 0 0;  % Constraint 1''
  0 0 0 0 0 1 1;  % Constraint 2''
  0 0 0 1 1 1 1;  % Constraint 3''
  1 0 1 1 0 0 0;  % Constraint 4
  1 0 1 0 1 0 0;  % Constraint 5
  0 1 1 0 0 1 0;  % Constraint 6
  0 1 1 0 0 0 1]; % Constraint 7

fprintf('Estimated rank = %d\n', sprank(sparsity_pattern) )

> Estimated rank = 7
```

The estimated rank is 7 so that matrix is **structurally non-simgular**.

To better understand how it work consider the Dulmage-Mendelsohn decomposition that find rows and columns permutation for maximum mathcing

```
[p,q,r,s,cc,rr] = dmperm(sparsity_pattern)

sparsity_pattern(p,q)

ans =

     1     0     1     1     0     0     0
     0     1     1     0     0     1     0
     1     0     1     0     1     0     0
     0     0     0     1     1     0     0
     0     0     0     1     1     1     1
     0     0     0     0     0     1     1
     0     1     1     0     0     0     1
```

you can see that the diagonal is all 1 thus a matrix with element non zero only on the diagonal is surely non singular. In practice

- do Dulmage-Mendelsohn decomposition
- Apply rows and columns permutation
- If the diadonal is non zeros (all 1) the matrix is structurally non singular.

Here, here we can do a step more because the system is linear. Let

$$\boldsymbol{X} = \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ u_1 \\ v_1 \\ u_2 \\ v_2 \end{pmatrix}, \quad \boldsymbol{Z} = \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ u_{1,\mathrm{dot}} \\ v_{1,\mathrm{dot}} \\ u_{2,\mathrm{dot}} \\ v_{2,\mathrm{dot}} \end{pmatrix}$$

then constraint become

$$\boldsymbol{A}(\boldsymbol{X}, t)\boldsymbol{Z} = \boldsymbol{b}(\boldsymbol{X}, t)$$

where

$$\boldsymbol{A}(\boldsymbol{X}, t) = \begin{pmatrix} 0 & 0 & 0 & x_1 + a & y_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x_2 - a & y_2 \\ 0 & 0 & 0 & \delta x & \delta y & -\delta x & -\delta y \\ x_1 + a & 0 & \delta x & m_1 & 0 & 0 & 0 \\ y_1 & 0 & \delta y & 0 & m_1 & 0 & 0 \\ 0 & x_2 - a & -\delta x & 0 & 0 & m_2 & 0 \\ 0 & y_2 & -\delta y & 0 & 0 & 0 & m_2 \end{pmatrix}$$

and

$$\boldsymbol{b}(\boldsymbol{X}, t) = -\begin{pmatrix} u_1^2 + v_1^2 \\ u_2^2 + v_2^2 \\ \delta u^2 + \delta v^2 \\ 0 \\ m_1 g \\ 0 \\ m_2 g \end{pmatrix}$$

and the DAE become

$$\left\{\begin{aligned} x_1' &= u_1 \\ y_1' &= v_1 \\ x_2' &= u_2 \\ y_2' &= v_2 \\ u_1' &= u_{1,\mathrm{dot}} \\ v_1' &= v_{1,\mathrm{dot}} \\ u_2' &= u_{2,\mathrm{dot}} \\ v_2' &= v_{2,\mathrm{dot}} \\ \boldsymbol{A}(\boldsymbol{X},t)\boldsymbol{Z} &= \boldsymbol{b}(\boldsymbol{X},t) \end{aligned}\right. \oplus \left\{\begin{aligned} &2(x_1+a)u_1 + 2y_1v_1 = 0, && \text{(Constraint 1)}' \\ &2(x_2-a)u_2 + 2y_2v_2 = 0, && \text{(Constraint 2)}' \\ &2\delta x\delta u + 2\delta y\delta v = 0, && \text{(Constraint 3)}' \\ & && \\ &(x_1+a)^2 + y_1^2 - \ell_1^2 = 0, && \text{(Constraint 1)} \\ &(x_2-a)^2 + y_2^2 - \ell_2^2 = 0, && \text{(Constraint 2)} \\ &\delta x^2 + \delta y^2 - \ell_3^2 = 0, && \text{(Constraint 3)} \end{aligned}\right.$$

where the green part are the hidden constraints that are satisfied by choosing the correct initial conditions.

Completion will follow the conclusion of the second part of the assignment.

13. DAEs from Lagrangian Mechanics

This chapter develops nonlinear DAEs from constrained Lagrangian mechanics and discusses their numerical integration and constraint drift.

Many physical systems contain constraints that must hold throughout the motion. When such systems are formulated through the principle of stationary action, Lagrange multipliers enforce the constraints and the resulting equations are nonlinear DAEs. Unlike ODEs, these systems combine differential dynamics with algebraic geometry and require consistent initial data.

The chapter first derives constrained equations of motion from variational principles, then studies numerical integration, constraint drift, and the structural choices needed for reliable simulation and control.

13.1 Lagrangian DAEs

13.1.1 Foundations

This section primarily draws from the foundational works of [1] and [16]. Mechanics explores the behavior of systems ranging from particles and rigid bodies to continuous media and field theories, such as electromagnetism and gravitation. Since Newton's formulation in the 17th century, mechanics has evolved into two dominant frameworks: *Lagrangian mechanics* and *Hamiltonian mechanics*. Here, we focus on the essentials of Lagrangian mechanics, with further details available in [16].

At the core of Lagrangian mechanics lies the principle of *stationary action* (or *least action*), a variational approach that generalizes Newton's second law, $\boldsymbol{F} = m\boldsymbol{a}$. This principle provides a powerful method for deriving equations of motion for a wide range of physical systems. While both Lagrangian and Hamiltonian mechanics share this foundation, we will restrict our discussion to systems with a finite number of *degrees of freedom*.

13.1.1.1 Coordinates and Energies

For such systems, the configuration is described by a set of *generalized coordinates*:

$$\boldsymbol{q} := (q_1, q_2, \dots, q_n) \in \mathbb{R}^n,$$

where n is the minimum number of independent variables needed to specify the system's state. The choice of $\boldsymbol{q}$ is not unique; different sets of generalized coordinates may describe the same system.

Given a coordinate system $\boldsymbol{q}$, we express

- the system's *kinetic energy* $T(\boldsymbol{q}, \dot{\boldsymbol{q}})$
- and *potential energy* $U(\boldsymbol{q}, t)$

as functions of these variables. The kinetic energy depends on both positions $\boldsymbol{q}$ and velocities $\dot{\boldsymbol{q}}$, while the potential energy generally depends on positions and may also explicitly depend on time t:

$$T : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}, \quad U : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}.$$

Though $\boldsymbol{q}$ may parametrize more complex geometries (i.e., *manifolds*), in many practical cases, the configuration space can be locally approximated as Euclidean.

13.1.1.2 Lagrangian Function

The *Lagrangian function*, $L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t)$, is defined as the difference between kinetic and potential energy:

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) := T(\boldsymbol{q}, \dot{\boldsymbol{q}}) - U(\boldsymbol{q}, t). \tag{13.1}$$

This function encapsulates the entire dynamics of the system and serves as the foundation for the Lagrangian formulation. Its domain consists of positions, velocities, and time:

$$L : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}.$$

13.1.1.3 Euler–Lagrange Equations

The equations of motion follow from the *Euler-Lagrange equations*, derived by applying the principle of **stationary action**:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = 0, \quad i = 1, \dots, n.$$

These equations provide a systematic method for obtaining the equations of motion for mechanical systems, from simple particles to more complex rigid-body dynamics.

The Lagrangian framework offers an elegant and unified approach to modeling physical phenomena, leveraging the principle of stationary action. By encoding dynamics in a single function $L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t)$, it simplifies problem-solving and provides deep insights into the structure of classical mechanics.

13.1.2 Euler–Lagrange equations

The calculus of variations used to derive the Euler–Lagrange equations is developed in Section 16.2. For a Lagrangian $L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t)$, stationarity of the action with fixed endpoints gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{\boldsymbol{q}}}\right)^{\mathsf{T}} - \left(\frac{\partial L}{\partial \boldsymbol{q}}\right)^{\mathsf{T}} = \mathbf{0}. \tag{13.2}$$

For non-conservative generalized forces $\boldsymbol{\tau}$, the right-hand side becomes $\boldsymbol{\tau}$. The constrained form, which introduces Lagrange multipliers and algebraic equations, is developed in the section on constrained mechanics below.

13.1.3 Equations of Motion

While the Euler-Lagrange equations provide a general framework, it is often helpful to consider explicit representations for mechanical systems with specific characteristics. For systems where

- the kinetic energy $T(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is a quadratic form in the generalized velocities $\dot{\boldsymbol{q}}$,

we will derive explicit expressions for the equations of motion. This assumption is broadly applicable, as any mechanical system with a finite number of degrees of freedom can be described by a quadratic kinetic energy.

13.1.3.1 Kinetic Energy

From this foundational assumption, many interesting properties emerge. For a more comprehensive treatment, refer to [22]. The kinetic energy can be written as:

$$T(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{2} \sum_{j=1}^{n} \sum_{k=1}^{n} m_{jk}(\boldsymbol{q}) \dot{q}_j \dot{q}_k = \frac{1}{2} \dot{\boldsymbol{q}}^\mathsf{T} \boldsymbol{M}(\boldsymbol{q}) \dot{\boldsymbol{q}},$$

where $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n \times n}$ is the symmetric positive definite matrix known as the *mass matrix*. The entries $m_{jk}(\boldsymbol{q})$ of $\boldsymbol{M}(\boldsymbol{q})$ are referred to as the *generalized masses* or *generalized inertias*.

R The matrix $\boldsymbol{M}(\boldsymbol{q})$ is symmetric because the quadratic form associated with a matrix depends only on its symmetric part.

13.1.3.2 Derivation

We now aim to derive the explicit equations of motion for mechanical systems that exhibit *quadratic* kinetic energy. This process begins with the Euler-Lagrange equations, given in Equation 13.2.

First Term: Time Derivative of the Kinetic Energy

Consider the term:

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{q}_i}$$

This term can be expressed as:

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{q}_i} = \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial T}{\partial \dot{q}_i} = \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial}{\partial \dot{q}_i} \left(\frac{1}{2} \sum_{j=1}^{n} \sum_{k=1}^{n} m_{jk}(\boldsymbol{q}) \dot{q}_j \dot{q}_k \right),$$

for $i = 1, \ldots, n$. By exploiting the linearity of summation, we can interchange the order of the partial derivative and the summation:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_i} &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n} m_{jk}(\boldsymbol{q})\frac{\partial \dot{q}_j}{\partial \dot{q}_i}\dot{q}_k + \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n} m_{jk}(\boldsymbol{q})\dot{q}_j\frac{\partial \dot{q}_k}{\partial \dot{q}_i}\right)\\ &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{2}\sum_{k=1}^{n} m_{ik}(\boldsymbol{q})\dot{q}_k + \frac{1}{2}\sum_{j=1}^{n} m_{ji}(\boldsymbol{q})\dot{q}_j\right)\\ &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\sum_{j=1}^{n} m_{ij}(\boldsymbol{q})\dot{q}_j\right).\end{aligned} \tag{13.3}$$

In the last step, we used the fact that:

$$\frac{\partial \dot{q}_j}{\partial \dot{q}_i} = \begin{cases} 1 & i = j \\ 0 & \text{otherwise} \end{cases}$$

and the symmetry condition $m_{ij}(\boldsymbol{q}) = m_{ji}(\boldsymbol{q})$ for $i, j = 1, \dots, n$.
Taking the time derivative of this expression yields:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_i} = \sum_{j=1}^{n} m_{ij}(\boldsymbol{q})\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n}\frac{\partial m_{ij}(\boldsymbol{q})}{\partial q_k}\dot{q}_k\dot{q}_j. \tag{13.4}$$

Second Term: Derivative of Potential Energy

Next, we compute the term $\frac{\partial L}{\partial q_i}$, for $i = 1, \dots, n$:

$$\frac{\partial L}{\partial q_i} = \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n}\frac{\partial m_{jk}(\boldsymbol{q})}{\partial q_i}\dot{q}_k\dot{q}_j - \frac{\partial U(\boldsymbol{q}, t)}{\partial q_i}. \tag{13.5}$$

We define the following:

$$g_i(\boldsymbol{q}) := \frac{\partial U(\boldsymbol{q}, t)}{\partial q_i},$$

for $i = 1, \dots, n$, and the vector of generalized forces:

$$\boldsymbol{g}(\boldsymbol{q}) := (g_1(\boldsymbol{q}), \dots, g_n(\boldsymbol{q})) \in \mathbb{R}^n.$$

Equations of Motion

Combining Equation 13.4 and Equation 13.5, we obtain the componentwise equations of motion:

$$\begin{aligned}\sum_{j=1}^{n} m_{ij}(\boldsymbol{q})\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n}\frac{\partial m_{ij}(\boldsymbol{q})}{\partial q_k}\dot{q}_j\dot{q}_k \\ - \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n}\frac{\partial m_{jk}(\boldsymbol{q})}{\partial q_i}\dot{q}_j\dot{q}_k + g_i(\boldsymbol{q}) = \tau_i,\end{aligned} \tag{13.6}$$

where τ_i represents the generalized forces acting on the i-th degree of freedom.

Introducing Inertia Terms

To simplify, we define the following term:

$$h_{ijk}(\boldsymbol{q}) := \frac{\partial m_{ij}(\boldsymbol{q})}{\partial q_k} - \frac{1}{2}\frac{\partial m_{jk}(\boldsymbol{q})}{\partial q_i}. \tag{13.7}$$

Substituting this into Equation 13.6, we get a more compact form of the equations of motion:

$$\sum_{j=1}^{n} m_{ij}(\boldsymbol{q})\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}(\boldsymbol{q})\dot{q}_j\dot{q}_k + g_i(\boldsymbol{q}) = \tau_i, \tag{13.8}$$

Matrix Formulation

The equations of motion can be rewritten in compact matrix notation:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau},$$

where $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is the matrix that captures the centrifugal and Coriolis effects. The exact form of $\boldsymbol{C}$ depends on the system's geometry and dynamics.

R It's crucial to note that the number of degrees of freedom in a system cannot be assumed to be $n - m$ without certain regularity conditions on the constraint function $\boldsymbol{\phi}(\boldsymbol{q}, t)$. Specifically, if:

$$\operatorname{rank}\left(\frac{\partial \boldsymbol{\phi}(\boldsymbol{q}, t)}{\partial \boldsymbol{q}}\right) = m,$$

then the system has exactly $n - m$ degrees of freedom, assuming that $\boldsymbol{\phi}$ is differentiable.

13.1.4 Constrained Mechanics

For mechanical systems subject to constraints, we use the method of Lagrange multipliers to handle the constraint equation

$$\boldsymbol{\phi}(\boldsymbol{q}, t) = 0.$$

The *modified action* $\bar{S}$ is defined as:

$$\bar{S} := \int_{t_1}^{t_2} \left(L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) + \boldsymbol{\lambda}^\intercal \boldsymbol{\phi}(\boldsymbol{q}, t)\right) \mathrm{d}t,$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of Lagrange multipliers. This formulation is critical for analyzing the dynamics of constrained systems.

Theorem 13.1.1 — Constrained Stationary Action. The motion of mechanical systems transitioning from configuration $\boldsymbol{q}_1$ to $\boldsymbol{q}_2$ over the time interval $t \in [t_1, t_2]$ corresponds to the extremal curves of the functional defined in the equation above.

In other words, for small variations in the curve $\boldsymbol{q}(\cdot)$, the modified action $\bar{S}$ becomes stationary:

$$\delta \bar{S} = 0.$$

R It is important to note that we implicitly assume that the initial and final configurations, $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$, satisfy the constraint equations:

$$\boldsymbol{\phi}(\boldsymbol{q}_1, t_1) = \boldsymbol{\phi}(\boldsymbol{q}_2, t_2) = 0.$$

Thus, we can perform variations as we did previously for systems with a minimal set of coordinates. However, in this case, $\bar{S}$ also depends on the Lagrange multipliers. The variational operator δ applies variations to $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, and $\boldsymbol{\lambda}$.

The result of the variation is:

$$\begin{aligned}\delta\bar{S} &= \int_{t_1}^{t_2} \delta L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) + \delta\boldsymbol{\lambda}^\mathsf{T}\boldsymbol{\phi}(\boldsymbol{q}, t) + \boldsymbol{\lambda}^\mathsf{T}\delta\boldsymbol{\phi}(\boldsymbol{q}, t)\,\mathrm{d}t \\ &= \int_{t_1}^{t_2} \left(\frac{\partial L}{\partial \boldsymbol{q}} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\boldsymbol{q}}} + \boldsymbol{\lambda}^\mathsf{T}\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}\right)\delta\boldsymbol{q} + \delta\boldsymbol{\lambda}^\mathsf{T}\boldsymbol{\phi}(\boldsymbol{q}, t)\,\mathrm{d}t + \left[\frac{\partial L}{\partial \dot{\boldsymbol{q}}}\delta\boldsymbol{q}\right]_{t_1}^{t_2} \\ &= \int_{t_1}^{t_2} \begin{bmatrix}\frac{\partial L}{\partial \boldsymbol{q}} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\boldsymbol{q}}} + \boldsymbol{\lambda}^\mathsf{T}\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} & \boldsymbol{\phi}^\mathsf{T}\end{bmatrix}\begin{bmatrix}\delta\boldsymbol{q} \\ \delta\boldsymbol{\lambda}\end{bmatrix}\,\mathrm{d}t = 0.\end{aligned}$$

We use the fact that the variations at the endpoints are zero: $\delta\boldsymbol{q}_1 = \delta\boldsymbol{q}_2 = 0$.

Finally, applying the *fundamental lemma of calculus of variations* to the above expression gives:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L^\mathsf{T}}{\partial \dot{\boldsymbol{q}}} - \frac{\partial L^\mathsf{T}}{\partial \boldsymbol{q}} = \frac{\partial \boldsymbol{\phi}^\mathsf{T}}{\partial \boldsymbol{q}}\boldsymbol{\lambda},$$

$$\boldsymbol{\phi}(\boldsymbol{q}, t) = 0.$$

The equations above describe the dynamics of a mechanical system subject to holonomic constraints. The term

$$\frac{\partial \boldsymbol{\phi}^\mathsf{T}}{\partial \boldsymbol{q}}\boldsymbol{\lambda}$$

acts as a generalized force ensuring that the constraints $\boldsymbol{\phi}(\boldsymbol{q}, t) = 0$ are satisfied. Unlike the equations derived from the Euler-Lagrange equations, this system involves both *differential* and *algebraic* equations, making it a *Differential Algebraic Equation* (DAE). Additionally, the system does not provide a direct method for determining the Lagrange multipliers.

When external non-conservative forces are present, the equations of motion modify to:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L^\mathsf{T}}{\partial \dot{\boldsymbol{q}}} - \frac{\partial L^\mathsf{T}}{\partial \boldsymbol{q}} = \frac{\partial \boldsymbol{\phi}^\mathsf{T}}{\partial \boldsymbol{q}}\boldsymbol{\lambda} + \boldsymbol{\tau},$$

$$\boldsymbol{\phi}(\boldsymbol{q}, t) = 0,$$

where $\boldsymbol{\tau} \in \mathbb{R}^n$ is the vector of generalized non-conservative forces.

▪ **Example 13.1 — Pendulum equations.** The variational derivation of the pendulum equations, in both minimal angular coordinates and redundant Cartesian coordinates, is given in Section 16.1. Here we record the resulting equations for use in the DAE discussion.

With the minimal coordinate θ and pendulum length l,

$$\ddot{\theta} + \frac{g}{l}\sin\theta = 0.$$

In redundant Cartesian coordinates $\boldsymbol{q} = (x, y)^\mathsf{T}$, with Lagrange multiplier λ, the pendulum equations are:

$$\begin{aligned} m\ddot{x} &= 2x\lambda, \\ m\ddot{y} &= 2y\lambda - mg, \\ x^2 + y^2 &= l^2. \end{aligned}$$

▪

13.2 Numerical Integration

As pointed out in [20], both from a theoretical and numerical point of view, DAEs are much more challenging than ordinary differential equations (ODEs). Some DAEs can be solved using numerical methods for stiff systems, but others cannot, and they are very sensitive to the step size, which may lead to large errors in the solution and numerical instability. For DAEs arising from constrained mechanical systems, a fairly general form is the following:

$$\begin{cases} \boldsymbol{M}(\boldsymbol{q}, \boldsymbol{q}', t)\boldsymbol{q}'' - \dfrac{\partial \boldsymbol{\phi}(\boldsymbol{q}, t)}{\partial \boldsymbol{q}}^T \boldsymbol{\lambda} = \boldsymbol{n}(\boldsymbol{q}, \boldsymbol{q}', t), \\ \boldsymbol{\phi}(\boldsymbol{q}, t) = 0 \end{cases}$$

where the term

$$\boldsymbol{n}(\boldsymbol{q}, \boldsymbol{q}', t) \in \mathbb{R}^n$$

collects all the contributions coming from Coriolis and centrifugal effects, conservative and non-conservative forces. It can be defined as follows:

$$\boldsymbol{n}(\boldsymbol{q}, \boldsymbol{q}', t) = \boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{q}, \boldsymbol{q}')\boldsymbol{q}' - \boldsymbol{g}(\boldsymbol{q}).$$

It is well known that the equations of motion for a mechanical system are a system of second-order (possibly non-linear) differential equations. However, for clarity and generality, it is useful to consider a *first-order representation*. For this purpose, let us define $\boldsymbol{v} \in \mathbb{R}^n$ as *generalized velocities*, i.e.,

$$\boldsymbol{v} := \boldsymbol{q}'.$$

Thus, substituting the above equation into the general form of the equation of motion results in:

$$\begin{cases} \boldsymbol{q}' = \boldsymbol{v}, \\ \boldsymbol{M}(\boldsymbol{q}, \boldsymbol{v}, t)\boldsymbol{v}' - \dfrac{\partial \boldsymbol{\phi}(\boldsymbol{q}, t)}{\partial \boldsymbol{q}}^T \boldsymbol{\lambda} = \boldsymbol{n}(\boldsymbol{q}, \boldsymbol{v}, t), \\ \boldsymbol{\phi}(\boldsymbol{q}, t) = 0. \end{cases} \tag{13.9}$$

Notice that the vector

$$\begin{pmatrix} \boldsymbol{q} \\ \boldsymbol{v} \end{pmatrix} \in \mathbb{R}^{2n}$$

represents the *state* of a mechanical system, namely the pair of *generalized position* and *generalized velocity*.

13.2.1 ODE Reduction

Now, a good question is how to solve the problem in the equation above. The issue is how to determine the value for the Lagrange multiplier $\boldsymbol{\lambda}$. One possible approach is to transform the problem into a system of pure differential equations. In order to do so, we need to differentiate the constraint $\boldsymbol{\phi}(\boldsymbol{q}, t)$, which yields:

$$\frac{\mathrm{d}\boldsymbol{\phi}}{\mathrm{d}t} = \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}\boldsymbol{v} + \frac{\partial \boldsymbol{\phi}}{\partial t} = \boldsymbol{0}.$$

Notice that when the constraint does not depend on time, this expression indicates that the generalized velocity $\boldsymbol{v}$ must be orthogonal to the gradient of the constraint. This also provides the opportunity to note that the constraint $\boldsymbol{\phi}(\boldsymbol{q}, t)$ imposes relationships at the velocity and

acceleration levels, restricting the set of feasible velocities. The constraints that we can obtain by differentiating the constraint are sometimes called *hidden constraints*.

Finally, notice that the equation above is still algebraic in the $\boldsymbol{v}$ coordinate. Thus, in order to obtain a differential relationship, let us differentiate again:

$$\frac{\mathrm{d}^2\boldsymbol{\phi}(\boldsymbol{q},t)}{\mathrm{d}t^2} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}\boldsymbol{v}\right) + \frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}\boldsymbol{v}' + \frac{\partial^2\boldsymbol{\phi}}{\partial t^2} = \boldsymbol{0}.$$

Notice that now this expression does not impose any algebraic constraint on the state of the mechanical system $(\boldsymbol{q};\boldsymbol{v})$. Substituting the constraint $\boldsymbol{\phi}(\boldsymbol{q},t) = \boldsymbol{0}$ with the one derived above results in the following modified problem:

$$\begin{cases} \boldsymbol{q}' = \boldsymbol{v}, \\ \boldsymbol{M}(\boldsymbol{q},\boldsymbol{v},t)\boldsymbol{v}' - \dfrac{\partial\boldsymbol{\phi}(\boldsymbol{v},t)}{\partial\boldsymbol{q}}^T\boldsymbol{\lambda} = \boldsymbol{n}(\boldsymbol{q},\boldsymbol{v},t), \\ -\dfrac{\partial\boldsymbol{\phi}(\boldsymbol{v},t)}{\partial\boldsymbol{q}}\boldsymbol{v}' = \boldsymbol{m}(\boldsymbol{q},\boldsymbol{v},t). \end{cases} \tag{13.10}$$

Here, to simplify the notation, we introduce the map

$$\boldsymbol{m}(\boldsymbol{q},\boldsymbol{v},t) := \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}\boldsymbol{v}\right) + \frac{\partial^2\boldsymbol{\phi}}{\partial t^2}, \quad \boldsymbol{m}(\boldsymbol{q},\boldsymbol{v},t) \in \mathbb{R}^m.$$

Even though differentiating the constraint seems promising, the problem in the above equations and the one derived earlier are *not equivalent*. This difference leads to the so-called **drift effect**, which will be discussed in the next section. For now, let's continue along the current path. The system of equations can also be written in compact matrix notation as follows:

$$\begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}(\boldsymbol{q},\boldsymbol{v},t) & -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}^T(\boldsymbol{q},t) \\ \boldsymbol{0} & -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}(\boldsymbol{q},t) & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}' \\ \boldsymbol{v}' \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{n}(\boldsymbol{q},\boldsymbol{v},t) \\ \boldsymbol{m}(\boldsymbol{q},\boldsymbol{v},t) \end{bmatrix}.$$

Now, it's easy to see that if the matrix on the left-hand side is invertible, we can express the system in explicit form and use this representation to perform numerical simulations. The matrix is invertible if and only if the sub-block:

$$\begin{bmatrix} \boldsymbol{M} & -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}^T \\ -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}} & \boldsymbol{0} \end{bmatrix} \in \mathbb{R}^{(n+m)\times(n+m)},$$

is non-singular (i.e., full rank).

Recall that $\boldsymbol{M}$ is positive definite (and thus non-singular), so that

$$\begin{bmatrix} \boldsymbol{M} & -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}^T \\ -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}} & \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ -\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}\boldsymbol{M}^{-1} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{M}^{-1}\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}^T \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$$

where

$$\boldsymbol{D} = \frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}\boldsymbol{M}^{-1}\frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{q}}^T$$

and thus the block is non-singular if $\boldsymbol{D}$ is non-singular.

This is not generally true, but assuming that the constraints are locally linearly independent (i.e., $\forall\boldsymbol{q} \in \mathbb{R}^n, \forall t$):

$$\operatorname{rank}\left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}\right) = m,$$

then $\boldsymbol{D}$ is non-singular. Thus, the system can be expressed in *explicit first-order form* as follows:

$$\begin{bmatrix} \boldsymbol{q}' \\ \boldsymbol{v}' \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & 0 & 0 \\ 0 & \boldsymbol{M}(\boldsymbol{q}, \boldsymbol{v}, t) & -\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^T(\boldsymbol{q}, t) \\ 0 & -\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}(\boldsymbol{q}, t) & 0 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{n}(\boldsymbol{q}, \boldsymbol{v}, t) \\ \boldsymbol{m}(\boldsymbol{q}, \boldsymbol{v}, t) \end{bmatrix}.$$

An explicit expression for the inverse matrix can be obtained through tedious calculations (a calculation is done when numerical simulation are done). Regardless of the specific expression for the solution, the important point to emphasize is that by differentiating the constraint, we were able to find an explicit expression for the Lagrange multipliers. For this purpose, we differentiated twice and obtained the equation:

$$\boldsymbol{\lambda} = -\left(\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}^{-1} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}^T\right)^{-1} \left(\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}}\boldsymbol{v} + \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{q}} \boldsymbol{M}^{-1} \boldsymbol{n}(\boldsymbol{q}, \boldsymbol{v}, t) + \frac{\partial^2 \boldsymbol{\phi}}{\partial t^2}\right).$$

Technically, mechanical systems subject to holonomic constraints are systems of differential-algebraic equations (DAEs) of *differential index* 3. The differential index is one plus the number of differentiations of the constraint that are needed in order to eliminate the Lagrange multipliers. Alternatively, it is the number of times that we need to differentiate the constraint in order to obtain a differential equation for the Lagrange multiplier.

13.2.2 Constraint Drift

In this section we want investigate the effects of the differentiating process done to transform problem in Equation 13.9 into problem Equation 13.10. Unfortunately this method is *not* priceless, indeed formally we substitute an algebraic constraint with its second derivative w.r.t time. To better understand the consequences let us consider a fictitious constraint $\widetilde{\boldsymbol{\phi}}(\boldsymbol{q}, t)$ of the form,

$$\widetilde{\boldsymbol{\phi}}(\boldsymbol{q}, t) := \boldsymbol{\phi}(\boldsymbol{q}, t) + \boldsymbol{a}_0 + \boldsymbol{a}_1 t.$$

It is easy to see that consider a mechanical system subject to constraint $\phi(\boldsymbol{q}, t)$ or $\widetilde{\boldsymbol{\phi}}(\boldsymbol{q}, t)$ differentiating twice is exactly the same, i.e.,

$$\frac{\mathrm{d}^2 \boldsymbol{\phi}}{\mathrm{d}t^2} = \frac{\mathrm{d}^2 \widetilde{\boldsymbol{\phi}}}{\mathrm{d}t^2}.$$

This fact produces the *drift effect*, i.e., the original constraint $\boldsymbol{\phi}(\boldsymbol{q}, t)$ is violated with an error growing with time. This error grows in the best case linearly, but it can be faster due to *numerical* errors. The reason of this behaviour is the term $\boldsymbol{a}_1 t$ which does not have any effect in the formulation Equation 13.10. Note also that this is not an undesired effect coming from not enough precision of the numerical scheme, but it is intrinsic effect due to substitution of the constraint $\boldsymbol{\phi}$ with its second derivative w.r.t time. A simple solution to this undesired effect is the Baumgarte's stabilization discussed in Baumgarte Stabilization.

▪ **Example 13.2** Continuing Example 13.2 we provide a first order representation for the pendulum; setting $\boldsymbol{v} = \boldsymbol{q}'$ results,

$$\boldsymbol{v} = \boldsymbol{q}',$$
$$m\boldsymbol{v}' + mg\boldsymbol{e}_2 = 2\boldsymbol{q}\lambda,$$
$$\boldsymbol{q}^T\boldsymbol{q} - \ell^2 = 0.$$

Then in order to eliminate the Lagrange multiplier let us differentiate the constraint,

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} = 2\boldsymbol{q}^T\boldsymbol{v} = 0.$$

Notice that as expected velocity $\boldsymbol{v}$ and position $\boldsymbol{q}$ are orthogonal each other. Differentiating again we obtain

$$\frac{\mathrm{d}^2\phi}{\mathrm{d}t^2} = 2(\boldsymbol{q}')^T\boldsymbol{v} + 2\boldsymbol{q}^T\boldsymbol{v}' = 2\boldsymbol{v}^T\boldsymbol{v} + 2\boldsymbol{q}^T\boldsymbol{v}' = 0.$$

Now we can write the problem in shape of Equation 13.11,

$$\begin{bmatrix} \boldsymbol{I} & 0 & 0 \\ 0 & m\boldsymbol{I} & -2\boldsymbol{q} \\ 0 & -2\boldsymbol{q}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q}' \\ \boldsymbol{v}' \\ \lambda \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ -mg\boldsymbol{e}_2 \\ 2\boldsymbol{v}^T\boldsymbol{v} \end{bmatrix}, \tag{13.11}$$

▪

where $\boldsymbol{I} \in \mathbb{R}^{2\times 2}$ is the identity matrix of dimension 2. The explicit value for the Lagrange multipliers is the following

$$\lambda = -\frac{m}{2\boldsymbol{q}^T\boldsymbol{q}}\left(\boldsymbol{v}^T\boldsymbol{v} - g\boldsymbol{q}^T\boldsymbol{e}_2\right) = -\frac{m}{2\ell^2}\left(\boldsymbol{v}^T\boldsymbol{v} - gq_2\right). \tag{13.12}$$

To better appreciate the drift effect we can simulate Equation 13.11 using an ODE solver: at each step λ is computed from Equation 13.12 and only $(\boldsymbol{q}, \boldsymbol{v})$ are integrated. The simulation runs for 50 seconds with

- $g = 9.81$, $m = 2$, $\ell = 2.5$;
- $\boldsymbol{q}_0 \approx (0.86; 2.35) \in \mathbb{R}^2$ (projected exactly onto $\|\boldsymbol{q}\| = \ell$);
- $\boldsymbol{q}_0' = \boldsymbol{v}_0 = (0, 0) \in \mathbb{R}^2$;
- $\lambda_0 = 3.69$,

which are all consistent with the constraint and with Equation 13.12. The results are shown in Figure 13.1.

Both panels show that the drift is not a defect of a particular integrator. Every numerical error introduces a small perturbation of $\boldsymbol{a}_0 + \boldsymbol{a}_1 t$ type, and the reduced system has no mechanism to pull the solution back onto the constraint, so the violation accumulates.

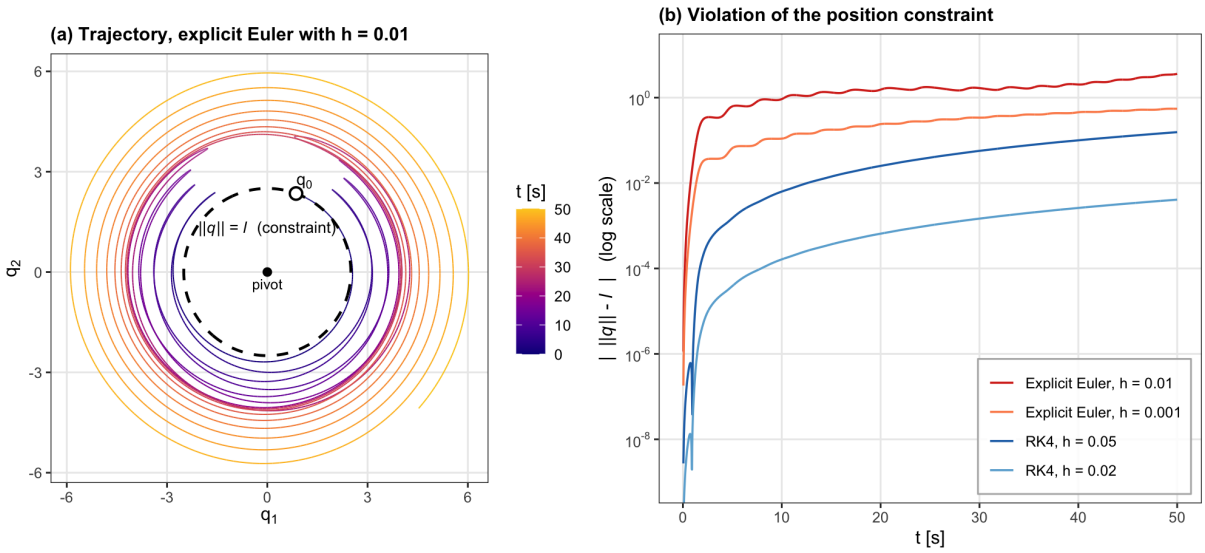


Figure 13.1: Pendulum integrated as an ODE after replacing the constraint with its second derivative. (a) The trajectory computed with explicit Euler ($h = 0.01$) spirals away from the circle $\|\boldsymbol{q}\| = \ell$. (b) The constraint violation grows with time for every method and step size: a more accurate scheme only slows the drift down.

14.1 The Brachistochrone
14.2 Admissible Function Spaces
14.3 Function-Space Geometry
14.4 First Variation
14.5 Integral Differentiation
14.6 Local Minima and Existence
14.7 Sobolev Spaces

IV Calculus of Variations

15 Fundamental Lemma 197
15.1 Statement
15.2 Sign Persistence
15.3 Proof Strategy: A Triangular Test Function
15.4 Smooth Bump Function
15.5 Rigorous Proof
15.6 Interpretation

16 Euler–Lagrange Equations 203
16.1 Pendulum Example
16.2 General Formulation
16.3 First Integrals and Vectors
16.4 Brachistochrone
16.5 Second Variation and Legendre Condition
16.6 Weierstrass Necessary Condition

17 Higher-Order Functionals 213
17.1 Euler–Poisson Equation
17.2 Natural Boundary Conditions
17.3 Corner Conditions
17.4 Endpoint Constraints
17.5 Boundary Multipliers

18 Isoperimetric Constraints 227
18.1 Feasible Variations
18.2 Finite Differences
18.3 Lagrange-Multiplier Method
18.4 General Formulation
18.5 Normal and Abnormal Multiplier Rules

19 Path Constraints 235
19.1 Discrete Derivation
19.2 Continuous Multipliers
19.3 Inequality Path Constraints
19.4 Regularity of Path-Constraint Multipliers

20 Summary 241
20.1 Admissible Spaces
20.2 General Problem
20.3 Path Constraints
20.4 Stationarity and Minima
20.5 Classification and Existence
20.6 Constraint Qualifications

Prerequisites, objectives, and expected outcomes

Prerequisites: multivariable calculus, integration by parts, and basic ODE boundary-value problems.
Objectives: compute first variations, derive Euler–Lagrange equations and natural boundary conditions, and handle integral and path constraints.
Expected outcomes: the reader can translate a functional minimization problem into necessary conditions, distinguish fixed and free endpoint data, and check a candidate extremal.

Contents

- Variational Foundations
- Fundamental Lemma
- Euler–Lagrange Equations
- Higher-Order Functionals
- Isoperimetric Constraints
- Path Constraints
- Summary
- Worked Exercises (companion volume)

14. Variational Foundations

A classical problem in minimization is the brachistochrone, i.e., find the curve (meaning determining the fastest trajectory) to move from a point A to a (lower) point B considering that the only action is the one of the gravity.

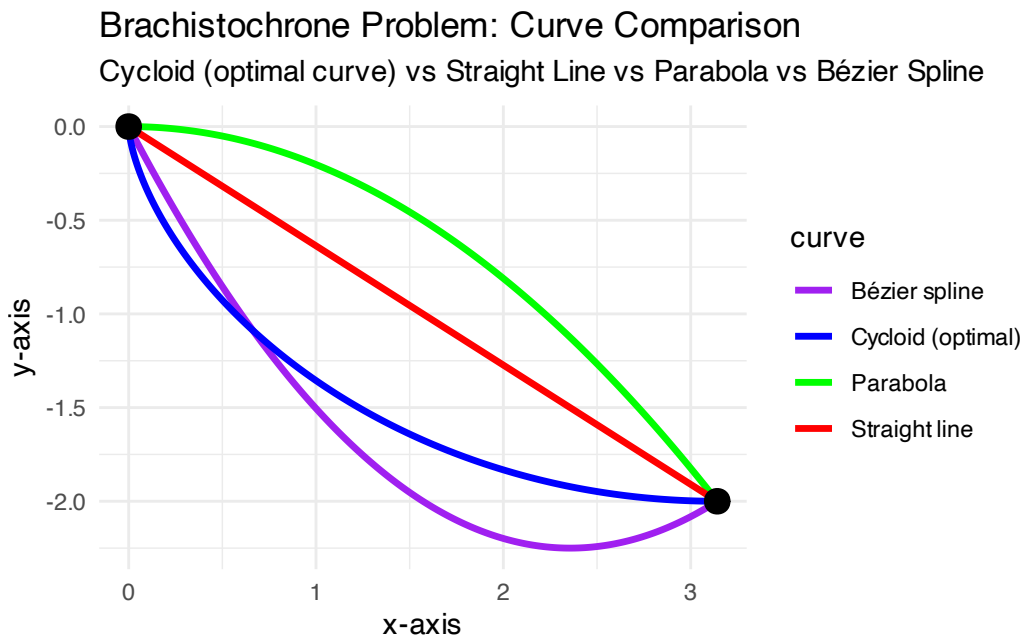


Figure 14.1: Candidate trajectories for the brachistochrone problem.

example of trajectories that an object can follow for moving from point A to B. Mathematically the problem is:

14.1 The Brachistochrone

Definition 14.1.1 — Problem Statement. Determine the curve

$$C : [a, b] \to \mathbb{R}$$

such that

$$C(a) = y_a \quad \text{and} \quad C(b) = y_b$$

that minimize the function $T(C)$ which represent the time to travel from $A = (a, y_a)$ to $B = (b, y_b)$. Thus, given $T(C)$ the problem become:

$$\text{minimize } T(C) \qquad \text{for all possible curves } C$$

A **function** takes a number as input and returns a number, like $f(x) = xe^x$. A **functional** is instead something that takes as input a function and returns a number and an example is

$$\mathcal{F}(x) = \int_a^b x(t)\,\mathrm{d}t$$

where x is a function. Considering for example $x(t) = t^2$ the previous functional becomes

$$\mathcal{F}(x) = \int_a^b t^2\,\mathrm{d}t = \frac{t^3}{3}\Big|_a^b = \frac{b^3 - a^3}{3}$$

Another example of functional can be

$$\mathcal{G}(z) = z'(0) \int_0^1 z^2(t)\,\mathrm{d}t$$

and this expression can be evaluated for every generic function $z(t)$.

The question connected to Definition 14.1.1 is how to define a *minimum* for a functional; in order to do so we have to firstly understand how to minimize a simple function and in particular given a function

$$f : A \subseteq \mathbb{R} \to \mathbb{R}$$

has a minimum in the point $x^\star$ if

$$f(x^\star) \leq f(x) \qquad \forall x \in A$$

Similarly given a function $\mathcal{F}(x)$ a *point* $x^\star$ (that in reality is a function) is a minimum if

$$\mathcal{F}(x^\star) \leq \mathcal{F}(x) \qquad \forall x \in ?$$

In this case we have to specify the *class* (functional space) of functions we are considering, as example x can be a function is continuous in the domain $[a, b]$ and so we can define

$$x \in C([a, b]) = \left\{ g : [a, b] \to \mathbb{R} \mid g \text{ is continuous} \right\}$$

In general changing the *domain* of the functional $\mathcal{F}$ may change the problem.

14.2 Admissible Function Spaces

▪ **Example 14.1 — Domain-Dependence Example.** For the equation $f(z) = z^2 + 2 = 0$, the roots exist over the complex numbers, $z = \pm \mathrm{i}\sqrt{2}$, but there is no solution over the real numbers.

Considering now the functional $\mathcal{F}$ defined as

$$\mathcal{F}(x) = \int_{-1}^1 \left(x(t) - |t|\right)^2 \mathrm{d}t$$

The $|t|$ introduce a cuspid in $t = 0$ that cannot be derived. If we minimize the functional in the continuous interval $x \in C([-1, 1])$, the solution is the function $x^\star(t) = |t|$, in fact

$$\mathcal{F}(x^\star) = \int_{-1}^1 \left(|t| - |t|\right)^2 \mathrm{d}t = \int_{-1}^1 0\,\mathrm{d}t = 0$$

Choosing any other continuous function will result in a functional with a positive value.

Considering now to minimize the function in the domain of functions with continuous derivatives, and so $x \in C^1([-1,1])$. In this case $|t| \notin C^1([-1,1])$ (due to the cuspid). An approximation of the function $|t|$ that is continuous with also continuous first derivative is the function

$$x_\varepsilon(t) = \begin{cases} t & t \geq \varepsilon \\ \frac{1}{2}\left(\frac{t^2}{\varepsilon} + \varepsilon\right) & -\varepsilon < t < \varepsilon \\ -t & t \leq -\varepsilon \end{cases}$$

As $\varepsilon \to 0$, this family approaches $|t|$ and

$$\mathcal{F}(x_\varepsilon) = \int_{-\varepsilon}^{\varepsilon} \left(\frac{t^2}{2\varepsilon} + \frac{\varepsilon}{2} - |t|\right)^2 \mathrm{d}t = \frac{\varepsilon^3}{10}$$

Therefore the infimum over $C^1([-1,1])$ is zero, but it is not attained: a minimizer in this space would have to coincide with $|t|$, which is not in $C^1([-1,1])$. This example distinguishes an infimum from an attained minimum.

▪

14.2.1 Basic Function Spaces

- Functions $f : [a,b] \mapsto \mathbb{R}$ continuous
- Functions $f : [a,b] \mapsto \mathbb{R}$ which are C^k (continuous with derivative up to order k).
- Functions $f : [a,b] \mapsto \mathbb{R}$ piecewise continuous

14.2.2 Lebesgue Spaces

- Space $L_1([a,b])$ of functions $f : [a,b] \mapsto \mathbb{R}$ which are module integrable
$$\int_a^b |f(t)|\,\mathrm{d}t < \infty$$
[the integral must exists and is finite]
- Space $L_2([a,b])$ of functions $f : [a,b] \mapsto \mathbb{R}$ which are square integrable
$$\int_a^b |f(t)|^2\,\mathrm{d}t < \infty$$
[the integral must exists and is finite]

▪ **Example 14.2** Let $f : [0,1] \mapsto \mathbb{R}$, $f(x) = 1/x$ is not $L_1([0,1])$ in fact

$$\int_0^1 \frac{1}{x}\mathrm{d}x \to \infty$$

The function $g : [0,1] \mapsto \mathbb{R}$, $g(x) = 1/\sqrt{x}$, is in $L_1([0,1])$ in fact

$$\int_0^1 \frac{1}{\sqrt{x}}\mathrm{d}x = \left[2\sqrt{|x|}\right]_{x=0}^{x=1} = 2 < \infty$$

however this function $g(x)$ is not in $L_2([0,1])$, in fact

$$\int_0^1 g(x)^2\mathrm{d}x = \int_0^1 \frac{1}{(\sqrt{x})^2}\mathrm{d}x = \int_0^1 \frac{1}{|x|}\mathrm{d}x \to \infty$$

the function $h(x) = x^{-1/4}$ is in $L_2([0,1])$, in fact

$$\begin{aligned}\int_0^1 h(x)^2 \mathrm{d}x &= \int_0^1 (x^{-1/4})^2 \mathrm{d}x \\ &= \int_0^1 \frac{1}{\sqrt{x}} \mathrm{d}x \\ &= \left[2\sqrt{|x|}\right]_{x=0}^{x=1} \\ &= 2 < \infty\end{aligned}$$

∎

14.3 Function-Space Geometry

To solve the problem of minimizing a functional we can see some relations with the linear algebra. The domain of the functional can be in fact see as a **functional space** $\mathbb{V}$ analogous to the vectorial one which the vectors are represented by the functions and the scalars are represented by real values. With this definition example of functional spaces might be

$$\mathbb{V}_1 = \left\{ f \,:\, [0,1] \mapsto \mathbb{R} \text{ continuous} \right\}$$
$$\mathbb{V}_2 = \left\{ f \,:\, [a,b] \mapsto \mathbb{R} \text{ such that } f \in C^k([a,b]) \right\}$$

with $a, b \in \mathbb{R}$ and $k \in \mathbb{N}$.

Given in fact two function $f, g \in \mathbb{V}$ that are member of the same functional space, we can see that each linear combination of the functions determine a function h that's still in the vectorial space:

$$h(x) = \alpha f(x) + \beta g(x) \in \mathbb{V} \qquad \forall \alpha, \beta \in \mathbb{R},\ f, g \in \mathbb{V}$$

As example let's consider the two continuous function

$$f(x) = x^2 \quad \text{and} \quad g(x) = \sin x$$

then we can clearly see that the function

$$h(x) = 2f(x) + \frac{1}{3} g(x) = 2x^2 + \frac{1}{3} \sin x$$

is still continuous. This in general means that the functional space is *closed* respect to the operations of function summation and multiplication by a scalar.

14.3.1 Inner Product

In linear algebra given two vector $\boldsymbol{v}_1, \boldsymbol{v}_2$ it exists the bilinear operator scalar product $\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle = \boldsymbol{v}_1 \cdot \boldsymbol{v}_2$ that satisfy the following rules:

$$\langle \alpha \boldsymbol{v} + \beta \boldsymbol{w}, \boldsymbol{z} \rangle = \langle \boldsymbol{z}, \alpha \boldsymbol{v} + \beta \boldsymbol{w} \rangle = \alpha \langle \boldsymbol{v}, \boldsymbol{z} \rangle + \beta \langle \boldsymbol{w}, \boldsymbol{z} \rangle$$

for all $\alpha, \beta \in \mathbb{R}$, $\boldsymbol{v}, \boldsymbol{w}, \boldsymbol{z} \in \mathbb{V}$ and

$$\langle \boldsymbol{v}, \boldsymbol{v} \rangle \geq 0 \quad \text{and} \quad \langle \boldsymbol{v}, \boldsymbol{v} \rangle = 0 \quad \Leftrightarrow \quad \boldsymbol{v} = \boldsymbol{0}$$

Also in the functional space $\mathbb{V}$ can exists definitions of product scalar such the one here presented:

$$\langle f, g \rangle = \int_0^1 f(x) g(x) \, \mathrm{d}x \tag{14.1}$$

> **Note**
>
> This is one possible inner product. It is naturally defined on $L^2([0,1])$, whose elements are equivalence classes of functions equal almost everywhere:
>
> $$\mathbb{V} = L^2([0,1]).$$

In this case we can prove that this definitions meets the requirements stated for the scalar product considering the linear properties of the integrals as follows:

$$\begin{aligned}\langle \alpha f(\cdot) + \beta g(\cdot), h\rangle &= \int_0^1 \big(\alpha f(x) + \beta g(x)\big) h(x)\,\mathrm{d}x \\ &= \alpha \int_0^1 f(x)h(x)\,\mathrm{d}x + \beta \int_0^1 g(x)h(x)\,\mathrm{d}x \\ &= \alpha\langle f, h\rangle + \beta\langle g, h\rangle\end{aligned}$$

and also

$$\langle f, f\rangle = \int_0^1 f(x)^2\,\mathrm{d}x \geq 0$$

Moreover, $\langle f, f\rangle = 0$ if and only if $f = 0$ almost everywhere.

14.3.2 Norm

In linear algebra it's also defined the norm of a vector as

$$\|\boldsymbol{v}\| := \sqrt{\boldsymbol{v}\cdot\boldsymbol{v}}$$

and this expression is used to compute, from a vector, a single positive value (and in particular $\|\boldsymbol{v}\| = 0$ if and only if $\boldsymbol{v} = \mathbf{0}$). This definition also holds for the functional space and considering the scalar product defined (as example) in Equation 14.1 we can see that one definition of norm for function can be the one

$$\|f\|_{L^2} = \sqrt{\langle f, f\rangle} = \left(\int_0^1 f(x)^2\,\mathrm{d}x\right)^{1/2}$$

In this case we can clearly see that the norm $\|f\| \geq 0$ is always positive defined (and is zero only in the case on which the function f is identically null).

▪ **Example 14.3 — L^p Spaces.** A *space of functions* can be thought of as a set of functions satisfying certain properties.

For example, one such space is the set of continuous functions

$$f : [a,b] \to \mathbb{R}, \quad \text{with } f \in C([a,b]).$$

More generally, we may consider the space of k-times continuously differentiable functions

$$f \in C^k([a,b]), \quad k \in \mathbb{N}.$$

Similarly, we can work with spaces of piecewise continuous functions.

A less trivial example is the space of *integrable* functions, namely

$$f : [a,b] \to \mathbb{R}, \quad \text{such that } \int_a^b |f(x)|\,dx < \infty.$$

This is precisely the space of functions that are absolutely integrable on $[a,b]$.

In general we define as $L^p([a,b])$ the space of p-integrable functions, so such that

$$f \in L^p([a,b]) \qquad \Longleftrightarrow \qquad \int_a^b |f(x)|^p \, \mathrm{d}x < \infty$$

▪

14.4 First Variation

Let $\mathcal{J} : \mathbb{V} \to \mathbb{R}$ be a functional defined on a function space $\mathbb{V}$.
To study conditions for $\mathcal{J}$ to attain a minimum, we introduce the notion of the **derivative of a functional**, called the **first variation** (or *directional derivative*).

Given a function $x \in \mathbb{V}$ and a perturbation $\eta \in \mathbb{V}$, we consider the **variation**

$$x_\alpha(\cdot) = x(\cdot) + \alpha\eta(\cdot), \qquad \alpha \in \mathbb{R}.$$

Defining the auxiliary function

$$g(\alpha) = \mathcal{J}\big(x(\cdot) + \alpha\eta(\cdot)\big).$$

then the **first variation** of $\mathcal{J}$ at x in the direction η is

$$\delta\mathcal{J}(x;\eta) = g'(0). \tag{14.2}$$

14.4.1 Equivalent Forms

From the definition of the derivative, we can write

$$\delta\mathcal{J}(x;\eta) = \lim_{\alpha\to 0} \frac{\mathcal{J}(x+\alpha\eta) - \mathcal{J}(x)}{\alpha}.$$

Equivalently, using the small-o notation,

$$\mathcal{J}(x+\alpha\eta) = \mathcal{J}(x) + \alpha\,\delta\mathcal{J}(x;\eta) + o(\alpha).$$

14.4.2 Necessary Condition

If $x^\star$ is an interior local minimizer and both positive and negative perturbations $x^\star + \alpha\eta$ are admissible for sufficiently small $|\alpha|$, then

$$\delta\mathcal{J}(x^\star;\eta) = 0 \qquad \text{for every admissible } \eta.$$

This is only a necessary condition: a stationary function may be a maximum or a saddle point.

14.4.3 Second Variation

When the second directional derivative exists, the second variation is

$$\delta^2\mathcal{J}(x;\eta) = \frac{\mathrm{d}^2}{\mathrm{d}\alpha^2}\mathcal{J}(x+\alpha\eta)\bigg|_{\alpha=0}.$$

At a local minimizer, $\delta^2\mathcal{J}(x^\star;\eta) \geq 0$ for every admissible direction. Strict positivity together with suitable coercivity gives a strict local minimum. For an integral functional

$$\mathcal{J}(x) = \int_a^b L(x, x', t)\,\mathrm{d}t,$$

the second variation along a stationary function is

$$\delta^2 \mathcal{J}(x;\eta) = \int_a^b \left(L_{xx}\eta^2 + 2L_{xx'}\eta\eta' + L_{x'x'}(\eta')^2 \right) \mathrm{d}t.$$

Consequently, the strengthened Legendre condition $L_{x'x'} > 0$ is an important local convexity test. If L is jointly convex in (x, x'), every stationary admissible function is a global minimizer; strict convexity gives uniqueness.

▪ **Example 14.4 — First-Variation Example.** Given the functional

$$\mathcal{F}(x) = \int_0^1 \left(x^2(t) + 1 \right) \mathrm{d}t \; + x(1)$$

its first variation with respect to a generic direction $d(t)$ can be calculated by first defining the scalar function $g : \mathbb{R} \to \mathbb{R}$ as

$$\begin{aligned} g(\alpha) &= \mathcal{F}\big(x(\cdot) + \alpha\, d(\cdot)\big) \\ &= \int_0^1 \big(x(t) + \alpha\, d(t)\big)^2 + 1 \,\mathrm{d}t + x(1) + \alpha\, d(1) \end{aligned}$$

As reported in Equation 14.2, the first variation of the functional $\mathcal{F}(x)$ can be regarded as the derivative of $g(\alpha)$ respect to α evaluated for $\alpha = 0$; the first step is so determining

$$\begin{aligned} g'(\alpha) &= \frac{\mathrm{d}}{\mathrm{d}\alpha} g(\alpha) \\ &= \frac{\mathrm{d}}{\mathrm{d}\alpha} \int_0^1 \big(x(t) + \alpha\, d(t)\big)^2 + 1 \,\mathrm{d}t \\ &+ \frac{\mathrm{d}}{\mathrm{d}\alpha} \big(x(1) + \alpha\, d(1)\big) \end{aligned}$$

Assuming that $\frac{\mathrm{d}}{\mathrm{d}\alpha} \int = \int \frac{\mathrm{d}}{\mathrm{d}\alpha}$ (operation that cannot always be performed) then the derivative of $g(\alpha)$ can be regarded as

$$\begin{aligned} g'(\alpha) &= \int_0^1 \frac{\mathrm{d}}{\mathrm{d}\alpha} \Big(x(t) + \alpha\, d(t) \Big)^2 \mathrm{d}t + \frac{\mathrm{d}}{\mathrm{d}\alpha} \Big(x(1) + \alpha\, d(1) \Big) \\ &= \int_0^1 2\big(x(t) + \alpha\, d(t)\big) d(t) \,\mathrm{d}t + d(1) \end{aligned}$$

Evaluating this expression for $\alpha = 0$ then the first variation of $\mathcal{F}(x)$ with direction $d(t)$ becomes

$$\delta\mathcal{F}(x;d) = g'(0) = \int_0^1 2xd \,\mathrm{d}t + d(1)$$

▪

14.5 Integral Differentiation

In the computation of variations it is often assumed that

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} \circ \int_a^b \quad = \quad \int_a^b \circ \frac{\mathrm{d}}{\mathrm{d}\alpha}.$$

However, this property does **not** hold in general: some regularity conditions on the functions involved are required.
To better understand the issue, let us first examine a simpler problem.

14.5.1 Limit Interchange

Since a derivative is defined as a limit of a quotient, the question reduces to whether the operations of **limit** and **integration** can always be interchanged:

$$\lim \circ \int \quad \stackrel{?}{=} \quad \int \circ \lim .$$

Concretely, let $\{f_n\}_{n=1}^{\infty}$ be a sequence of functions such that

$$f(x) = \lim_{n\to\infty} f_n(x), \qquad \forall x \in \mathbb{R}.$$

We ask whether

$$\lim_{n\to\infty} \int_{-\infty}^{\infty} f_n(t)\,dt \stackrel{?}{=} \int_{-\infty}^{\infty} \lim_{n\to\infty} f_n(t)\,dt = \int_{-\infty}^{\infty} f(t)\,dt.$$

14.5.2 Counterexample

Consider the sequence

$$f_n(x) = \begin{cases} 0, & x \le 0, \\ nx, & 0 < x \le \frac{1}{n}, \\ 2 - nx, & \frac{1}{n} < x \le \frac{2}{n}, \\ 0, & x > \frac{2}{n}. \end{cases}$$

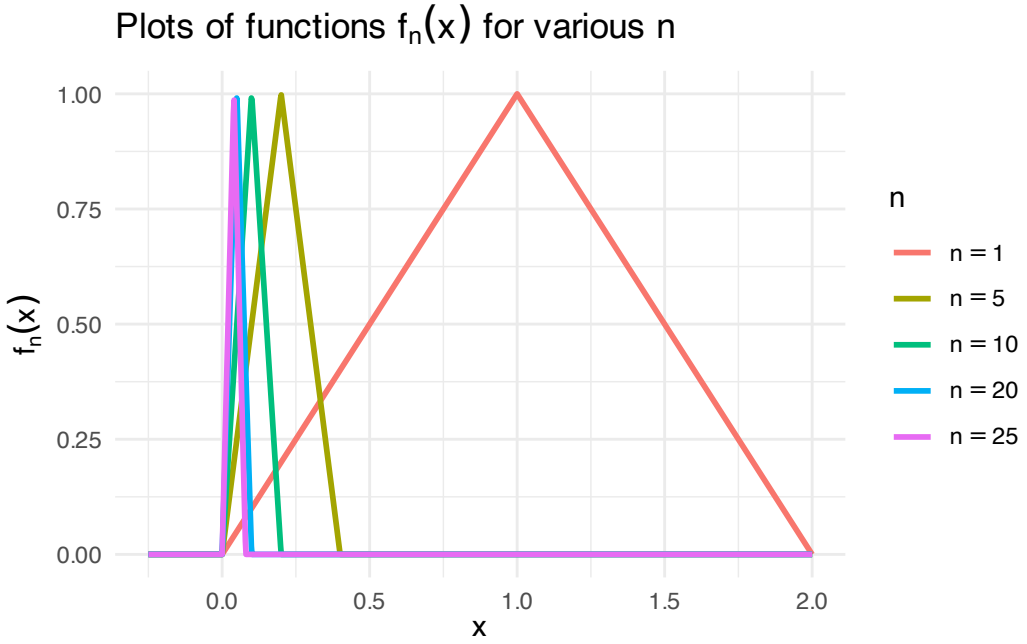


Figure 14.2: A sequence of triangular functions with shrinking support.

Each f_n is supported in the interval $(0, 2/n)$ and takes the shape of a triangular "bump" of base $2/n$ and height 1.

For every fixed $x \in \mathbb{R}$ we have

$$\lim_{n\to\infty} f_n(x) = 0,$$

so the pointwise limit is the zero function:

$$f(x) \equiv 0.$$

However,

$$\int_{-\infty}^{\infty} f_n(t)\,dt = \text{area of the triangle } = \frac{1}{2} \cdot \frac{2}{n} \cdot 1 = \frac{1}{n}.$$

or in more detail

$$\begin{aligned}
\int_{-\infty}^{\infty} f_n(x)\,\mathrm{d}x &= \int_0^{1/n} nx\,\mathrm{d}x + \int_{1/n}^{2/n} 2 - nx\,\mathrm{d}x \\
&= \left[n\frac{x^2}{2}\right]_{x=0}^{x=1/n} + \left[2x - n\frac{x^2}{2}\right]_{x=1/n}^{x=2/n} \\
&= \left[n\frac{(1/n)^2}{2} - n\frac{(0)^2}{2}\right] + \left[2\frac{2}{n} - n\frac{(2/n)^2}{2} - 2\frac{1}{n} + n\frac{(1/n)^2}{2}\right] \\
&= \frac{1}{2n} + \frac{4}{n} - \frac{2}{n} - \frac{2}{n} + \frac{1}{2n} = \frac{1}{n}
\end{aligned}$$

Thus

$$\lim_{n\to\infty} \int_{-\infty}^{\infty} f_n(t)\,dt = 0, \qquad \int_{-\infty}^{\infty} f(t)\,dt = 0,$$

so in this case the interchange works.

Consider now the functions

$$g_n(x) = n f_n(x) = \begin{cases} 0 & x \le 0 \\ n^2 x & 0 < x \le \frac{1}{n} \\ 2n - n^2 x & \frac{1}{n} < x \le \frac{2}{n} \\ 0 & x > \frac{2}{n} \end{cases}$$

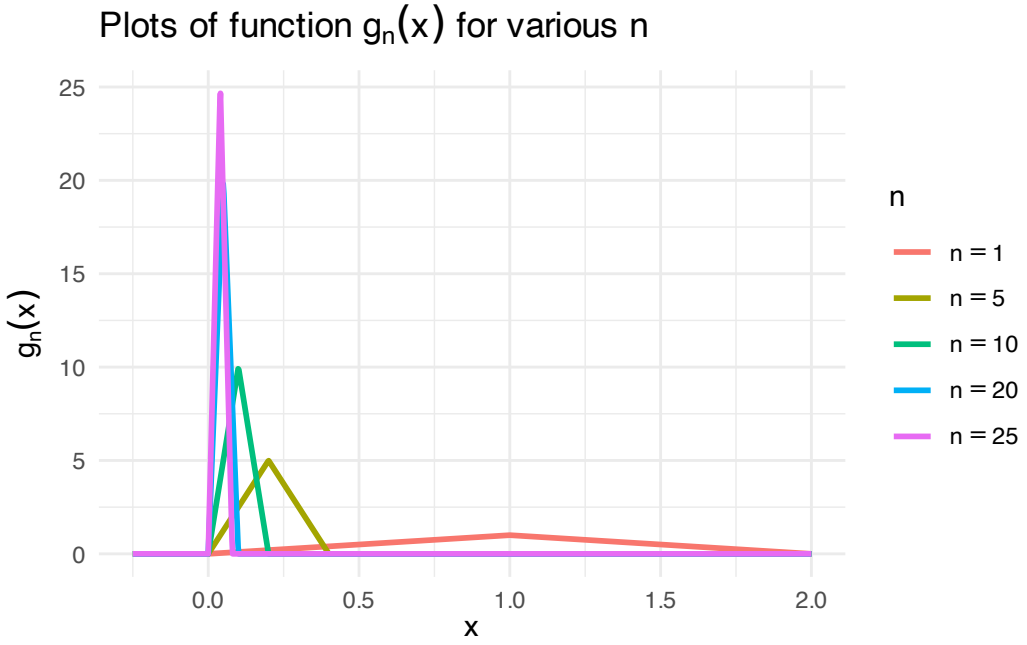


Figure 14.3: Derivatives of the triangular sequence as the support shrinks.

it is clear that

$$\lim_{n\to\infty} g_n(x) = 0, \qquad \forall x \in \mathbb{R}$$

and

$$\begin{aligned}\int_{-\infty}^{\infty} g_n(x)\,\mathrm{d}x &= \int_0^{1/n} n^2 x\,\mathrm{d}x + \int_{1/n}^{2/n} 2n - n^2 x\,\mathrm{d}x \\ &= \frac{1}{2} + \frac{1}{2} = 1\end{aligned}$$

and thus

$$\begin{aligned}\lim_{n\to\infty} \int_{-\infty}^{\infty} g_n(x)\mathrm{d}x &= \lim_{n\to\infty} 1 = 1 \\ \int_{-\infty}^{\infty} \lim_{n\to\infty} g_n(x)\mathrm{d}x &= \int_{-\infty}^{\infty} 0\mathrm{d}x = 0\end{aligned}$$

and in this case limit and integral **cannot** be exchanged.

Thus limits, derivatives, and integrals cannot be interchanged without additional hypotheses.

14.5.3 A Sufficient Condition

A useful version of Leibniz's rule is the following. Suppose $F(t, \alpha)$ and $\partial F/\partial\alpha$ are continuous on $[a, b] \times (-\varepsilon, \varepsilon)$. Then

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} \int_a^b F(t, \alpha)\,\mathrm{d}t = \int_a^b \frac{\partial F}{\partial \alpha}(t, \alpha)\,\mathrm{d}t.$$

More generally, the same conclusion follows from a dominated-convergence argument when $|\partial F/\partial\alpha|$ is bounded by an integrable function independent of α.

14.6 Local Minima and Existence

The word *minimum* must be qualified by the topology used on the admissible space. A function $x^\star$ is a **weak local minimizer** if $\mathcal{F}(x^\star) \le \mathcal{F}(x)$ for all admissible x sufficiently close to $x^\star$ in a norm controlling x and its derivatives (for example a C^1 norm). A **strong local minimizer** is defined using a weaker distance, typically the C^0 norm. Every global minimizer is local, but a stationary function need not be a local minimizer.

Existence is a separate question from stationarity. In modern variational analysis the direct method typically proceeds by showing:

1. the admissible set is nonempty and sequentially closed in a suitable topology;
2. the functional is coercive, so minimizing sequences remain bounded;
3. a bounded minimizing sequence has a convergent subsequence (compactness or weak compactness);
4. the functional is lower semicontinuous, so the limit attains the infimum.

For integral functionals on Sobolev spaces, convexity of the integrand with respect to the derivative variable is a standard sufficient hypothesis for weak lower semicontinuity. These hypotheses are not needed to derive Euler–Lagrange equations, but they are essential when one wants to prove that a minimizer actually exists.

14.7 Sobolev Spaces

Classical derivations often assume $x \in C^1$ or C^2. For existence theory the natural spaces are frequently Sobolev spaces. For example,

$$W^{1,p}(a,b) = \{x \in L^p(a,b) \,:\, x' \in L^p(a,b)\},$$

where x' is the weak derivative. Boundary values are interpreted through the trace operator. This setting allows admissible minimizers whose derivatives need not be classically continuous and separates the regularity required for existence from the additional regularity needed to write the Euler–Lagrange equation pointwise.

15. Fundamental Lemma

Consider the unconstrained minimization of a differentiable function

$$f : A \subseteq \mathbb{R}^n \to \mathbb{R}.$$

If $\boldsymbol{x}^\star$ is an interior local minimizer, the first-order necessary condition is

$$\nabla f(\boldsymbol{x}^\star) = \mathbf{0}.$$

The analogous statement in the calculus of variations is that, at a local minimizer x, the first variation must vanish in every admissible direction:

$$\delta \mathcal{J}(x; \eta) = 0.$$

An **admissible perturbation** η is a function such that $x + \alpha\eta$ remains in the admissible set for all sufficiently small $|\alpha|$. The fundamental lemma converts an integral identity valid for every admissible test function into a pointwise identity.

15.1 Statement

Lemma 15.1 — Lemma. Let

$$f : [a, b] \to \mathbb{R}, \qquad f \in C([a, b]).$$

If

$$\int_a^b f(x) g(x)\, \mathrm{d}x = 0 \tag{15.1}$$

for every test function $g \in C_c^\infty((a, b))$, then

$$f(x) = 0 \qquad \forall x \in [a, b].$$

Here $C_c^\infty((a, b))$ denotes the set of infinitely differentiable functions whose support is a compact subset of (a, b).

15.2 Sign Persistence

The proof relies on a simple consequence of continuity.

Lemma 15.2 — Lemma. Let $f : [a, b] \to \mathbb{R}$ be continuous and suppose that $f(c) > 0$ for some $c \in (a, b)$. Then there exists $\delta > 0$, with

$$[c - \delta, c + \delta] \subset (a, b),$$

such that

$$f(x) \geq \frac{f(c)}{2} > 0 \qquad \forall x \in [c - \delta, c + \delta].$$

Proof. Choose

$$\varepsilon = \frac{f(c)}{2} > 0.$$

Since f is continuous at c, there exists $\delta > 0$ such that

$$|x - c| < \delta \quad \Longrightarrow \quad |f(x) - f(c)| < \frac{f(c)}{2}.$$

Therefore

$$f(x) > f(c) - \frac{f(c)}{2} = \frac{f(c)}{2}.$$

Reducing δ if necessary ensures that $[c - \delta, c + \delta] \subset (a, b)$. □ ■

15.3 Proof Strategy: A Triangular Test Function

Before giving the fully rigorous proof, it is useful to see the central idea with a test function for which every computation is explicit.

Assume, seeking a contradiction, that f is not identically zero. If necessary replace f by $-f$, so that for some $c \in (a, b)$,

$$f(c) > 0.$$

By Lemma 15.2, choose $\delta > 0$ such that

$$f(x) \geq \frac{f(c)}{2} > 0 \qquad \forall x \in [c - \delta, c + \delta].$$

Now consider the triangular function

$$g_{\triangle}(x) = \begin{cases} \frac{x-(c-\delta)}{\delta}, & c - \delta \leq x \leq c, \\ \frac{c+\delta-x}{\delta}, & c \leq x \leq c + \delta, \\ 0, & \text{otherwise.} \end{cases} \tag{15.2}$$

It satisfies

$$g_{\triangle}(x) \geq 0, \qquad \operatorname{supp}(g_{\triangle}) = [c - \delta, c + \delta], \qquad g_{\triangle}(c) = 1.$$

Moreover, its integral is simply the area of a triangle of base 2δ and height 1:

$$\int_a^b g_{\triangle}(x)\,dx = \frac{1}{2}(2\delta)(1) = \delta.$$

Therefore

$$\begin{aligned}\int_a^b f(x)g_\triangle(x)\,\mathrm{d}x &= \int_{c-\delta}^{c+\delta} f(x)g_\triangle(x)\,\mathrm{d}x \\ &\geq \frac{f(c)}{2}\int_{c-\delta}^{c+\delta} g_\triangle(x)\,\mathrm{d}x \\ &= \frac{f(c)\delta}{2} > 0.\end{aligned} \tag{15.3}$$

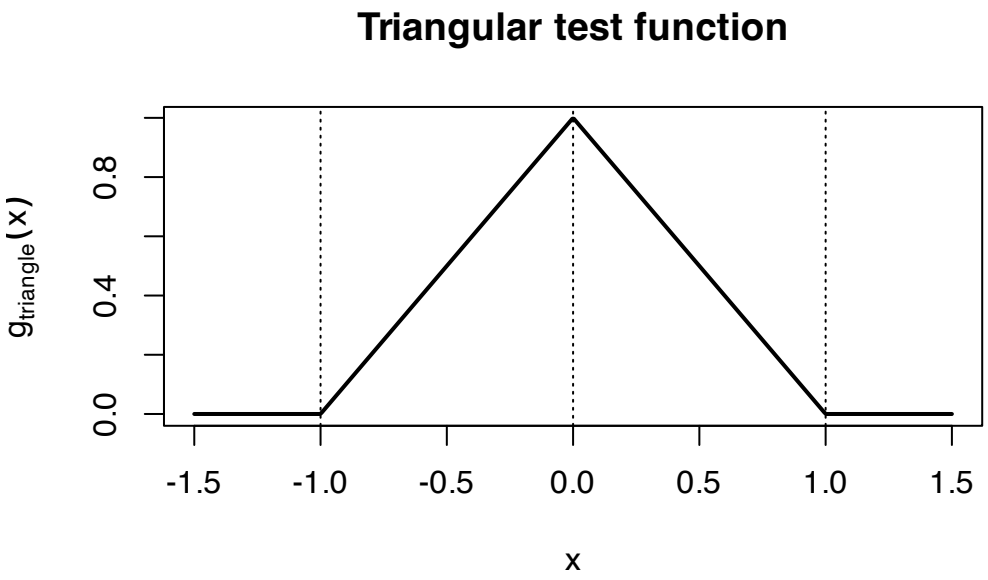


Figure 15.1: Localized triangular test function used in the fundamental lemma.

The triangular test function $g_\triangle$ makes the sign argument and the integral estimate completely explicit.

15.3.1 Why This Is Not Yet a Proof

The computation above contains the essential logic of the fundamental lemma, but it does **not** yet prove Lemma 15.1 as stated.

Indeed,

$$g_\triangle \in C_c^0((a,b)),$$

but $g_\triangle$ is not differentiable at

$$x = c-\delta, \qquad x = c, \qquad x = c+\delta,$$

and hence

$$g_\triangle \notin C_c^\infty((a,b)).$$

The hypothesis in Equation 15.1 is only known for smooth compactly supported test functions. Consequently, inserting $g_\triangle$ directly into Equation 15.1 would be a logical gap.

The important point is that **nothing is wrong with the sign argument**. Only the regularity of the chosen test function must be repaired.

15.4 Smooth Bump Function

A standard smooth building block is

$$h(s) = \begin{cases} \exp(-1/s), & s > 0, \\ 0, & s \le 0. \end{cases} \tag{15.4}$$

This function belongs to $C^\infty(\mathbb{R})$, and all of its derivatives vanish at $s = 0$.
A compactly supported bump centered at c is obtained more directly as

$$g_\infty(x) = \begin{cases} \exp\left(-\dfrac{1}{\delta^2 - (x-c)^2}\right), & |x-c| < \delta, \\ 0, & |x-c| \ge \delta. \end{cases} \tag{15.5}$$

It has exactly the properties required for the proof:

$$g_\infty \in C_c^\infty((a,b)),$$

$$g_\infty(x) \ge 0,$$

and

$$\operatorname{supp}(g_\infty) = [c-\delta, c+\delta].$$

Furthermore,

$$g_\infty(x) > 0 \qquad \text{for } |x-c| < \delta,$$

so that

$$\int_{c-\delta}^{c+\delta} g_\infty(x)\,dx > 0.$$

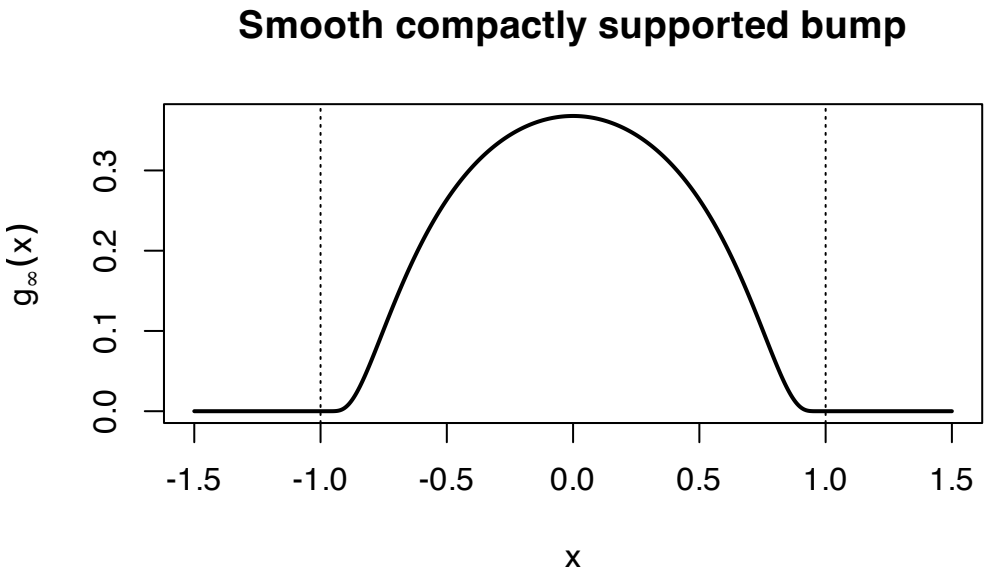


Figure 15.2: Smooth localized bump function used as a test variation.

A smooth test function with the same localization and sign properties used in the triangular argument.

15.5 Rigorous Proof

Proof. Assume that f is not identically zero. Then there exists $c \in (a,b)$ such that $f(c) \neq 0$. Replacing f by $-f$ if necessary, suppose that

$$f(c) > 0.$$

By Lemma 15.2, there exists $\delta > 0$ such that

$$[c - \delta, c + \delta] \subset (a, b)$$

and

$$f(x) \geq \frac{f(c)}{2} > 0 \qquad \forall x \in [c - \delta, c + \delta].$$

Choose the smooth bump g_∞ defined in Equation 15.5. Since

$$g_\infty \in C_c^\infty((a, b)),$$

it is an admissible test function in Equation 15.1. Hence the hypothesis of the fundamental lemma would give

$$\int_a^b f(x) g_\infty(x)\, \mathrm{d}x = 0.$$

On the other hand, because $g_\infty \geq 0$ and is supported in $[c - \delta, c + \delta]$,

$$\begin{aligned} \int_a^b f(x) g_\infty(x)\, \mathrm{d}x &= \int_{c-\delta}^{c+\delta} f(x) g_\infty(x)\, \mathrm{d}x \\ &\geq \frac{f(c)}{2} \int_{c-\delta}^{c+\delta} g_\infty(x)\, \mathrm{d}x \\ &> 0, \end{aligned}$$

which is a contradiction. Therefore

$$f(x) = 0 \qquad \forall x \in (a, b).$$

Since f is continuous on $[a, b]$, the same conclusion extends to the endpoints:

$$f(a) = f(b) = 0.$$

Thus

$$f \equiv 0 \quad \text{on } [a, b].$$

□ ■

15.6 Interpretation

The triangular and smooth proofs use exactly the same mechanism:

1. if $f(c) \neq 0$, continuity forces f to retain the same sign in a neighborhood of c;
2. choose a nonnegative test function supported entirely in that neighborhood;
3. the product fg then has one strict sign on a set of positive measure;
4. consequently $\int fg$ cannot vanish.

The triangular function makes Steps 2–4 transparent because its area is elementary. The smooth bump then reproduces the same argument inside the actual test space $C_c^\infty((a, b))$ and turns the intuitive argument into a rigorous proof.

16. Euler–Lagrange Equations

The **Euler–Lagrange equation** expresses the stationary-action principle used in mechanics.

16.1 Pendulum Example

Consider a pendulum of mass m connected to a fixed pivot by a rod of length l. The same configuration may be represented by redundant Cartesian coordinates (x, y) or by polar coordinates (r, θ).

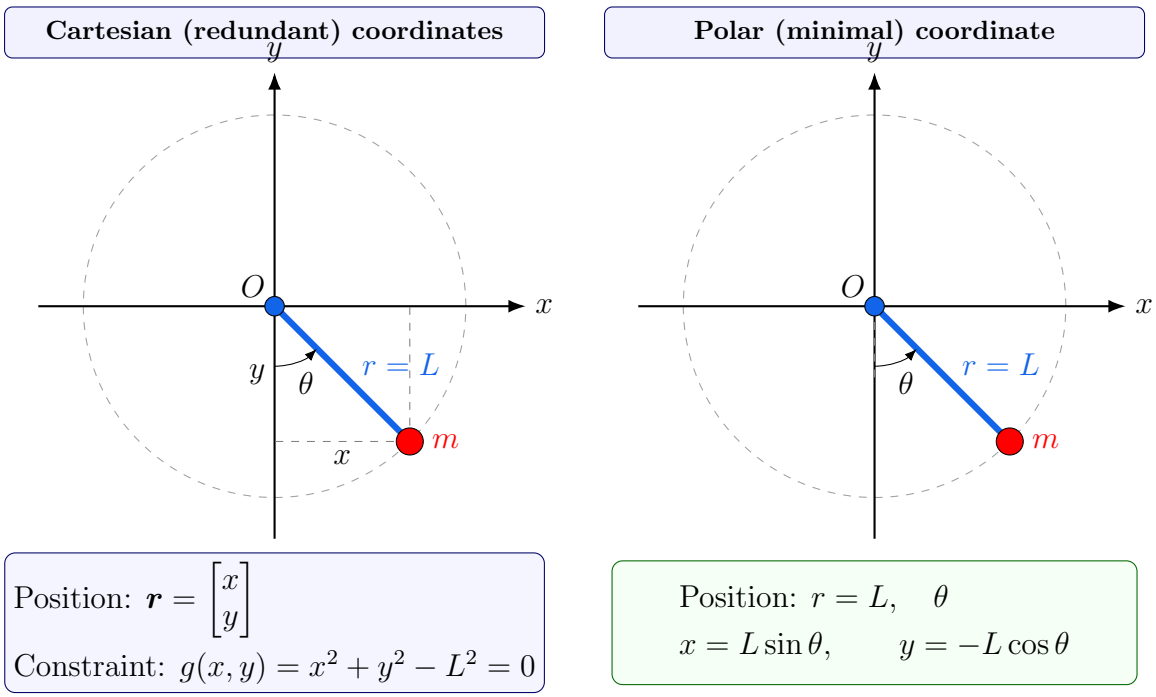


Figure 16.1: Pendulum coordinates in Cartesian and polar representations.

The coordinate transformation and its time derivative are

$$x = r\sin\theta, \qquad y = -r\cos\theta,$$

$$\dot{x} = \dot{r}\sin\theta + r\cos\theta\,\dot{\theta}, \qquad \dot{y} = -\dot{r}\cos\theta + r\sin\theta\,\dot{\theta}.$$

Consequently, the kinetic and potential energies are

$$T = \frac{m}{2}\left(\dot{r}^2 + r^2\dot{\theta}^2\right), \qquad V = -mgr\cos\theta.$$

The fixed-length constraint $r = l$ eliminates the radial coordinate and gives

$$T(\theta,\dot{\theta}) = \frac{m}{2}l^2\dot{\theta}^2, \qquad V(\theta) = -mgl\cos\theta.$$

To solve the dynamic equation $\theta(t)$ of the mechanism we can compute the lagrangian L of the system defined as

$$L(\theta,\dot{\theta}) = T(\theta,\dot{\theta}) - V(\theta) = \frac{m}{2}l^2\dot{\theta}^2 + lmg\cos\theta$$

The physical trajectory makes the **action** $\mathcal{A}$ stationary:

$$\mathcal{A}(\theta) = \int_{t_0}^{t_1} L(\theta,\dot{\theta})\,\mathrm{d}t \tag{16.1}$$

where the values $\theta(t_0) = \theta_0$ and $\theta(t_1) = \theta_1$ are known parameters.

In practice to determine the required solution we can use analytical tool to find the trajectory that can then be demonstrated to be the minimum of the action $\mathcal{A}$ (that's indeed a functional).

However we can also try to analytically determine the function $\theta^\star(t)$ that minimize the functional $\mathcal{A}(\cdot)$ by computing the directional derivatives.

Let now consider the function $\theta^\star(t)$ and a direction $\delta\theta(t)$ that satisfy

$$\delta\theta(t_0) = \delta\theta(t_1) = 0$$

we can then use the fundamental lemma of the calculus of variation to determine the minimal function (considering that the expression Equation 16.1 of the action is comparable to Equation 15.1 of the lemma).

To determine a stationary trajectory $\theta^\star$, the first variation of the action must vanish in every admissible direction $\delta\theta$:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\alpha}\mathcal{A}\left(\theta^\star + \alpha\,\delta\theta\right) &= \frac{\mathrm{d}}{\mathrm{d}\alpha}\int_{t_0}^{t_1} L\left(\theta^\star + \alpha\,\delta\theta, \dot{\theta}^\star + \alpha\,\delta\dot{\theta}\right)\mathrm{d}t \\
&= \int_{t_0}^{t_1}\left[\frac{\partial L}{\partial\theta}\frac{\mathrm{d}}{\mathrm{d}\alpha}\left(\theta^\star + \alpha\,\delta\theta\right) + \frac{\partial L}{\partial\dot{\theta}}\frac{\mathrm{d}}{\mathrm{d}\alpha}\left(\dot{\theta}^\star + \alpha\,\dot{\delta\theta}\right)\right]\mathrm{d}t \\
&= \int_{t_0}^{t_1}\left[\left(-lmg\sin(\theta^\star + \alpha\,\delta\theta)\right)\delta\theta + ml^2\left(\dot{\theta}^\star + \alpha\,\delta\dot{\theta}\right)\delta\dot{\theta}\right]\mathrm{d}t
\end{aligned}$$

Note that from in the first step we made the implicit assumption that

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}\int = \int\frac{\mathrm{d}}{\mathrm{d}\alpha}$$

while however this is not always possible. Evaluating the previous expression for $\alpha = 0$ gives the first variation of the action $\mathcal{A}$ that's

$$\begin{aligned}
\delta\mathcal{A}(\theta^\star;\delta\theta) &= \frac{\mathrm{d}}{\mathrm{d}\alpha}\mathcal{A}\left(\theta^\star + \alpha\,\delta\theta\right)\Big|_{\alpha=0} \\
&= \int_{t_0}^{t_1}\left(-lmg\sin\theta^\star\delta\theta + ml^2\dot{\theta}^\star\delta\dot{\theta}\right)\mathrm{d}t = 0
\end{aligned}$$

for all $\delta\theta$. Performing an integration by part allows to remove the term $\delta\dot{\theta}$ that's hard to determine, and in fact

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\,\delta\theta\right) = \frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\right)\delta\theta + ml^2\dot{\theta}^\star\delta\dot{\theta}$$

$$\Downarrow$$

$$ml^2\dot{\theta}^\star\delta\dot{\theta} = \frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\,\delta\theta\right) - \frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\right)\delta\theta$$

Performing the substitution of the integration by parts determines the result

$$\begin{aligned}\delta\mathcal{A}\left(\theta^\star;\delta\theta\right) &= \int_{t_0}^{t_1}\left[-lmg\sin\theta^\star - \frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\right)\right]\delta\theta\,\mathrm{d}t + \cancel{\left[ml^2\dot{\theta}^\star\delta\theta\right]_{t_0}^{t_1}} \\ &= \int_{t_0}^{t_1}\underbrace{\left[-lmg\sin\theta^\star - \frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\right)\right]}_{f(t)}\delta\theta\,\mathrm{d}t = 0\end{aligned}$$

We can now see in this formulation that the marked expression represent the function $f(x)$ of the fundamental lemma considering that the relation must be true for all approaching direction $\delta\theta$, and so it must be

$$-lmg\sin\theta^\star - \frac{\mathrm{d}}{\mathrm{d}t}\left(ml^2\dot{\theta}^\star\right) = 0$$

The problem now to complete the analyses of the pendulum motion is determining the function $\theta^\star(t)$ that satisfy this expression and also match the boundary conditions $\theta(t_0) = \theta_0$ and $\theta(t_1) = \theta_1$.

Equivalently, the stationary path satisfies the pendulum equation

$$ml^2\ddot{\theta} + mgl\sin\theta = 0, \qquad \text{or} \qquad \ddot{\theta} + \frac{g}{l}\sin\theta = 0.$$

The derivation above uses the minimal coordinate θ, which automatically enforces the fixed-length constraint. The same motion can be written with the redundant Cartesian coordinates $\boldsymbol{q} = (x, y)^\mathsf{T}$ and the holonomic constraint $\phi(\boldsymbol{q}) = x^2 + y^2 - l^2 = 0$.

In Cartesian coordinates, the Lagrangian is

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{m}{2}\dot{\boldsymbol{q}}^\mathsf{T}\dot{\boldsymbol{q}} - mgy,$$

and the constraint force is represented by a multiplier λ. Applying the constrained Euler–Lagrange equations gives

$$\begin{aligned} m\ddot{\boldsymbol{q}} + mg\boldsymbol{e}_2 = \nabla\phi(\boldsymbol{q})\lambda = 2\boldsymbol{q}\lambda, \\ \phi(\boldsymbol{q}) = x^2 + y^2 - l^2 = 0, \end{aligned} \qquad \boldsymbol{e}_2 = (0, 1)^\mathsf{T}.$$

Equivalently,

$$\begin{aligned} m\ddot{x} &= 2x\lambda, \\ m\ddot{y} &= 2y\lambda - mg, \\ x^2 + y^2 &= l^2. \end{aligned}$$

This redundant-coordinate formulation is a differential–algebraic equation: the two differential equations are coupled to the algebraic length constraint. Its derivation uses the same variational principle as the reduced equation above, with the constraint included through the Lagrange multiplier.

16.2 General Formulation

Given a generic functional of the form

$$\mathcal{A}(x) = \int_a^b L\big(x(t), x'(t), t\big)\,\mathrm{d}t$$

the problem is to minimize the functional $\mathcal{A}(x)$ for a function $x \in \mathbb{V}$ subject to the boundary conditions $x(a) = x_a$ and $x(b) = x_b$ in the function space $\mathbb{V}$ defined as

$$\mathbb{V} = \big\{ x \mid x \in C^2([a,b]) \text{ with } x(a) = x_a, x(b) = x_b \big\}$$

Let $\mathbb{D}$ the function space of all the feasible directions of derivatives defined as

$$\mathbb{D} = \big\{ \delta x \mid \delta x \in C^\infty([a,b]) \text{ with } \delta x(a) = \delta x(b) = 0 \big\}$$

the function $x(t)$ that minimize the functional is the one whose first variation is zero for any admissible perturbation, and so such that

$$\delta\mathcal{A}(x;\delta x) = \frac{\mathrm{d}}{\mathrm{d}\alpha}\mathcal{A}(x + \alpha\,\delta x)\Big|_{\alpha=0} = 0 \qquad \text{for all } \delta x \in \mathbb{D}$$

Assuming the possibility to correctly apply the rule

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}\int = \int \frac{\mathrm{d}}{\mathrm{d}\alpha}$$

we can express the directional derivative of the functional $\mathcal{A}(x)$ evaluated in $x + \alpha\,\delta x$ as

$$\begin{aligned}
\frac{\mathrm{d}\mathcal{A}(x+\alpha\delta x)}{\mathrm{d}\alpha} &= \frac{\mathrm{d}}{\mathrm{d}\alpha}\int_a^b L\big(x + \alpha\,\delta x, x' + \alpha\,\delta x', t\big)\,\mathrm{d}t \\
&= \int_a^b \left(\frac{\partial L}{\partial x}\delta x + \frac{\partial L}{\partial x'}\delta x'\right)\mathrm{d}t \\
\frac{\mathrm{d}\mathcal{A}}{\mathrm{d}\alpha}\Big|_{\alpha=0} &= \int_a^b \left(\frac{\partial L}{\partial x}(x,x',t)\delta x + \frac{\partial L}{\partial x'}(x,x',t)\delta x'\right)\mathrm{d}t
\end{aligned}$$

By performing the integration by part it's possible do reconvert the term associated do $\delta x'$ into pieces depending on δx one of which, when evaluated, becomes zero due to the fact that $\delta x(a) = \delta x(b) = 0$:

$$\begin{aligned}
\frac{\mathrm{d}\mathcal{A}}{\mathrm{d}\alpha}\Big|_{\alpha=0} &= \int_a^b \left[\frac{\partial L}{\partial x}(x,x',t) - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\right)\right]\delta x\,\mathrm{d}t \\
&\quad + \int_a^b \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\delta x\right)\mathrm{d}t \\
&= \int_a^b \left[\frac{\partial L}{\partial x}(x,x',t) - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\right)\right]\delta x\,\mathrm{d}t \\
&\quad + \cancel{\left[\frac{\partial L}{\partial x'}(x,x',t)\delta x\right]_{t=a}^{t=b}}
\end{aligned}$$

> **Note**
>
> In this case the substitution by part has been obtained by performing the operation
>
> $$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\delta x\right) = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\right)\delta x + \frac{\partial L}{\partial x'}(x,x',t)\delta x'$$
>
> If we in fact explicit the term containing $\delta x'$ we have
>
> $$\frac{\partial L}{\partial x'}(x,x',t)\delta x' = \underbrace{\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\delta x\right)}_{\text{term \#1}} - \underbrace{\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\right)\delta x}_{\text{term \#2}}$$

The second term still remains in the variation of the functional $\mathcal{A}$, while the first term is eliminated because it's evaluation becomes

$$\int_a^b \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}\delta x\right)\mathrm{d}t = \left[\frac{\partial L}{\partial x'}(x(t),x'(t),t)\delta x(t)\right]_a^b$$
$$= \frac{\partial L}{\partial x'}(x(b),x'(b),b)\delta x(b) - \frac{\partial L}{\partial x'}(x(a),x'(a),a)\delta x(a)$$

From the definition of $\mathbb{D}$, $\delta x(a) = \delta x(b) = 0$, so the boundary term vanishes.

We can now see that the necessary stationarity condition for the functional $\mathcal{A}$, by using the fundamental lemma of calculus of variation, is requiring that the function $f(t)$ defined as follows is identically null:

$$\left.\frac{\mathrm{d}\mathcal{A}(x+\alpha\,\delta x)}{\mathrm{d}\alpha}\right|_{\alpha=0} = \int_a^b \underbrace{\left(\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}\right)}_{=f(t)}\delta x\,\mathrm{d}t = 0 \tag{16.2}$$

Thus every smooth stationary function must solve the following second-order boundary value problem:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'} - \frac{\partial L}{\partial x} = 0 \\ x(a) = x_a \\ x(b) = x_b \end{cases}$$

For a vector of generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$, the same stationarity condition is the Euler–Lagrange system

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{\boldsymbol{q}}}\right)^{\mathsf{T}} - \left(\frac{\partial L}{\partial \boldsymbol{q}}\right)^{\mathsf{T}} = \boldsymbol{0}. \tag{16.3}$$

16.3 First Integrals and Vectors

If L does not depend explicitly on t, the Euler–Lagrange equation implies the **Beltrami identity**

$$L - x'\frac{\partial L}{\partial x'} = C.$$

If L does not depend on x, then $\partial L/\partial x'$ is constant. These first integrals often reduce a second-order Euler–Lagrange equation to a first-order equation.

For a vector-valued curve $\boldsymbol{x} : [a,b] \to \mathbb{R}^n$ and $L(\boldsymbol{x}, \dot{\boldsymbol{x}}, t)$, the equations hold componentwise:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{x}_i} - \frac{\partial L}{\partial x_i} = 0, \qquad i = 1, \dots, n.$$

▪ **Example 16.1 — First-Variation Example.** Given the generic functional defined as

$$\mathcal{F}(x) = \int_a^b G(x, x', t)\,\mathrm{d}t + x(a)x(b)$$

to compute its first variation, we use the standard approach and perturb the function x:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\alpha}\mathcal{F}\big(x+\alpha\,\delta x\big) &= \frac{\mathrm{d}}{\mathrm{d}\alpha}\int_a^b G\big(x+\alpha\,\delta x, x'+\alpha\,\delta x', t\big)\,\mathrm{d}t \\
&+ \frac{\mathrm{d}}{\mathrm{d}\alpha}\Big(\big(x(a)+\alpha\delta x(a)\big)\big(x(b)+\alpha\,\delta x(b)\big)\Big) \\
&= \int_a^b \left(\frac{\partial G}{\partial x}\,\delta x + \frac{\partial G}{\partial x'}\delta x'\right)\mathrm{d}t \\
&+ \delta x(a)\big(x(b)+\alpha\,\delta x(b)\big) + \big(x(a)+\alpha\,\delta x(a)\big)\delta x(b) \\
&\downarrow \quad \alpha = 0 \\
&= \int_a^b \left(\frac{\partial G}{\partial x}\delta x + \frac{\partial G}{\partial x'}\delta x'\right)\mathrm{d}t + \delta x(a)x(b) + x(a)\delta x(b)
\end{aligned}$$

This redundant formulation can be simplified by using the Gateaux derivative notation δ that allows to express the first variation as

$$\begin{aligned}
\delta\mathcal{F}(x) &= \delta\left(\int_a^b G(x,x',t)\,\mathrm{d}t + x(a)x(b)\right) \\
&= \int_a^b \delta G(x,x',t)\,\mathrm{d}t + \delta\big(x(a)x(b)\big) \\
&= \int_a^b \left(\frac{\partial G}{\partial x}(x,x',t)\delta x + \frac{\partial G}{\partial x'}(x,x',t)\delta x'\right)\mathrm{d}t \\
&+ \delta x(a)x(b) + x(a)\delta x(b)
\end{aligned}$$

using integration by part

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G}{\partial x'}(x,x',t)\delta x\right) = \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial G}{\partial x'}(x,x',t)\delta x + \frac{\partial G}{\partial x'}(x,x',t)\delta x'$$

and substituting

$$\begin{aligned}
\delta\mathcal{F}(x) = \int_a^b \underbrace{\left(\frac{\partial G}{\partial x}(x,x',t) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial G}{\partial x'}(x,x',t)\right)}_{(A)}\delta x\,\mathrm{d}t \\
+ \underbrace{\left[x(a) + \frac{\partial G}{\partial x'}(x(b),x'(b),b)\right]}_{(B)}\delta x(b) \\
+ \underbrace{\left[x(b) - \frac{\partial G}{\partial x'}(x(a),x'(a),a)\right]}_{(C)}\delta x(a)
\end{aligned}$$

using fundamental lemma of calculus of variations $(A) = 0$ and then using variation differents from zero on the border it follows that $(B) = (C) = 0$ and the stationary points of the functional $F(x)$ satisfy the BVP

$$\begin{cases} \dfrac{\partial G}{\partial x}(x, x', t) - \dfrac{\mathrm{d}}{\mathrm{d}t}\dfrac{\partial G}{\partial x'}(x, x', t) = 0 \\ x(a) + \dfrac{\partial G}{\partial x'}(x(b), x'(b), b) = 0 \\ x(b) - \dfrac{\partial G}{\partial x'}(x(a), x'(a), a) = 0 \end{cases}$$

∎

16.4 Brachistochrone

As saw at the introduction, functional minimization can be used to solve real problem. As example the brachistochrone problem wants to determine the path that minimize the time T to reach point B from a point A for a mass m subjected only to the force of gravity: this means minimizing the function

$$T = \int_A^B \frac{1}{v}\,\mathrm{d}s$$

where v is the velocity of the point over the arc length $\mathrm{d}s$. In order to describe the problem we consider x as the horizontal axis and y the vertical one (with positive direction the one *facing upwards*);

- the point A is described by the coordinates $(x_a, y_a) = (0, 0)$ (so it is at the origin),
- the point $B = (x_b, y_b)$ is placed anywhere in the lower semi-plane (in fact we must require that $y_b < 0$ in order to have a free fall of the object).

Considering that at the initial position A both the kinetic and potential energy are zero, due to the conservation of energy we can derive that

$$0 = mgy + \frac{1}{2}mv^2 \qquad \Longrightarrow \qquad v = \sqrt{-2gy}$$

We so have the velocity v as depending from the vertical coordinate y of the point, and so the next step is to relate the arc length $\mathrm{d}s$ with the horizontal variation $\mathrm{d}x$: we can see that

$$\mathrm{d}s = \sqrt{\mathrm{d}x^2 + \mathrm{d}y^2} = \sqrt{\mathrm{d}x^2 + y'^2(x)\mathrm{d}x^2} = \sqrt{1 + y'^2}\mathrm{d}x$$

The initial brachistochrone problem can so be states as the following functional minimization problem:

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(y) = \int_{x_a}^{x_b} L(y, y')\,\mathrm{d}x \\ &\text{subject to:} && y\big(x_a\big) = y_a = 0 \qquad y\big(x_b\big) = y_b < 0 \end{aligned}$$

where the independent variable is the x coordinate, and

$$L(y, y') = \sqrt{\frac{1 + y'^2}{-2gy}}$$

Because L does not depend explicitly on x, the Beltrami identity gives

$$L - y'L_{y'} = \frac{1}{\sqrt{-2gy\,(1 + y'^2)}} = C.$$

Differentiating this first integral, or applying the Euler–Lagrange equation directly, gives the boundary value problem

$$\begin{cases} 1 + y'(x)^2 + 2\,y(x)y''(x) = 0 \\ y(0) = 0 \\ y(x_b) = y_b \end{cases}$$

Even if it's not clear at first sight, it can be proven that the solution of the differential equation is a cycloid described by the equations

$$x(\theta) = \frac{1}{2}k^2\big(\theta - \sin\theta\big), \qquad y(\theta) = -\frac{1}{2}k^2\big(1 - \cos\theta\big)$$

where the constant k and the range of the new angle θ of the parametrization can be found by using the boundary conditions.

In particular, the initial condition on points A allows to state that $\theta_a = 0$ (in fact $x(0) = \frac{1}{2}k^2 0 = 0$ and also $y(0) = 0$), while considering the final conditions we can determine both θ_b and k solving this system of non-linear equation in such variables:

$$\begin{cases} x_b = \frac{1}{2}k^2\big(\theta_b - \sin\theta_b\big) \\ y_b = -\frac{1}{2}k^2\big(1 - \cos\theta_b\big) \end{cases}$$

16.5 Second Variation and Legendre Condition

The Euler–Lagrange equation is only a first-order necessary condition. For

$$\mathcal{F}(x) = \int_a^b L(x, x', t)\,\mathrm{d}t,$$

with fixed endpoints, the second variation at a smooth extremal $x^\star$ in the direction η, $\eta(a) = \eta(b) = 0$, is

$$\delta^2\mathcal{F}(x^\star;\eta) = \int_a^b \left[L_{xx}\eta^2 + 2L_{xx'}\eta\eta' + L_{x'x'}(\eta')^2\right]\,\mathrm{d}t,$$

where all derivatives of L are evaluated along $(x^\star, x^{\star\prime}, t)$. A necessary condition for a weak local minimum is the **Legendre condition**

$$L_{x'x'}(x^\star, x^{\star\prime}, t) \geq 0.$$

The strengthened condition $L_{x'x'} > 0$ is useful but is not by itself sufficient for a minimum; positivity of the complete second variation must be controlled.

16.5.1 Jacobi Condition

After integrating the mixed term by parts, write

$$\delta^2\mathcal{F} = \int_a^b \left[P(t)(\eta')^2 + Q(t)\eta^2\right]\,\mathrm{d}t,$$

with $P = L_{x'x'}$ and $Q = L_{xx} - \frac{\mathrm{d}}{\mathrm{d}t}L_{xx'}$. The associated Jacobi equation is

$$-\frac{\mathrm{d}}{\mathrm{d}t}\big(P(t)v'(t)\big) + Q(t)v(t) = 0.$$

A point $c > a$ conjugate to a is a point for which a nonzero Jacobi field satisfies $v(a) = v(c) = 0$. Under the strengthened Legendre condition, absence of conjugate points on $(a, b]$ is the classical Jacobi criterion associated with positivity of the second variation.

16.6 Weierstrass Necessary Condition

For a strong local minimum, first-order stationarity is not enough. Define the Weierstrass excess function

$$E(x, p, q, t) = L(x, q, t) - L(x, p, t) - (q - p)L_{x'}(x, p, t).$$

A smooth strong local minimizer satisfies

$$E(x^\star(t), x^{\star\prime}(t), q, t) \geq 0$$

for every admissible slope q. Convexity of L with respect to x' implies this inequality. The Weierstrass condition is therefore stronger than the pointwise Legendre condition and is relevant when perturbations may change the derivative substantially while keeping the curve uniformly close.

17. Higher-Order Functionals

17.1 Euler–Poisson Equation

Considering now the functional $\mathcal{B}(x)$ defined as

$$\mathcal{B}(x) = \int_a^b L(x(t), x'(t), x''(t), t)\,\mathrm{d}t \tag{17.1}$$

the problem to solve now is the minimization of $\mathcal{B}(x)$ for all the function $x \in \mathbb{V}$ where the functional space is defined as

$$\mathbb{V} = \left\{ x \text{ such that } \begin{array}{l} x \in C^4([a,b]) \\ x(a) = x_a, x'(a) = x'_a, x(b) = x_b, x'(b) = x'_b \end{array} \right\}$$

and directional derivatives $\delta x \in \mathbb{D}$ in the functional domain

$$\mathbb{D} = \left\{ \delta x \text{ such that } \begin{array}{l} \delta x \in C^\infty([a,b]) \\ \delta x(a) = \delta x'(a) = \delta x(b) = \delta x'(b) = 0 \end{array} \right\}$$

When trying to calculate the directional derivative of the functional $\mathcal{B}$ (skipping all the unnecessary computation that's similar to the cases yet studies) we end up to the following results:

$$\begin{aligned} \left.\frac{\mathrm{d}}{\mathrm{d}\alpha}\right|_{\alpha=0} \mathcal{B}(x + \alpha\delta x) &= \left.\frac{\mathrm{d}}{\mathrm{d}\alpha}\right|_{\alpha=0} \int_a^b L\big(x + \alpha\,\delta x, x' + \alpha\,\delta x', x'' + \alpha\,\delta x'', t\big)\,\mathrm{d}t \\ &= \int_a^b \left(\frac{\partial L}{\partial x}(x, x', x'', t)\delta x + \frac{\partial L}{\partial x'}\delta x' + \frac{\partial L}{\partial x''}\delta x'' \right) \mathrm{d}t \end{aligned}$$

To *cancel out* the terms involving the terms $\delta x'$, $\delta x''$ it's necessary to use integration by parts considering the following derivatives:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}\delta x\right) = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}\right)\delta x + \frac{\partial L}{\partial x'}\delta x'$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x''}\delta x'\right) = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x''}\right)\delta x' + \frac{\partial L}{\partial x''}\delta x''$$

and so with that said the derivative becomes

$$\begin{aligned}\delta\mathcal{B} &= \int_a^b \frac{\partial L}{\partial x}\delta x + \frac{\mathrm{d}}{\mathrm{d}t}\cancel{\left(\frac{\partial L}{\partial x'}\delta x\right)} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}\delta x \\ &\quad + \cancel{\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x''}\delta x'\right)} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}\delta x'\,\mathrm{d}t \\ &= \int_a^b \left[\left(\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}\right)\delta x - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}\delta x'\right]\mathrm{d}t\end{aligned}$$

In the first line the terms are cancelled because if evaluated singularly we can see that they become in the form

$$\int_a^b \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}\delta x\right)\mathrm{d}t = \left[\frac{\partial L}{\partial x'}\delta x\right]_a^b \xrightarrow{\delta x(a)=\delta x(b)=0} 0$$

To finish the analysis of the derivation we have to consider one last integration by part for the element

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}\delta x\right) = \frac{\mathrm{d}^2}{\mathrm{d}t^2}\frac{\partial L}{\partial x''}\delta x + \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}\delta x'$$

$$\implies \qquad \delta\mathcal{B}(x) = \int_a^b \underbrace{\left(\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'} + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\frac{\partial L}{\partial x''}\right)}_{=f}\delta x\,\mathrm{d}t \tag{17.2}$$

Using so the fundamental lemma of calculus of variation in order to determine the function x that minimize the functional $\mathcal{B}$ we need to solve the system of differential equation

$$\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'} + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\frac{\partial L}{\partial x''} = 0$$

subject to the boundary conditions initially defined.

More generally, for

$$\mathcal{J}(x) = \int_a^b L\left(t, x, x', \dots, x^{(m)}\right)\mathrm{d}t,$$

the stationary functions satisfy the **Euler–Poisson equation**

$$\sum_{k=0}^{m}(-1)^k\frac{\mathrm{d}^k}{\mathrm{d}t^k}\left(\frac{\partial L}{\partial x^{(k)}}\right) = 0.$$

▪ **Example 17.1 — Fourth-Order Example.** Let's consider the problem

$$\begin{aligned}\text{minimize:} \qquad & \int_a^b \left(\left(x''\right)^2 - x^2 + t\right)\mathrm{d}t \\ \text{with:} \qquad & x(a) = 0, \quad x'(a) = 1 \\ & x(b) = 1, \quad x'(b) = 2\end{aligned}$$

In this case the function in minimization functional is

$$L(x, x', x'', t) = (x'')^2 - x^2 + t$$

and all the boundaries conditions are well defined. In this case to calculate the variation of the functional

$$\mathcal{A}(x) = \int_a^b L(x, x', x'', t)\,\mathrm{d}t$$

by using Equation 17.2:

$$\delta\mathcal{A} = \int_a^b \left(-2x + 2\frac{\mathrm{d}^2x''}{\mathrm{d}t^2}\right)\delta x\,\mathrm{d}t$$

The fundamental lemma gives the differential equation satisfied by every smooth stationary candidate:

$$\begin{cases} x'''' - x = 0 \\ x(a) = 0, x'(a) = 1 \\ x(b) = 1, x'(b) = 2 \end{cases}$$

As we can see the differential equation is of order 4 and having 4 boundary condition and so it's possible to compute the solution.

▪

17.2 Natural Boundary Conditions

So far we have considered minimization problems in which every boundary value is prescribed. This need not be the case. Equation Equation 17.2 assumes that all admissible endpoint variations $\delta x^{(k)}$ vanish. When some endpoint values are free, the general expression for the first variation is

$$\begin{aligned} \delta\mathcal{B}(x;\delta x) &= \int_a^b \left(\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'} + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\frac{\partial L}{\partial x''} \right) \delta x \, \mathrm{d}t \\ &+ \left[\left(\frac{\partial L}{\partial x'} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''} \right) \delta x + \frac{\partial L}{\partial x''}\delta x' \right]_a^b \\ &= \int_a^b (A)\delta x \, \mathrm{d}t + (B)\delta x(b) - (C)\delta x(a) + (D)\delta x'(b) - (E)\delta x'(a) \end{aligned}$$

where

$$\begin{aligned} (A) &= \frac{\partial L}{\partial x}(x, x', t) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}(x, x', t) + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\frac{\partial L}{\partial x''}(x, x', t) \\ (B) &= \frac{\partial L}{\partial x'}(x(b), x'(b), b) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}(x(b), x'(b), b) \\ (C) &= \frac{\partial L}{\partial x'}(x(a), x'(a), a) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}(x(a), x'(a), a) \\ (D) &= \frac{\partial L}{\partial x''}(x(b), x'(b), x''(b), b) \\ (E) &= \frac{\partial L}{\partial x''}(x(a), x'(a), x''(a), a) \end{aligned}$$

In this case if we consider that the boundary conditions fixed are only the one

$$x(a) = x_a \qquad x'(b) = x'_b$$

this also reflect on the functional domain of the variations $\delta x \in \mathbb{D}$ that becomes

$$\widetilde{\mathbb{D}} = \left\{ \delta x \in C^\infty([a, b]) \text{ such that } \delta x(a) = \delta x'(b) = 0 \right\}$$

We can now see that in general the domain $\widetilde{\mathbb{D}}$ of the variation set condition for value of the function (and it's derivative) on points only where the boundary conditions are defined. In the case described considering the values fixed it means that the solution for the minimum is the one that satisfies:

$$\delta\mathcal{B}(x;\delta x) = \int_a^b (A)\delta x \, \mathrm{d}t \quad + (B)\delta x(b) - (E)\delta x'(a) \tag{17.3}$$

for all $\delta x \in \widetilde{\mathbb{D}}$. In this case we have 2 boundary degrees of freedom for the variation that we can define $\delta x(b) = \delta_{xb}$ and $\delta x'(a) = \delta'_{xa}$ (that are real evaluated variable); the simplest function that we might determine in order to create a variation $\delta x(t)$ is a polynomial function depending on this parameters, and so

$$\delta x(t) = \delta'_{xa}(t-a) + \frac{2\delta'_{xa}(a-b) + 3\delta_{xb}}{(a-b)^2}\big(t-a\big)^2 + \frac{\delta'_{xa}(a-b) + 2\delta_{xb}}{(a-b)^3}\big(t-a\big)^3$$

Because the Gâteaux derivative $\delta\mathcal{B}(x;\delta x)$ must vanish for every admissible variation $\delta x(t)$, choose a variation with $\delta_{xb} = 1$ and $\delta'_{xa} = 0$. Then

$$\delta x(t) = \frac{3}{(a-b)^2}\big(t-a\big)^2 + \frac{2}{(a-b)^3}\big(t-a\big)^3$$

Considering now that $\delta x'(a) = 0$ we can see that in expression Equation 17.3 the terms (E) is free and in order to have a null derivative it must be that $(B) = 0$. Similarly we can create a particular polynomial (in particular considering

$$\delta_{xb} = \delta x(b) = 0 \quad \text{and} \quad \delta'_{xa} = 1$$

that set to zero the coefficients associated to (B) (and so leaving it *free* to change) and so, to have a zero evaluated derivative, it must be that $(E) = 0$.

In this case the differential equation associated to the terms B, E are called **transversality conditions** and are necessary when the boundary conditions of the problem are not enough (in fact that expressions are evaluated at precise point on the *border* of the domain). At this point the solution of the original problem becomes

$$\begin{cases} \frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'} + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\frac{\partial L}{\partial x''} = 0 \\ \frac{\partial L}{\partial x'}\Big|_b - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}\Big|_b = 0 \\ \frac{\partial L}{\partial x''}\Big|_a = 0 \\ x(a) = x_a,\ x'(b) = x'_b \end{cases}$$

▪ **Example 17.2 — Partly Free Boundaries.** Let's consider the problem of minimizing the functional $\mathcal{A}$ as in Example 17.1 here reported

$$\text{minimize:} \qquad \int_a^b \Big(\big(x''\big)^2 - x^2 + t\Big)\,\mathrm{d}t$$

where the boundary conditions in this case are only

$$x(a) = 0 \qquad x'(b) = 2$$

In this case the boundary conditions are not enough and the case is similar to the theory yet described: in this case the domain of the variations δx is described as

$$\mathbb{D} = \Big\{\delta x \in C^\infty([a,b]) \text{ such that } \delta x(a) = \delta x'(b) = 0\Big\}$$

No information are set for the values $\delta x(b)$ and $\delta x'(a)$ are free, producing the following two transversality conditions:

$$\frac{\partial L}{\partial x'}\Big|_b - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x''}\Big|_b = 0 - 2x'''\Big|_b = -2x'''(b) = 0$$
$$\frac{\partial L}{\partial x''}\Big|_a = 2x''(a) = 0$$

Together with the Euler–Poisson equation and the prescribed boundary data, these conditions determine the stationary boundary value problem:

$$\begin{cases} x'''' - x = 0 \\ x'''(b) = 0 \\ x''(a) = 0 \\ x(a) = 0, x'(b) = 2 \end{cases}$$

▪

▪ **Example 17.3 — Missing Stationary Point.** Let's consider the problem

$$\text{minimize:} \quad \mathcal{A}(y) = \int_0^1 \left(\frac{\left(y'(x)\right)^2}{2} + y(x)y'(x) + y(x) \right) \mathrm{d}x$$
$$\text{with:} \quad y(1) = 1$$

In this case the lagrangian of the problem is defined as

$$L(y, y', x) = \frac{(y')^2}{2} + yy' + y$$

the first variation of this functional so becomes

$$\delta\mathcal{A}(y; \delta y) = \int_0^1 \left(\frac{\partial L}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}\frac{\partial L}{\partial y'} \right) \delta y \, \mathrm{d}x + \left[\frac{\partial L}{\partial y'} \delta y \right] \Big|_{x=0}^{x=1} = 0$$

To formally compute the derivative we firstly need to define the domain of the variation δy that, having only one boundary condition, is

$$\mathbb{D} = \left\{ \delta y \in C^\infty([0,1]) \text{ with } \delta y(1) = 0 \right\}$$

At this point the Gateaux derivative can be computed as

$$\delta\mathcal{A}(y; \delta y) = \int_0^1 \underbrace{\left((y' + 1) - (y'' + y')\right)}_{=f} \delta y \, \mathrm{d}x + \Big[\left(y' + y\right) \delta y \Big]_{x=0}^{x=1}$$
$$= \int_0^1 1 - y'' \delta y \, \mathrm{d}x - \underbrace{\left(y' + y\right) \delta y(0)}_{\text{trasv. cond.}}$$

As we can see we have that the term into the integral (associated to y'') related to the fundamental lemma must be zero and so represent the differential equation associated to the solution of the problem while the second term $y' + y$ evaluated for $x = 0$ represent the transversality condition that allow to have a unique solution of the system of ordinary differential equation that is:

$$\begin{cases} y''(x) = 1 \\ y'(0) + y(0) = 0 \\ y(1) = 1 \end{cases}$$

By integration of the first differential equation we can determine the parametric solution of the system whose coefficients can be matched considering the boundary conditions:

$$y(x) = \frac{1}{2}x^2 + c_1 x + c_2$$

Substituting the parametric solution on the boundary conditions we can solve for the parameters:

$$\begin{cases} c_1 + c_2 = 0 \\ \frac{1}{2} + c_1 + c_2 = 1 \end{cases}$$

The two equations are inconsistent. Therefore the functional has no smooth stationary point satisfying the stated boundary condition; this conclusion concerns the first-order necessary conditions and is not, by itself, a general existence theorem.

▪

17.3 Corner Conditions

An extremal may be only piecewise smooth. If x is continuous and has an interior corner at $t = c$, independent variations on the two sides give the **Weierstrass–Erdmann conditions**

$$\left[\frac{\partial L}{\partial x'}\right]_{c^-}^{c^+} = 0, \qquad \left[L - x'\frac{\partial L}{\partial x'}\right]_{c^-}^{c^+} = 0.$$

The first condition expresses continuity of the conjugate momentum. If the corner location is also free, stationarity with respect to its displacement yields the second condition, i.e. continuity of the Beltrami quantity. For vector-valued curves the momentum condition is imposed componentwise.

17.4 Endpoint Constraints

Consider the minimization of a functional $\mathcal{F}$ subject to an equality constraint at the boundary:

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(z) = \int_a^b L(z, z', t)\,\mathrm{d}t \\ &\text{subject to:} && w\big(z(a), z(b)\big) = 0 \end{aligned} \tag{17.4}$$

In this case we want to find the solutions for the function $z(t)$ in the functional spaced defined as

$$\mathbb{V} = \Big\{ z \in C^2([a,b]) \text{ such that } w\big(z(a), z(b)\big) = 0 \Big\}$$

In this case we cannot consider the linearity of the functional space (in fact two functions z_1, z_2 can satisfy the constraint b, but their sum $z_1 + z_2$ doesn't belong to the domain), and more difficult is the definition of the directional domain $\mathbb{D}$ on which we can compute all the possible derivatives.

17.4.1 Discrete Derivation

A way to derive the endpoint conditions is to discretize $[a, b]$ into n equal subintervals of length $h = (b - a)/n$; the axis t is discretized at

$$t_k = t_0 + kh = a + k\frac{b-a}{n}$$

With this definition we can compute the function z in the various point considering that $z(t_k) = z_k$ (and indeed we can also note that $t_0 = a$ and $t_n = b$). Using the mid-point squaring numerical method to integrate the original function, we can approximate the functional as

$$\mathcal{F}(z) = \int_a^b L(z, z', t)\,\mathrm{d}t \approx h\sum_{k=1}^{n} L\left(z_{k-\frac{1}{2}}, z'_{k-\frac{1}{2}}, t_{k-\frac{1}{2}}\right)$$

where

$$z_{k-\frac{1}{2}} = \frac{z_k + z_{k-1}}{2} \qquad z'_{k-\frac{1}{2}} = \frac{z_k - z_{k-1}}{h} \qquad t_{k-\frac{1}{2}} = t_k - \frac{h}{2}$$

With this being said the initial constrained minimization problem (Equation 17.4) can be discretized so obtaining the form

$$\begin{aligned} &\text{minimize:} && f(z) = h\sum_{k=1}^{n} L\left(z_{k-\frac{1}{2}}, z'_{k-\frac{1}{2}}, t_{k-\frac{1}{2}}\right) \\ &\text{subject to:} && w\big(z_0, z_n\big) = 0 \end{aligned} \tag{17.5}$$

where $z = \big(z_0, z_1, \dots, z_n\big)^t$ is the vector off all the discretized values of the initial function $z(t)$. This formulation recall the constrained minimization problem with equality constraints and so we can use the Lagrange multiplier method by defining the expression

$$\mathscr{L}(z, \lambda) = f(z) - \lambda w(z_0, z_n)$$

To solve this kind of problem we need to find the stationary point of the yet built lagrangian $\mathcal{L}$,and so this means solving the following non-linear system determined by the equations

$$\frac{\partial \mathscr{L}}{\partial z_i} = 0 \qquad \forall i = 0, 1, \dots, n \qquad \frac{\partial \mathscr{L}}{\partial \lambda} = 0$$

Starting with $i = 0$ we can compute that derivative of the lagrangian as

$$\begin{aligned} \frac{\partial \mathscr{L}}{\partial z_0} &= \frac{\partial}{\partial z_0}\left(h\sum_{k=1}^{n} L\left(\frac{z_k + z_{k-1}}{2}, \frac{z_k - z_{k-1}}{h}, t_{k-\frac{1}{2}}\right) - \lambda w(z_0, z_n)\right) \\ &= \frac{\partial}{\partial z_0}\Bigg(hL\Big(\underbrace{\frac{z_1 + z_0}{2}, \frac{z_1 - z_0}{h}, t_{\frac{1}{2}}}_{[1]}\Big) - \lambda w(z_0, z_n)\Bigg) \\ &= h\frac{\partial L([1])}{\partial z}\frac{\partial}{\partial z_0}\left(\frac{z_1 + z_0}{2}\right) + h\frac{\partial L([1])}{\partial z'}\frac{\partial}{\partial z_0}\left(\frac{z_1 - z_0}{h}\right) - \lambda\frac{\partial w(z_0, z_n)}{\partial z_0} \\ &= \frac{h}{2}\frac{\partial L([1])}{\partial z} - \frac{\partial L([1])}{\partial z'} - \lambda\frac{\partial w(z_0, z_n)}{\partial z_0} \end{aligned}$$

Note that passing from the first to the second line only the terms associated to $k = 1$ present terms depending on z_0, and so only that part of the summation has been considered.

For all the other values $i \neq 0, n$, the mathematical expression of the derivative $\partial\mathscr{L}/\partial z_i$ becomes more *complex* (due to the fact that we have to consider two elements of the summation) and so we can use the simplified notation to express the partial terms such

$$\frac{\partial L}{\partial z}\Big|_{k+\frac{1}{2}} := \frac{\partial}{\partial z} L\left(\frac{z_{k+1} + z_k}{2}, \frac{z_{k+1} - z_k}{h}, t_{k+\frac{1}{2}}\right)$$

With this, doing the steps as previously shown, we can compute the partial derivatives as

$$\begin{aligned} \frac{\partial \mathscr{L}}{\partial z_k} &= \frac{\partial}{\partial z_k}\left(h\sum_{j=1}^{n} L\left(\frac{z_j + z_{j-1}}{2}, \frac{z_j - z_{j-1}}{h}, t_{j-\frac{1}{2}}\right) - \lambda w(z_0, z_n)\right) \\ &= \frac{h}{2}\left(\frac{\partial L}{\partial z}\Big|_{k-\frac{1}{2}} + \frac{\partial L}{\partial z}\Big|_{k+\frac{1}{2}}\right) + \frac{\partial L}{\partial z'}\Big|_{k-\frac{1}{2}} - \frac{\partial L}{\partial z'}\Big|_{k+\frac{1}{2}} \\ &= h\left(\frac{\frac{\partial L}{\partial z}\Big|_{k-\frac{1}{2}} + \frac{\partial L}{\partial z}\Big|_{k+\frac{1}{2}}}{2} - \frac{\frac{\partial L}{\partial z'}\Big|_{k+\frac{1}{2}} - \frac{\partial L}{\partial z'}\Big|_{k-\frac{1}{2}}}{h}\right) \end{aligned}$$

The last partial derivative (computed for $k = n$) is instead

$$\frac{\partial \mathscr{L}}{\partial z_n} = \frac{h}{2}\frac{\partial L}{\partial z}\Big|_{k-\frac{1}{2}} + \frac{\partial L}{\partial z'}\Big|_{k-\frac{1}{2}} - \lambda\frac{\partial w(z_0, z_n)}{\partial z_n}$$

With all this calculation being done we determine that the non-linear system of equations representing the first order necessary condition for the minimum point of the lagrangian $\mathscr{L}$ is

$$\begin{cases} \dfrac{\frac{\partial L}{\partial z}\Big|_{k-\frac{1}{2}} + \frac{\partial L}{\partial z}\Big|_{k+\frac{1}{2}}}{2} - \dfrac{\frac{\partial L}{\partial z'}\Big|_{k+\frac{1}{2}} - \frac{\partial L}{\partial z'}\Big|_{k-\frac{1}{2}}}{h} = 0 & : A \\ \frac{h}{2}\frac{\partial L}{\partial z}\Big|_{\frac{1}{2}} - \frac{\partial L}{\partial z'}\Big|_{\frac{1}{2}} - \lambda\frac{\partial w(z_0, z_n)}{\partial z_0} = 0 & : B \\ \frac{h}{2}\frac{\partial L}{\partial z}\Big|_{n-\frac{1}{2}} + \frac{\partial L}{\partial z'}\Big|_{n-\frac{1}{2}} - \lambda\frac{\partial w(z_0, z_n)}{\partial z_n} = 0 & : C \end{cases}$$

By pushing the limit for $h \to 0$ (to have a *continuous discretization*), we can see a correlation of the two terms composing equation A, in fact

$$\begin{aligned} \frac{\frac{\partial L}{\partial z}\Big|_{k-\frac{1}{2}} + \frac{\partial L}{\partial z}\Big|_{k+\frac{1}{2}}}{2} &\approx \frac{\partial L}{\partial z}\left(z_k, z'_k, t_k\right) \\ \frac{\frac{\partial L}{\partial z'}\Big|_{k+\frac{1}{2}} - \frac{\partial L}{\partial z'}\Big|_{k-\frac{1}{2}}}{h} &\approx \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial z'}\left(z_k, z'_k, t_k\right)\right) \end{aligned}$$

Similarly both the terms A and B presents the terms similar to the previous definition and they also presents another contribute due to the Lagrange multiplier λ. By so considering the limit $h \to 0$ we can see that $k = \frac{1}{2}$ tends to be a while $n - \frac{1}{2}$ tends to b, and so we can rewrite the systems of non-linear equations as

$$\begin{cases} \frac{\partial L}{\partial z}(z, z', t) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial z'}(z, z', t) = 0 \\ -\frac{\partial L}{\partial z'}(z(a), z'(a), a) - \lambda\frac{\partial w(z(a), z(b))}{\partial z(a)} = 0 \\ \frac{\partial L}{\partial z'}(z(b), z'(b), b) - \lambda\frac{\partial w(z(a), z(b))}{\partial z(b)} = 0 \\ w\big(z(a), z(b)\big) = 0 \end{cases}$$

This is so the general formulation of the initial problem of minimizing a functional

$$\mathcal{F}(z) = \int_a^b L(z, z', t)\,\mathrm{d}t$$

subject to an equality constraint w.

17.5 Boundary Multipliers

The same result can also be achieved with an heuristic formulation. Given so the problem of minimizing a functional $\mathcal{F}$ subjected to an equality constraint w (as in Equation 17.4), the solution can be achieved by determining a new functional $\mathcal{L}$ defined as the lagrangian of the system:

$$\mathcal{L}(z, \lambda) = \int_a^b L(z, z', t)\,\mathrm{d}t - \lambda w\big(z(a), z(b)\big)$$

At this point we can compute the variation of this functional considering it's Gateaux derivative δ that's

$$
\begin{aligned}
\delta\mathcal{L}(z, \lambda; \delta z, \delta\lambda) &= \frac{\mathrm{d}}{\mathrm{d}\alpha}\Big|_{\alpha=0} \mathcal{L}\big(z + \alpha\,\delta z, \lambda + \alpha\,\delta\lambda\big) \\
&= \delta \int_a^b L(z, z', t)\,\mathrm{d}t - \delta\lambda\, w\big(z(a), z(b)\big) - \lambda\,\delta w\big(z(a), z(b)\big) \\
&= \int_a^b \left(\frac{\partial L}{\partial z}\delta z + \frac{\partial L}{\partial z'}\delta z'\right)\mathrm{d}t - \delta\lambda\, w\big(z(a), z(b)\big) \\
&- \lambda\left(\frac{\partial w(z(a), z(b))}{z(a)}\delta z(a) + \frac{\partial w(z(a), z(b))}{z(b)}\delta z(b)\right) \\
&= \int_a^b \left(\frac{\partial L}{\partial z} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial z'}\right)\delta z\,\mathrm{d}t + \left[\frac{\partial L}{\partial z'}\delta z\right]\Big|_a^b - \delta\lambda\, w\big(z(a), z(b)\big) \\
&- \lambda\left(\frac{\partial w}{\partial z(a)}\delta z(a) + \frac{\partial w}{\partial z(b)}\delta z(b)\right)
\end{aligned}
$$

By evaluating the partial derivative $\partial\mathcal{L}/\partial z'$ and collecting common terms we can reduce the variation of the functional to the form

$$
\delta\mathcal{L}(z, \lambda; \delta z, \delta\lambda) = \int_a^b (A)\,\delta z\,\mathrm{d}t - \delta\lambda\, w\big(z(a), z(b)\big) + (B)\delta z(a) + (C)\delta z(b)
$$

where

$$
\begin{aligned}
(A) &= \frac{\partial L}{\partial z}(z, z', t) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial z'}(z, z', t) \\
(B) &= -\frac{\partial L}{\partial z'}(z(a), z'(a), a) - \lambda\frac{\partial w}{\partial z(a)}(z(a), z(b)) \\
(C) &= \frac{\partial L}{\partial z'}(z(b), z'(b), b) - \lambda\frac{\partial w}{\partial z(b)}(z(a), z(b))
\end{aligned}
$$

We start by noting that the variation $\delta\mathcal{L}(z, \lambda; \delta z, \delta\lambda)$ must be equal to zero for all variations $\delta z \in C^\infty([a, b])$ and for all constants $\delta\lambda \in \mathbb{R}$.

First, consider the case where $\delta\lambda = 0$, $\delta z(a) = 0$, and $\delta z(b) = 0$. In this scenario, we can apply the fundamental lemma of the calculus of variations, which implies that the term A must equal zero.

Next, let's examine the case where $\delta\lambda \neq 0$ while maintaining $\delta z(a) = 0$ and $\delta z(b) = 0$. Under these conditions, we arrive at the initial equality constraint:

$$
w(z(a), z(b)) = 0.
$$

Now, consider the situation where $\delta\lambda = 0$, $\delta z(a) \neq 0$, and $\delta z(b) = 0$. In this case, we can conclude that the term (B) must also equal zero. Similarly, if we have $\delta z(b) \neq 0$ while keeping $\delta\lambda = 0$ and $\delta z(a) = 0$, we find that the term (C) must be zero as well.

This means that the resultant system of non-linear equation that solves the problem can be expressed as

$$\begin{cases} \frac{\partial L}{\partial z}(z, z', t) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial z'}(z, z', t) = 0 \\ w\big(z(a), z(b)\big) = 0 \\ \frac{\partial L}{\partial z'}(z(a), z'(a), a) + \lambda \frac{\partial w\left(z(a), z(b)\right)}{\partial z(a)} = 0 \\ \frac{\partial L}{\partial z'}(z(b), z'(b), b) - \lambda \frac{\partial w\left(z(a), z(b)\right)}{\partial z(b)} = 0 \end{cases} \tag{17.6}$$

We can observe that the resulting system is equivalent to the one obtained by pushing the limit $h \to 0$ with the discretized version previously performed.

17.5.1 Fixed-End Verification

Considering the common case described by the problem

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(x) = \int_a^b L(x, x', t)\,\mathrm{d}t \\ &\text{subject to:} && x(a) = x_a,\ x(b) = x_b \end{aligned}$$

than the solution can be still achieved using the method yet shown. By in fact building the Lagrangian functional

$$\mathcal{L}(x, \lambda_1, \lambda_2) = \int_a^b L(x, x', t)\,\mathrm{d}t - \lambda_1\big(x(a) - x_a\big) - \lambda_2\big(x(b) - x_b\big)$$

substituting this function in the result of Equation 17.6 we get the following differential system of equations:

$$\begin{cases} \frac{\partial L}{\partial x}(x, x', t) - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}(x, x', t) = 0 \\ x(a) = x_a, \quad x(b) = x_b \\ -\lambda_1 - \frac{\partial L}{\partial x'}(x(a), x'(a), a) = 0 \\ -\lambda_2 + \frac{\partial L}{\partial x'}(x(b), x'(b), b) = 0 \end{cases}$$

The first two lines represent the terms that were always presents in the previous statement of the problem (without the equality constraints), while the last two depending from λ_i gives no real information because they can always be verified (in fact there will always exists a parameter λ_i that equals the derivative $\partial L/\partial x'$); in general the variable λ_i can appear in the other equations, and so this last can be used to determine the stationary solution of the lagrangian $\mathcal{L}$.

▪ **Example 17.4 — Endpoint-Cost Example.** Given the functional

$$\mathcal{F}(x) = x(0) + \int_0^1 tx^2 + \left(x'\right)^2 \mathrm{d}t$$

its first variation is obtained from

$$L(x, x', t) = tx^2 + \left(x'\right)^2$$

so that

$$\frac{\partial L}{\partial x}(x, x', t) = 2tx \qquad \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}(x, x', t) = \frac{\mathrm{d}}{\mathrm{d}t}(2x') = 2x''$$

To compute the complete variation of the system it's mandatory to consider the terms resulting from the integration by part (due to the fact that no boundary conditions are set) related to the term

$$\left[\frac{\partial L}{\partial x'}\delta x\right]_a^b$$

With that said, the overall first variation can be computed as

$$\begin{aligned}\delta\mathcal{F}(x;\delta x) &= \delta x(0) + 2\int_0^1 \big(tx - x''\big)\delta x\,\mathrm{d}t + 2x'\delta x(1) - 2x'\delta x(0)\\ &= 2\int_0^1 \big(tx - x''\big)\delta x\,\mathrm{d}t + 2x'(1)\delta x(1) + \big(1 - 2x'(0)\big)\delta x(0)\end{aligned}$$

and the resulting BVP by the application of fundamental lemma of calculus of variations is

$$\begin{cases} tx - x'' = 0\\ x'(1) = 0\\ 2x'(0) = 1\end{cases}$$

▪

▪ **Example 17.5 — Integral-and-Boundary Example.** Given the problem

$$\begin{aligned}&\text{minimize:} && \mathcal{F}(x) = x(0) + \int_0^1 x^2 + \big(x' - t\big)^2\,\mathrm{d}t\\ &\text{subject to:} && \int_0^1 x\,\mathrm{d}t = 0\\ & && x(1) = 2\end{aligned}$$

the resulting boundary value problem can be computed considering the lagrangian $\mathcal{L}$ of the system that's

$$\begin{aligned}\mathcal{L}(x,\lambda,\mu) &= \mathcal{F}(x) - \lambda\int_0^1 x\,\mathrm{d}t - \mu\big(x(1) - 2\big)\\ &= \int_0^1 \underbrace{x^2 + \big(x' - t\big)^2 - \lambda x}_{L}\,\mathrm{d}t + x(0) - \mu\big(x(1) - 2\big)\end{aligned}$$

Starting from this point we can so compute the first variation of $\mathcal{L}$ that's:

$$\begin{aligned}\delta\mathcal{L} &= \int_0^1 \left(\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}\right)\delta x\,\mathrm{d}t + \left[\frac{\partial L}{\partial x'}\delta x\right]_0^1 - \int_0^1 x\,\delta\lambda\,\mathrm{d}t\\ &\quad + \delta x(0) - \mu\delta x(1) - \big(x(1) - 2\big)\delta\mu\\ &= \int_0^1 \big(-2x'' + 2x + 2 - \lambda\big)\delta x\,\mathrm{d}t\\ &\quad + 2\big(x'(1) - 1\big)\delta x(1) - 2x'(0)\delta x(0)\\ &\quad - \int_0^1 x\delta\lambda\,\mathrm{d}t + \delta x(0) - \mu\delta x(1) - \big(x(1) - 2\big)\delta\mu\end{aligned}$$

The resulting boundary valued problem is so determined by setting to zero the terms that multiply every variation $\delta\cdot$ and so:

$$\begin{cases} \delta x: & 2x'' - 2x - 2 + \lambda = 0 \\ \delta x(0): & 2x'(0) = 1 \\ \delta x(1): & \cancel{2x'(1) - 2 = \mu} \quad : \text{trivially satisfied} \\ \delta\lambda: & \int_0^1 x = 0 \\ \delta\mu: & x(1) = 2 \end{cases}$$

▪

▪ **Example 17.6 — Coupled-Function Example.** Given the functional

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(x, y, z) = y(1) + \int_0^1 x^2 + z^2 + x'y' + z'^2 \, \mathrm{d}t \\ &\text{subject to:} && x(1) = 2 \end{aligned}$$

the resulting boundary value problem can be stated by determining the first variation of the lagrangian

$$\mathcal{L}(x, y, z, \lambda) = y(1) + \int_0^1 \underbrace{x^2 + z^2 + x'y' + z'^2}_{=L(x,y,z,x',y',z')} \, \mathrm{d}t - \lambda\big(x(1) - 2\big)$$

that results in

$$\begin{aligned} \delta\mathcal{L}(x, y, z, \lambda) = \delta y(1) &+ \int_0^1 \left(\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial x'}\right)\delta x \, \mathrm{d}t + \left[\frac{\partial L}{\partial x'}\delta x\right]_0^1 \\ &+ \int_0^1 \left(\frac{\partial L}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial y'}\right)\delta y \, \mathrm{d}t + \left[\frac{\partial L}{\partial y'}\delta y\right]_0^1 \\ &+ \int_0^1 \left(\frac{\partial L}{\partial z} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial z'}\right)\delta z \, \mathrm{d}t + \left[\frac{\partial L}{\partial z'}\delta z\right]_0^1 \\ &- \lambda\delta x(1) - \big(x(1) - 2\big)\delta\lambda \\ &= \delta y(1) + \int_0^1 \big(2x - y''\big)\delta x \, \mathrm{d}t + y'(1)\delta x(1) - y'(0)\delta x(0) \\ &+ \int_0^1 -x''\delta y \, \mathrm{d}t + x'(1)\delta y(1) - x'(0)\delta y(0) \\ &+ \int_0^1 \big(2z - 2z''\big)\delta z \, \mathrm{d}t \\ &+ 2z'(1)\delta z(1) - 2z'(0)\delta z(0) - \lambda\delta x(1) - \big(x(1) - 2\big)\delta\lambda \end{aligned}$$

Setting to zero the terms associated to each variation $\delta\cdot$ determines the following boundary value problem, solution of the functional minimization:

$$\begin{cases} 2x - y'' = 0 & : \delta x \\ x'' = 0 & : \delta y \\ z - z'' = 0 & : \delta z \\ y'(0) = 0 & : \delta x(0) \\ \cancel{y'(1) - \lambda = 0} & : \delta x(1), \text{ trivially solved in } \lambda \\ x'(0) = 0 & : \delta y(0) \\ 1 + x'(1) = 0 & : \delta y(1) \\ z'(0) = 0 & : \delta z(0) \\ z'(1) = 0 & : \delta z(1) \\ x(1) = 2 & : \delta \lambda \end{cases}$$

The system is inconsistent: $x'' = 0$ and $x'(0) = 0$ imply $x'(1) = 0$, whereas the condition from $\delta y(1)$ requires $x'(1) = -1$. Hence this example has no smooth stationary point.

▪

18. Isoperimetric Constraints

We consider the minimization of a functional $\mathcal{A}(x)$ defined as

$$\mathcal{A}(x) = \int_0^1 (x')^2 \, \mathrm{d}t,$$

subject to the following constraints:

1. **Integral Constraint:** $\int_0^1 x \, \mathrm{d}t = 1$,
2. **Boundary Conditions:** $x(0) = 2$ and $x(1) = 1$.

Thus, the minimization problem can be stated as

$$\begin{aligned} \text{minimize} \quad & \mathcal{A}(x) = \int_0^1 (x')^2 \, \mathrm{d}t, \\ \text{subject to} \quad & \int_0^1 x \, \mathrm{d}t = 1, \\ & x(0) = 2, \quad x(1) = 1. \end{aligned}$$

To address this problem, we define a feasible space $\mathbb{W}$ for x, which includes functions satisfying the boundary conditions:

$$\mathbb{W} = \left\{ x \in C^2([0,1]) \,\middle|\, x(0) = 2,\ x(1) = 1,\ \int_0^1 x(t) \, \mathrm{d}t = 1 \right\}.$$

The problem can thus be reformulated as finding $x \in \mathbb{W}$ that minimizes $\mathcal{A}(x)$.

18.1 Feasible Variations

Feasible Directions

To analyze feasible directions for variations of x, we introduce a perturbation $x + \alpha \delta x$ with δx satisfying

$$x(0) + \alpha\delta x(0) = 2,$$
$$x(1) + \alpha\delta x(1) = 1,$$
$$\int_0^1 (x(t) + \alpha\delta x(t))\,\mathrm{d}t = 1.$$

Expanding these conditions yields:

1. $\delta x(0) = 0$,
2. $\delta x(1) = 0$,
3. $\int_0^1 \delta x(t)\,\mathrm{d}t = 0$.

This means the set of feasible variations δx lies in the space

$$\mathbb{V} = \left\{ \delta x \in C^\infty([0,1]) \mid \delta x(0) = 0,\ \delta x(1) = 0,\ \int_0^1 \delta x\,\mathrm{d}t = 0 \right\}.$$

Stationarity

A function $x \in \mathbb{W}$ is stationary for this problem if

$$\left.\frac{\mathrm{d}}{\mathrm{d}\alpha}\right|_{\alpha=0} \mathcal{A}(x + \alpha\delta x) = 0, \quad \forall \delta x \in \mathbb{V}.$$

Calculating this derivative, we get

$$\left.\frac{\mathrm{d}}{\mathrm{d}\alpha}\right|_{\alpha=0} \int_0^1 (x' + \alpha\delta x')^2\,\mathrm{d}t = \int_0^1 2x'\delta x'\,\mathrm{d}t.$$

Using integration by parts, we find

$$\int_0^1 2x'\delta x'\,\mathrm{d}t = 2x'\delta x\Big|_0^1 - 2\int_0^1 x''\delta x\,\mathrm{d}t.$$

Since $\delta x(0) = \delta x(1) = 0$, the boundary term vanishes, leaving

$$0 = \int_0^1 x''\delta x\,\mathrm{d}t \quad \forall \delta x \in \mathbb{V}.$$

At this point, we cannot directly conclude that $x'' = 0$ because $\mathbb{V}$ is not rich enough for the fundamental lemma of the calculus of variations to apply.

Constructing a Feasible Variation

To circumvent this limitation, consider an auxiliary function $\delta y \in C^\infty([0,1])$ with $\delta y(0) = \delta y(1) = 0$. Define δx as

$$\delta x(t) = \delta y(t) - \lambda t(1-t)\int_0^1 \delta y(\tau)\,\mathrm{d}\tau.$$

Choosing $\lambda = 6$ ensures that $\delta x \in \mathbb{V}$.

Differential Equation

Using this construction in our stationarity condition, we obtain

$$0 = \int_0^1 \left(x''(t) - 6\left(x(1) + x(0) - 2\int_0^1 x(\tau)\,\mathrm{d}\tau \right) \right) \delta y(t)\,\mathrm{d}t.$$

Now, since δy satisfies the conditions for the fundamental lemma of calculus of variations, we conclude that

$$x''(t) = 6\left(x(1) + x(0) - 2\int_0^1 x(\tau)\,\mathrm{d}\tau \right).$$

Boundary Conditions

Substituting the boundary conditions and integral constraint, we get

$$x''(t) = 6, \quad x(0) = 2, \quad x(1) = 1.$$

Solving this system, we find that

$$x(t) = 3t^2 - 4t + 2,$$

which satisfies all the boundary conditions and integral constraint.

Because the objective is strictly convex on the affine feasible set, this stationary function is the unique global minimizer.

Remark

While this approach correctly finds the solution, it would be advantageous to simplify the derivation by seeking a direct application of the fundamental lemma of calculus of variations. This simplification would streamline the solution process and avoid the need for constructing additional variations.

18.2 Finite Differences

To approximate the variational problem

$$\begin{aligned} \text{minimize} \quad & \mathcal{A}(x) = \int_0^1 (x')^2 \, \mathrm{d}t, \\ \text{subject to} \quad & \int_0^1 x \, \mathrm{d}t = 1, \\ & x(0) = 2, \quad x(1) = 1, \end{aligned}$$

we apply the finite difference method. This approach transforms the continuous problem into a discrete one, allowing us to approximate the solution numerically.

18.2.1 Discretization Setup

1. **Divide the Interval:** Let n be the number of intervals, and define $h = \frac{1}{n}$ as the interval width. We set sample points $t_k = kh$ for $k = 0, 1, \ldots, n$, where $t_0 = 0$ and $t_n = 1$.
2. **Approximate the Solution:** Define $y_k \approx x(t_k)$ as the approximate value of the solution at each sample point t_k.
3. **Quadrature Formula for the Integral:** To approximate the integral of $(x')^2$, we use the midpoint rule:
$$\int_0^1 (x')^2 \, \mathrm{d}t \approx h \sum_{k=1}^{n} \left(x'(t_{k-\frac{1}{2}})\right)^2 + \mathcal{O}(h^2),$$
where $x'(t_{k-\frac{1}{2}})$ represents the approximate derivative at the midpoint between t_{k-1} and t_k.
4. **Finite Difference Approximation for the Derivative:** The first derivative at $t_{k-\frac{1}{2}}$ can be approximated by
$$x'(t_{k-\frac{1}{2}}) = \frac{x(t_k) - x(t_{k-1})}{h} + \mathcal{O}(h^2).$$
5. **Quadrature Formula for the Integral constraint:** To approximate the integral of x, we use the trapezoidal rule:

$$\int_0^1 x(t)\,\mathrm{d}t = \frac{h}{2}(x(0) + x(1)) + h\sum_{k=1}^{n-1} x(t_k) + \mathcal{O}(h^2),$$

Using this setup, we can now write the discretized form of the problem.

18.2.2 Discretized Problem Formulation

Substituting the finite difference approximation into the functional, the discretized problem becomes

$$\begin{aligned}
\text{minimize} \quad & \mathcal{A}_h(\boldsymbol{x}) = h\sum_{k=1}^{n}\left(\frac{x_k - x_{k-1}}{h}\right)^2, \\
\text{subject to} \quad & h_1(\boldsymbol{x}) = \frac{h}{2}(x_0 + x_n) + h\sum_{k=1}^{n-1} x_k - 1 = 0, \\
& h_2(\boldsymbol{x}) = x_0 - 2 = 0, \\
& h_3(\boldsymbol{x}) = x_n - 1 = 0.
\end{aligned}$$

This formulation transforms our continuous problem into a finite-dimensional optimization problem, where we seek values of $\boldsymbol{x}$ that minimize $\mathcal{A}_h(\boldsymbol{x})$ while satisfying the discretized constraints.

18.2.3 Discrete Lagrangian

Let λ_1, λ_2, and λ_3 be the Lagrange multipliers corresponding to constraints $h_1(\boldsymbol{x})$, $h_2(\boldsymbol{x})$, and $h_3(\boldsymbol{x})$, respectively. We define the **Lagrange function** as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{x}, \lambda_1, \lambda_2, \lambda_3) = {} & \frac{1}{h}\sum_{k=1}^{n}(x_k - x_{k-1})^2 \\
& - \lambda_1\left(\frac{h}{2}(x_0 + x_n) + h\sum_{k=1}^{n-1} x_k - 1\right) \\
& - \lambda_2 \cdot (x_0 - 2) + \lambda_3 \cdot (x_n - 1).
\end{aligned}$$

18.2.4 First-Order Conditions

To find the stationary points of $\mathcal{L}$, we compute the partial derivatives with respect to each x_k and set them equal to zero:

1. **For $k = 1, 2, \ldots, n-1$:**

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial x_k} &= \frac{2}{h}(x_k - x_{k-1}) - \frac{2}{h}(x_{k+1} - x_k) - \lambda_1 \cdot h \\
&= h\left(\frac{2}{h^2}(2x_k - x_{k-1} - x_{k+1}) - \lambda_1\right) = 0.
\end{aligned}$$

2. **For $k = 0$ (boundary point):**

$$\frac{\partial \mathcal{L}}{\partial x_0} = \frac{2}{h}(x_0 - x_1) - \lambda_1 \cdot \frac{h}{2} - \lambda_2 = 0.$$

3. **For $k = n$ (boundary point):**

$$\frac{\partial \mathcal{L}}{\partial x_n} = \frac{2}{h}(x_n - x_{n-1}) - \lambda_1 \cdot \frac{h}{2} + \lambda_3 = 0.$$

These equations, combined with the constraints, form a system of equations that can be solved for $\boldsymbol{x}$, λ_1, λ_2, and λ_3.

$$\begin{cases} 2\dfrac{x_{k+1}-2x_k+x_{k-1}}{h^2}+\lambda_1=0. \\ x_0=2 \\ x_n=1 \\ \dfrac{2}{h}(x_1-x_0)-\lambda_1\cdot\dfrac{h}{2}=\lambda_2 \\ -\dfrac{2}{h}(x_n-x_{n-1})-\lambda_1\cdot\dfrac{h}{2}=\lambda_3 \end{cases}$$

notice that λ_2 and λ_3 are trivially solved and can be removed to the nonlinear system to obtain

$$\begin{cases} 2\dfrac{x_{k+1}-2x_k+x_{k-1}}{h^2}+\lambda_1=0. \\ x_0=2 \\ x_n=1 \end{cases}$$

which can be interpreted as the discrete approximation of the Boundary Value Problem (BVP)

$$\begin{cases} 2x''+\lambda_1=0. \\ x(0)=2 \\ x(1)=1 \end{cases}$$

solving this last BVP

$$x(t)=-\frac{\lambda_1}{4}t^2+C_1t+C_2$$

using BC

$$\begin{aligned} x(0)=C_2=2, &\qquad C_2=2, \\ x(1)=-\frac{\lambda_1}{4}+C_1+C_2=1, &\qquad C_1=\frac{\lambda_1}{4}-1 \end{aligned}$$

Using integral constraint

$$\begin{aligned} 0&=\int_0^1 x(t)\,\mathrm{d}t-1 \\ &=\int_0^1 -\frac{\lambda_1}{4}t^2+\left(\frac{\lambda_1}{4}-1\right)t+2\,\mathrm{d}t-1 \\ &=-\frac{\lambda_1}{12}+\frac{\lambda_1-4}{8}+2-1 \\ &=\frac{\lambda_1}{24}+\frac{1}{2} \end{aligned}$$

so that $\lambda_1=-12$ and

$$x(t)=-\frac{\lambda_1}{4}t^2+C_1t+C_2=3t^2-4t+2$$

the same solution.

18.3 Lagrange-Multiplier Method

18.3.1 Worked Example

The finite difference approach motivates solving the problem

$$\begin{aligned} \text{minimize} \quad & \mathcal{A}(x) = \int_0^1 (x')^2 \, \mathrm{d}t, \\ \text{subject to} \quad & \int_0^1 x \, \mathrm{d}t = 1, \\ & x(0) = 2, \quad x(1) = 1, \end{aligned}$$

by introducing a Lagrangian functional that incorporates the integral constraint. We define:

$$\begin{aligned} \mathcal{L}(x, \lambda) &= \mathcal{A}(x) - \lambda \left(\int_0^1 x \, \mathrm{d}t - 1 \right) \\ &= \int_0^1 (x')^2 \, \mathrm{d}t - \lambda \left(\int_0^1 x \, \mathrm{d}t - 1 \right). \end{aligned}$$

The stationary points of this Lagrangian, $\mathcal{L}(x, \lambda)$, are candidates for the minimizers. To find these stationary points, we perform the **first variation** of $\mathcal{L}$.

18.3.2 First Variation

The variation of $\mathcal{L}(x, \lambda)$ with respect to both x and λ is:

$$\begin{aligned} \delta\mathcal{L}(x, \lambda; \delta x, \delta\lambda) &= \delta \int_0^1 (x')^2 \, \mathrm{d}t - \delta\lambda \left(\int_0^1 x \, \mathrm{d}t - 1 \right) - \lambda\delta \int_0^1 x \, \mathrm{d}t \\ &= \int_0^1 2x' \delta x' \, \mathrm{d}t - \delta\lambda \left(\int_0^1 x \, \mathrm{d}t - 1 \right) - \lambda \int_0^1 \delta x \, \mathrm{d}t. \end{aligned}$$

Using integration by parts on the term involving x' and using $\delta x(0) = \delta x(1) = 0$, we get:

$$\begin{aligned} \delta\mathcal{L}(x, \lambda; \delta x, \delta\lambda) &= \int_0^1 2x' \delta x' - \lambda \delta x \, \mathrm{d}t - \delta\lambda \left(\int_0^1 x \, \mathrm{d}t - 1 \right) \\ &= -\int_0^1 (2x'' + \lambda) \delta x \, \mathrm{d}t - \cancel{2x' \delta x \Big|_{t=0}^{t=1}} - \delta\lambda \left(\int_0^1 x \, \mathrm{d}t - 1 \right). \end{aligned}$$

Applying the **fundamental lemma of calculus of variations** yields the following conditions:

$$\begin{cases} 2x'' + \lambda = 0, \\ x(0) = 2, \\ x(1) = 1, \\ \displaystyle\int_0^1 x(t) \, \mathrm{d}t = 1. \end{cases}$$

Solving this boundary value problem gives the same solution as obtained before.

18.4 General Formulation

Consider a variational problem with an objective functional and one or more integral constraints. The problem can be stated as follows:

18.4.1 Problem

Given:

$$\begin{aligned} \text{minimize} \quad & \mathcal{F}(x) = \int_a^b F(x, x', x'', \dots, x^{(n)}, t)\, \mathrm{d}t, \\ \text{subject to} \quad & \mathcal{G}(x) = \int_a^b G(x, x', x'', \dots, x^{(n)}, t)\, \mathrm{d}t = 0. \end{aligned}$$

we want to find functions $x(t)$ that minimize the objective functional $\mathcal{F}(x)$ while satisfying the constraint $\mathcal{G}(x)$.

18.4.2 Augmented Functional

To solve this constrained problem, we introduce a **Lagrange multiplier** λ and form a **Lagrangian functional** $\mathcal{L}(x, \lambda)$:

$$\begin{aligned} \mathcal{L}(x, \lambda) &= \mathcal{F}(x) - \lambda \mathcal{G}(x) \\ &= \int_a^b \left(F(x, x', x'', \dots, x^{(n)}, t) - \lambda G(x, x', x'', \dots, x^{(n)}, t) \right) \mathrm{d}t. \end{aligned}$$

18.4.3 Stationarity

The functions $x(t)$ that are candidates for minimizing $\mathcal{F}(x)$ must be stationary points of the Lagrangian $\mathcal{L}(x, \lambda)$. To find these stationary points, we compute the **first variation** of $\mathcal{L}(x, \lambda)$ with respect to $x(t)$ and λ, and set it to zero. This will lead to a set of Euler-Lagrange equations that must be satisfied by $x(t)$ and λ.

18.4.4 Multiple Constraints

If the problem includes multiple integral constraints, we can generalize the formulation. Let the constraints be given by:

$$\begin{aligned} \text{minimize} \quad & \mathcal{F}(x) = \int_a^b F(x, x', x'', \dots, x^{(n)}, t)\, \mathrm{d}t, \\ \text{subject to} \quad & \mathcal{G}_1(x) = \int_a^b G_1(x, x', x'', \dots, x^{(n)}, t)\, \mathrm{d}t = 0, \\ & \mathcal{G}_2(x) = \int_a^b G_2(x, x', x'', \dots, x^{(n)}, t)\, \mathrm{d}t = 0, \\ & \vdots \\ & \mathcal{G}_p(x) = \int_a^b G_p(x, x', x'', \dots, x^{(n)}, t)\, \mathrm{d}t = 0. \end{aligned}$$

Augmented Functional

To incorporate these constraints, we introduce a Lagrange multiplier λ_k for each constraint $\mathcal{G}_k(x)$, resulting in the Lagrangian functional:

$$\begin{aligned} \mathcal{L}(x, \lambda_1, \lambda_2, \dots, \lambda_p) &= \mathcal{F}(x) - \sum_{k=1}^{p} \lambda_k \mathcal{G}_k(x) \\ &= \int_a^b \left(F(x, x', x'', \dots, x^{(n)}, t) - \sum_{k=1}^{p} \lambda_k G_k(x, x', x'', \dots, x^{(n)}, t) \right) \mathrm{d}t. \end{aligned}$$

Stationarity Conditions

The stationary points of $\mathcal{L}$ with respect to $x(t)$ and λ_k (for $k = 1, 2, \dots, p$) provide candidates for the solution. Setting the first variations of $\mathcal{L}(x, \lambda_1, \dots, \lambda_p)$ to zero leads to a system of equations analogous to the Euler-Lagrange equations for unconstrained problems, now modified to account for the integral constraints.

Constraint Qualification

The multiplier rule requires a constraint qualification: the first variations of the active constraints must be linearly independent. Without it, an abnormal multiplier associated with the objective may be needed. Under the regularity assumption, the Euler–Lagrange equations of the augmented integrand, the original integral constraints, and the boundary conditions form the complete first-order necessary system. They identify stationary candidates, not automatically minimizers.

18.5 Normal and Abnormal Multiplier Rules

For equality-constrained variational problems it is useful to state the multiplier rule in homogeneous form. Introduce a scalar multiplier $\lambda_0 \geq 0$ for the objective and multipliers $\lambda_1, \dots, \lambda_m$ for the constraints, not all zero. The augmented integrand is

$$M = \lambda_0 L + \sum_{k=1}^{m} \lambda_k g_k.$$

If a constraint qualification holds, one may normalize $\lambda_0 = 1$; this is the **normal** case used in the preceding derivations. If the qualification fails, λ_0 may vanish and an **abnormal extremal** is possible. Consequently, writing the objective multiplier implicitly as one is justified only after a regularity assumption has been stated.

19. Path Constraints

We consider a variational problem in which we seek to minimize an objective functional with respect to a **path constraint**—an algebraic constraint that must hold continuously along the entire path of the function $x(t)$.

19.1 Discrete Derivation

Problem Formulation

The problem can be stated as follows:

$$\begin{aligned} \text{minimize} \quad & \mathcal{F}(x) = \int_a^b L(x, x', t)\, \mathrm{d}t, \\ \text{subject to} \quad & g(x, x', t) = 0. \end{aligned}$$

where $L(x, x', t)$ is the Lagrangian, which depends on $x(t)$, its derivative $x'(t)$, and t; and $g(x, x', t) = 0$ represents the algebraic (path) constraint that must be satisfied at each point $t \in [a, b]$.

Feasible Space

We define a feasible space $\mathbb{W}$ for x, which includes all functions that satisfy both the path constraint $g(x, x', t) = 0$ and the boundary conditions of the problem:

$$\mathbb{W} = \left\{ x \in C^1([a, b]) \mid g(x, x', t) = 0, \;\; t \in [a, b] \right\}.$$

The goal, then, is to find an $x \in \mathbb{W}$ that minimizes $\mathcal{F}(x)$.

Feasible Directions

To analyze feasible directions, we introduce a perturbation of the form $x + \alpha \delta x$, where δx is a variation such that:

$$g(x(t), x'(t), t) = 0, \quad \text{for all } t \in [a, b].$$

To obtain the variation of $\mathcal{F}(x)$, we compute:

$$\begin{aligned}\delta\mathcal{F}(x;\delta x) &= \int_a^b \left(\frac{\partial L}{\partial x}(x,x',t)\delta x + \frac{\partial L}{\partial x'}(x,x',t)\delta x'\right) \mathrm{d}t \\ &= \int_a^b \left(\frac{\partial L}{\partial x}(x,x',t) - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x'}(x,x',t)\right)\right) \delta x \,\mathrm{d}t \\ &\quad + \left[\frac{\partial L}{\partial x'}(x,x',t)\delta x\right]_{t=a}^{t=b}.\end{aligned}$$

However, for δx to be feasible, it must also satisfy the constraint:

$$\delta g(x,x',t) = 0, \quad \Rightarrow \quad \frac{\partial g}{\partial x}(x,x',t)\delta x + \frac{\partial g}{\partial x'}(x,x',t)\delta x' = 0.$$

this approach is impractical, thus, we try to use finite difference.

Finite Differences

To discretize the problem, we divide the interval $[a,b]$ into n subintervals of width $h = \frac{b-a}{n}$ and define points $t_k = a + kh$ for $k = 0, 1, \dots, n$. Let x_k approximate the solution at each point t_k, so that $x_k \approx x(t_k)$.

1. **First Derivative Approximation:** We approximate the first derivative $x'(t)$ at each point t_k using a finite difference formula:
$$x'(t_{k+1/2}) = \frac{x(t_{k+1}) - x(t_k)}{h} + \mathcal{O}(h^2)$$
(check using Taylor series)
2. **Midpoint Approximation:** We approximate the midpoint $x(t_{k+1/2})$ using left and right values:
$$x(t_{k+1/2}) = \frac{x(t_{k+1}) + x(t_k)}{2} + \mathcal{O}(h^2)$$
(check using Taylor series)
3. **Applying the Constraint:** We impose the path constraint $g(x,x',t) = 0$ at each (mid-)point $t_{k+1/2}$ by substituting the finite difference approximation:
$$g\left(\frac{x_{k+1}+x_k}{2}, \frac{x_{k+1}-x_k}{h}, \frac{t_{k+1}+t_k}{2}\right) = 0.$$
This transforms the continuous path constraint into a set of discrete constraints, one at each point $t_{k+1/2}$.

Discrete Objective

To approximate the integral $\mathcal{F}(x)$, we use the midpoint rule:

$$\mathcal{F}_h(\boldsymbol{x}) = h\sum_{k=0}^{n-1} L\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right),$$

$$x_{k+1/2} = \frac{x_{k+1}+x_k}{2}, \quad x'_{k+1/2} = \frac{x_{k+1}-x_k}{h}, \quad t_{k+1/2} = \frac{t_{k+1}+t_k}{2},$$

Our goal is now to minimize $\mathcal{F}_h(\boldsymbol{x})$ with respect to $\boldsymbol{x} = (x_0, x_1, \dots, x_n)$, subject to the discrete constraints on $g(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}) = 0$ for each k.

Discrete Multipliers

We introduce Lagrange multipliers λ_k at each point t_k to incorporate the constraints. The discrete Lagrangian becomes:

$$\begin{aligned}\mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda}) &= \mathcal{F}_h(\boldsymbol{x}) - h\sum_{k=0}^{n-1}\lambda_{k+1/2}\, g\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right).\\ &= h\sum_{k=0}^{n-1} L\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right) - h\sum_{k=0}^{n-1}\lambda_{k+1/2}\, g\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right)\\ &= h\sum_{k=0}^{n-1} M\left(x_{k+1/2}, x'_{k+1/2}, \lambda_{k+1/2}, t_{k+1/2}\right)\end{aligned}$$

where

$$M(x,x',\lambda,t) = L(x,x',t) - \lambda g(x,x',t)$$

Nonlinear System

To find stationary points of $\mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})$, we differentiate with respect to each x_k and λ_k, leading to a system of equations:

1. **Derivative with respect to x_k:**
$$\frac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_k} = 0, \qquad k = 0,1,\dots,n$$
2. **Constraint equations for each k:**
$$g\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right) = 0, \qquad k = 0,1,\dots,n-1$$

in details:

$$\begin{aligned}\frac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_0} &= h\frac{\partial}{\partial x_0} M\left(x_{1/2}, x'_{1/2}, \lambda_{1/2}, t_{1/2}\right)\\ &= \frac{h}{2}\frac{\partial M}{\partial x}\left(x_{1/2}, x'_{1/2}, \lambda_{1/2}, t_{1/2}\right) - \frac{\partial M}{\partial x'}\left(x_{1/2}, x'_{1/2}, \lambda_{1/2}, t_{1/2}\right)\\ \frac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_n} &= h\frac{\partial}{\partial x_n} M\left(x_{n-1/2}, x'_{n-1/2}, \lambda_{n-1/2}, t_{n-1/2}\right)\\ &= \frac{h}{2}\frac{\partial M}{\partial x}\left(x_{n-1/2}, x'_{n-1/2}, \lambda_{n-1/2}, t_{n-1/2}\right) + \frac{\partial M}{\partial x'}\left(x_{n-1/2}, x'_{n-1/2}, \lambda_{n-1/2}, t_{n-1/2}\right)\end{aligned}$$

for $k = 1,2,\dots n-1$

$$\begin{aligned}\frac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_k} &= h\frac{\partial}{\partial x_k} M\left(x_{k-1/2}, x'_{k-1/2}, \lambda_{k-1/2}, t_{k-1/2}\right) + h\frac{\partial}{\partial x_k} M\left(x_{k+1/2}, x'_{k+1/2}, \lambda_{k+1/2}, t_{k+1/2}\right)\\ &= \frac{h}{2}\left[\frac{\partial M}{\partial x}\left(x_{k-1/2}, x'_{k-1/2}, \lambda_{k-1/2}, t_{k-1/2}\right) + \frac{\partial M}{\partial x}\left(x_{k+1/2}, x'_{k+1/2}, \lambda_{k+1/2}, t_{k+1/2}\right)\right]\\ &+ \left[\frac{\partial M}{\partial x'}\left(x_{k-1/2}, x'_{k-1/2}, \lambda_{k-1/2}, t_{k-1/2}\right) - \frac{\partial M}{\partial x'}\left(x_{k+1/2}, x'_{k+1/2}, \lambda_{k+1/2}, t_{k+1/2}\right)\right]\end{aligned}$$

consider now the discrete nonlinear system

$$\begin{cases}\dfrac{1}{h}\dfrac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_k} = 0, & k = 1,2,\dots,n-1\\ \dfrac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_0} = 0\\ \dfrac{\partial \mathcal{L}_h(\boldsymbol{x},\boldsymbol{\lambda})}{\partial x_n} = 0\\ g\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right) = 0, & k = 0,1,\dots,n-1\end{cases}$$

As $h \to 0$, this finite difference approximation corresponds to the following approximations

$$\begin{aligned}
\frac{1}{h}\frac{\partial \mathcal{L}_h(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial x_k} &\approx \left[\frac{\partial M(x, x', \lambda, t)}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial M(x, x', \lambda, t)}{\partial x'}\right]_{t=t_k} \\
\frac{\partial \mathcal{L}_h(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial x_0} &\approx -\frac{\partial M}{\partial x'}\left(x(a), x'(a), \lambda(a), a\right) \\
\frac{\partial \mathcal{L}_h(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial x_n} &\approx \frac{\partial M}{\partial x'}\left(x(b), x'(b), \lambda(b), b\right) \\
g\left(x_{k+1/2}, x'_{k+1/2}, t_{k+1/2}\right) &\approx g\left(x, x', t\right)\Big|_{t=t_k}
\end{aligned}$$

thus the finite difference approximation can be viewed as the discretization of the following BVP

$$\begin{cases}
\dfrac{\partial M(x, x', \lambda, t)}{\partial x} - \dfrac{\mathrm{d}}{\mathrm{d}t}\dfrac{\partial M(x, x', \lambda, t)}{\partial x'} = 0 \\
-\dfrac{\partial M}{\partial x'}\left(x(a), x'(a), \lambda(a), a\right) = 0 \\
\dfrac{\partial M}{\partial x'}\left(x(b), x'(b), \lambda(b), b\right) = 0 \\
g\left(x, x', t\right) = 0
\end{cases}$$

19.2 Continuous Multipliers

In this section, we develop an analogy for handling constrained variational problems, where we aim to minimize an objective functional subject to an algebraic (or path) constraint. This technique introduces an augmented functional by incorporating a Lagrange multiplier to enforce the constraint continuously over the interval.

19.2.1 Problem

We start with the following constrained minimization problem:

$$\begin{aligned}
&\text{minimize} \quad \mathcal{F}(x) = \int_a^b L(x, x', t)\,\mathrm{d}t, \\
&\text{subject to} \quad g(x, x', t) = 0.
\end{aligned}$$

Where:

- $\mathcal{F}(x)$ is the objective functional, with $L(x, x', t)$ as the Lagrangian depending on the function $x(t)$, its derivative $x'(t)$, and the independent variable t.
- The path constraint $g(x, x', t) = 0$ must hold continuously over the interval $[a, b]$.

19.2.2 Augmented Functional

To incorporate the constraint into the minimization process, we construct an augmented functional $L(x, \lambda)$ by introducing a Lagrange multiplier function $\lambda(t)$ that enforces the constraint $g(x, x', t) = 0$ at every point in the interval:

$$\mathcal{L}(x, \lambda) = \int_a^b M(x, x', \lambda, t)\,\mathrm{d}t$$

where we define the augmented Lagrangian (or the modified integrand) as:

$$M(x, x', \lambda, t) = L(x, x', t) - \lambda g(x, x', t).$$

19.2.3 First Variation

To find stationary points of $\mathcal{L}(x, \lambda)$, we calculate the first variation $\delta\mathcal{L}(x, \lambda; \delta x, \delta\lambda)$ with respect to x and λ:

$$\delta\mathcal{L}(x, \lambda; \delta x, \delta\lambda) = \int_a^b \frac{\partial M}{\partial x}\delta x + \frac{\partial M}{\partial x'}\delta x' + \frac{\partial M}{\partial \lambda}\delta\lambda \, \mathrm{d}t$$

To further simplify, we integrate by parts on the term involving $\delta x'$:

$$\delta\mathcal{L}(x, \lambda; \delta x, \delta\lambda) = \int_a^b \left[\frac{\partial M}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial M}{\partial x'}\right]\delta x + \frac{\partial M}{\partial \lambda}\delta\lambda \, \mathrm{d}t + \frac{\partial M}{\partial x'}\delta x\Big|_{t=a}^{t=b}$$

19.2.4 Necessary Conditions

For free endpoints, stationarity with respect to arbitrary variations δx and $\delta\lambda$ gives the Euler–Lagrange equation, two natural boundary conditions, and the path constraint:

$$\begin{cases} \dfrac{\partial M}{\partial x}(x, x', \lambda, t) - \dfrac{\mathrm{d}}{\mathrm{d}t}\dfrac{\partial M}{\partial x'}(x, x', \lambda, t) = 0 \\ \dfrac{\partial M}{\partial x'}(x(b), x'(b), \lambda(b), b) = 0 \\ -\dfrac{\partial M}{\partial x'}(x(a), x'(a), \lambda(a), a) = 0 \\ -\dfrac{\partial M}{\partial \lambda}(x, x', \lambda, t) = g(x, x', t) = 0 \end{cases}$$

If an endpoint value is fixed, the corresponding variation vanishes and its natural boundary condition is omitted. The multiplier $\lambda(t)$ is a function because the constraint is imposed at every t.

19.3 Inequality Path Constraints

For a pointwise inequality

$$c(x, x', t) \le 0,$$

introduce a multiplier function $\mu(t)$ and the augmented integrand

$$M(x, x', \mu, t) = L(x, x', t) + \mu(t)c(x, x', t).$$

Under an appropriate constraint qualification, the first-order KKT conditions are

$$\begin{aligned} M_x - \frac{\mathrm{d}}{\mathrm{d}t}M_{x'} &= 0, \\ c(x, x', t) &\le 0, \\ \mu(t) &\ge 0, \\ \mu(t)c(x, x', t) &= 0. \end{aligned}$$

Thus $\mu(t) = 0$ wherever the constraint is inactive and may be positive only on active arcs. Junctions between active and inactive arcs require matching conditions; nonsmooth multipliers may arise when the active set changes.

19.4 Regularity of Path-Constraint Multipliers

The notation $\lambda(t)$ or $\mu(t)$ suggests an ordinary function, but this regularity is not automatic. For smooth equality constraints with constant rank, a sufficiently regular multiplier function is

often obtained locally. For state inequalities and changing active sets, the natural multiplier can be less regular and, in more general formulations, may be a measure. Therefore the pointwise KKT equations above should be read under the stated smoothness and constraint-qualification assumptions.

For a constraint involving x' the augmented Euler–Lagrange equation is valid on smooth arcs. At junctions, one must additionally enforce the appropriate Weierstrass–Erdmann matching conditions for the augmented integrand. These junction conditions are part of the necessary system and cannot in general be recovered by solving the differential equation separately on each arc.

20. Summary

1. **Notation**

$$\left.\frac{\mathrm{d}}{\mathrm{d}\alpha}\right|_{\alpha=0} = \delta$$

2. **Problem: minimize**

$$\int_a^b L(x, x', t)\,\mathrm{d}t \;+\; \Phi(x(a), x(b))$$

3. **Boundary conditions:**

$$x(a) = x_a, \quad x(b) = \text{free}, \quad b_k(x(a), x(b)) = 0, \cdots$$

4. **Integral constraints:**

$$\int_a^b g_k(x, x', t)\,\mathrm{d}t \;+\; \Psi_k(x(a), x(b)) = 0$$

5. **Higher order problem: minimize**

$$\int_a^b L(x, x', x'', \ldots, x^{(n)}, t)\,\mathrm{d}t \;+\; \Phi(x(a), x(b))$$

20.1 Admissible Spaces

- $\mathbb{V} = \left\{ x \in C^m([a,b]), \ldots \right\}$
- $\mathbb{D} = \left\{ \delta x \in C^\infty([a,b]), \delta x(a) = \delta x(b) = 0, \delta x'(a) = \delta x'(b) = 0, \ldots \right\}$

20.1.1 Endpoint Costs

$$
\begin{aligned}
\mathcal{F}(x) &= \int_a^b L(x, x', t)\,\mathrm{d}t + \Phi(x(a), x(b)) \\
\delta\mathcal{F}(x) &= \delta \int_a^b L(x, x', t)\,\mathrm{d}t + \delta\Phi(x(a), x(b), x'(a), x'(b)) \\
&= \int_a^b \delta L(x, x', t)\,\mathrm{d}t + \delta\Phi(x(a), x(b)) \\
&= \int_a^b L_x(x, x', t)\delta x + L_{x'}(x, x', t)\delta x'\,\mathrm{d}t \\
&\quad + \Phi_{x_a}(x(a), x(b))\delta x(a) + \Phi_{x_b}(x(a), x(b))\delta x(b)
\end{aligned}
$$

integration by part

$$
\begin{aligned}
&= \int_a^b \left[L_x(x, x', t) - \frac{\mathrm{d}}{\mathrm{d}t}L_{x'}(x, x', t)\right]\delta x\,\mathrm{d}t \\
&\quad + \left(\Phi_{x_a}(x(a), x(b)) - L_{x'}(x(a), x'(a), a)\right)\delta x(a) \\
&\quad + \left(\Phi_{x_b}(x(a), x(b)) + L_{x'}(x(b), x'(b), b)\right)\delta x(b)
\end{aligned}
$$

20.1.2 Endpoint Constraints

Define

$$
\begin{aligned}
\mathcal{F}(x) &= \int_a^b L(x, x', t)\,\mathrm{d}t - \sum_{k=1}^{p} \lambda_k b_k(x(a), x(b)) \\
\delta\mathcal{F}(x) &= \delta \int_a^b L(x, x', t)\,\mathrm{d}t - \delta \sum_{k=1}^{p} \lambda_k b_k(x(a), x(b)) \\
&= \int_a^b L_x(x, x', t)\delta x + L_{x'}(x, x', t)\delta x'\,\mathrm{d}t - \sum_{k=1}^{p} b_k(x(a), x(b))\delta\lambda_k \\
&\quad - \sum_{k=1}^{p} \lambda_k \left(\frac{\partial b_k}{\partial x_a}(x(a), x(b))\delta x(a) + \frac{\partial b_k}{\partial x_b}(x(a), x(b))\delta x(b)\right)
\end{aligned}
$$

integration by part

$$
\begin{aligned}
&= \int_a^b \left[L_x(x, x', t) - \frac{\mathrm{d}}{\mathrm{d}t}L_{x'}(x, x', t)\right]\delta x\,\mathrm{d}t - \sum_{k=1}^{p} b_k(x(a), x(b))\delta\lambda_k \\
&\quad - \left(\sum_{k=1}^{p} \lambda_k \frac{\partial b_k}{\partial x_a}(x(a), x(b)) + L_{x'}(x(a), x'(a), a)\right)\delta x(a) \\
&\quad - \left(\sum_{k=1}^{p} \lambda_k \frac{\partial b_k}{\partial x_b}(x(a), x(b)) - L_{x'}(x(b), x'(b), b)\right)\delta x(b)
\end{aligned}
$$

20.1.3 Integral Constraints

Define

$$M(x, x', \boldsymbol{\lambda}, t) = L(x, x', t) - \sum_{k=1}^{m} \lambda_k g_k(x, x', t)$$

$$N(x_a, x_b, \boldsymbol{\lambda}) = \Phi(x_a, x_b) - \sum_{k=1}^{m} \lambda_k \Psi_k(x_a, x_b)$$

$$\begin{aligned}
\mathcal{F}(x, \boldsymbol{\lambda}) &= \int_a^b M(x, x', \boldsymbol{\lambda}, t)\,\mathrm{d}t + N(x(a), x(b), \boldsymbol{\lambda}) \\
\delta\mathcal{F}(x, \boldsymbol{\lambda}) &= \delta \int_a^b M(x, x', \boldsymbol{\lambda}, t)\,\mathrm{d}t + \delta N(x(a), x(b), \boldsymbol{\lambda}) \\
&= \int_a^b M_x(x, x', \boldsymbol{\lambda}, t)\delta x + M_{x'}(x, x', \boldsymbol{\lambda}, t)\delta x' + M_{\boldsymbol{\lambda}}(x, x', \boldsymbol{\lambda}, t) \cdot \delta\boldsymbol{\lambda}\,\mathrm{d}t \\
&\quad + N_{x_a}(x(a), x(b), \boldsymbol{\lambda})\delta x(a) + N_{x_b}(x(a), x(b), \boldsymbol{\lambda})\delta x(b) + N_{\boldsymbol{\lambda}} \cdot \delta\boldsymbol{\lambda}
\end{aligned}$$

integration by part

$$\begin{aligned}
&= \int_a^b \left[M_x(x, x', \boldsymbol{\lambda}, t) - \frac{\mathrm{d}}{\mathrm{d}t} M_{x'}(x, x', t) \right] \delta x\,\mathrm{d}t \\
&\quad - \sum_{k=1}^{m} \delta\lambda_k \left[\int_a^b g_k(x, x', t)\,\mathrm{d}t + \Psi_k(x(a), x(b)) \right] \\
&\quad + \Big(N_{x_a}(x(a), x(b), \boldsymbol{\lambda}) - M_{x'}(x(a), x'(a), \boldsymbol{\lambda}, a) \Big) \delta x(a) \\
&\quad + \Big(N_{x_b}(x(a), x(b), \boldsymbol{\lambda}) + M_{x'}(x(b), x'(b), \boldsymbol{\lambda}, b) \Big) \delta x(b)
\end{aligned}$$

20.1.4 Higher-Order Reduction

Transform to optimal control problem

$$\begin{aligned}
&\text{minimize:} && \int_a^b L(x_1, x_2, x_3, \dots, x_n, u, t)\,\mathrm{d}t \\
&\text{subject to:} && \begin{cases} x_1' = x_2 \\ x_2' = x_3 \\ \quad\vdots \\ x_{n-1}' = x_n \\ x_n' = u \end{cases}
\end{aligned}$$

20.2 General Problem

$$\text{minimize:} \quad \int_a^b L(x, x', t)\,\mathrm{d}t \; + \; \Phi(x(a), x(b))$$

$$\text{subject to:} \quad b_k(x(a), x(b)) = 0, \qquad k = 1, 2, \dots, p$$

$$\int_a^b g_k(x, x', t)\,\mathrm{d}t \; + \; \Psi_k(x(a), x(b)) = 0, \qquad k = 1, 2, \dots, m$$

defining

$$M(x, x', \boldsymbol{\lambda}, t) = L(x, x', t) - \sum_{k=1}^{m} \lambda_k g_k(x, x', t)$$

$$N(x_a, x_b, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \Phi(x_a, x_b) - \sum_{k=1}^{m} \lambda_k \Psi_k(x_a, x_b) - \sum_{k=1}^{p} \mu_k b_k(x_a, x_b)$$

the stationary points satisfy the BVP

$$\begin{cases} M_x(x, x', \boldsymbol{\lambda}, t) - \dfrac{\mathrm{d}}{\mathrm{d}t} M_{x'}(x, x', \boldsymbol{\lambda}, t) = 0 & \text{Euler Lagrange} \\ N_{x_a}(x(a), x(b), \boldsymbol{\lambda}, \boldsymbol{\mu}) - M_{x'}(x(a), x'(a), \boldsymbol{\lambda}, a) = 0 & \text{Adjoint BC} \\ N_{x_b}(x(a), x(b), \boldsymbol{\lambda}, \boldsymbol{\mu}) + M_{x'}(x(b), x'(b), \boldsymbol{\lambda}, b) = 0 & \text{Adjoint BC} \\ b_k(x(a), x(b)) = 0, \qquad k = 1, 2, \dots, p \\ \displaystyle\int_a^b g_k(x, x', t)\,\mathrm{d}t + \Psi_k(x(a), x(b)) = 0, \qquad k = 1, 2, \dots, m \end{cases}$$

20.3 Path Constraints

For $g(x, x', t) = 0$ imposed pointwise, the multiplier is a function $\lambda(t)$ and the augmented integrand is $M = L - \lambda(t)g$. For an inequality $c(x, x', t) \le 0$, use $M = L + \mu(t)c$ together with

$$\mu(t) \ge 0, \qquad c(x, x', t) \le 0, \qquad \mu(t)c(x, x', t) = 0.$$

20.4 Stationarity and Minima

The Euler–Lagrange and multiplier equations identify stationary candidates. A local minimizer also satisfies $\delta^2 \mathcal{J}(x; \eta) \ge 0$ for every admissible direction. Joint convexity of the integrand and convexity of the admissible set make stationarity sufficient for global optimality; strict convexity gives uniqueness.

20.5 Classification and Existence

The Euler–Lagrange equations and transversality conditions characterize stationary candidates; they do not by themselves prove minimality or existence. For a fixed-endpoint first-order problem,

$$\delta^2 \mathcal{F}(x; \eta) = \int_a^b \left(L_{xx}\eta^2 + 2L_{xx'}\eta\eta' + L_{x'x'}(\eta')^2 \right) \mathrm{d}t.$$

A weak local minimum must satisfy the Legendre condition $L_{x'x'} \ge 0$. Strong local minima also satisfy the Weierstrass excess inequality. Jacobi's equation and conjugate points refine the second-order test. Existence is normally established separately through the direct method: coercivity, compactness, closedness of the feasible set, and lower semicontinuity.

20.6 Constraint Qualifications

Integral and path constraints require a regularity or constraint-qualification hypothesis for the usual normalized multiplier rule. If it fails, abnormal multipliers may occur. For inequality

constraints, primal feasibility, dual feasibility, complementarity, and stationarity must all be imposed; active-set junctions may require additional matching conditions.

V Optimal Control

21 Introduction 251

21.1 Isoperimetric problem
21.2 Discretized solution
21.3 Continuous interpretation
21.4 Observation
21.5 General formulation from Euler-Lagrange

22 Optimal Control Problems 257

22.1 Bolza form and first-order necessary conditions
22.2 Pontryagin minimum principle
22.3 What the Necessary Conditions Do Not Prove
22.4 Linear Controls and Singular Arcs

23 Transformations of Optimal Control Problems 271

23.1 Reference Problem and Sign Convention
23.2 Lagrange to Mayer Transformation
23.3 Mayer to Lagrange Transformation
23.4 Constant Parameters as States
23.5 Integral Constraints
23.6 Fixed-Interval Time Transformation
23.7 Free Terminal Time
23.8 Free Initial and Final Times
23.9 Path and Mixed Constraints
23.10 Equivalence and Numerical Conditioning

24 Summary of Optimal Control Conditions 277

24.1 Canonical Bolza problem
24.2 Integral-constraint augmentation
24.3 Hamiltonian and endpoint Lagrangian
24.4 Necessary conditions
24.5 Free terminal time
24.6 Singular controls
24.7 Numerical solution viewpoint

A Gaussian Elimination and Echelon Forms 281

A.1 Elimination Process
A.2 Echelon Form
A.3 Equivalence transformations for a linear DAE
A.4 Numerical rank, pivoting, and conditioning

The **optimal control problems** can be seen as a functional minimization, called **target**, subjected to ordinary differential constraints.

This part introduces the variational formulation of optimal control, the transformations that bring a problem into a standard form, a set of worked examples, numerical solution methods, and a summary of the necessary conditions.

Prerequisites, objectives, and expected outcomes

Prerequisites: ODEs, KKT conditions, and the Euler–Lagrange framework.
Objectives: formulate optimal-control problems, derive necessary conditions, transform constraints, and compare indirect and direct numerical methods.
Expected outcomes: the reader can build a Hamiltonian formulation, transcribe an OCP into a nonlinear program, interpret costates and path constraints, and validate a computed trajectory.

Contents

- Introduction
- Optimal Control Problems
- Transformations of Optimal Control Problems
- Numerical Methods for Optimal Control
- Examples of Optimal Control Problems (companion volume)
- Summary of Optimal Control Conditions

21. Introduction

The **optimal control problems** can be seen as a functional minimization, called **target**, subjected to ordinary differential constraints.

21.1 Isoperimetric problem

Let's consider a function

$$y(x) \in [a, b] \mapsto \mathbb{R}$$

where the are known the point $y(a) = y(b) = 0$. If we consider the function as a rope having fixed length l that, by calculus I, can be computed as

$$l = \int_a^b \sqrt{1 + y'(x)^2}\, \mathrm{d}x$$

As a first optimal-control problem, determine the function f that minimizes its integral over the prescribed domain:

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(y) = -\int_a^b y(x)\, \mathrm{d}x \\ &\text{subject to:} && y(a) = 0, \quad y(b) = 0 \\ & && \int_a^b \sqrt{1 + y'(x)^2}\, \mathrm{d}x - l = 0 \end{aligned}$$

21.2 Discretized solution

A way to solve the problem is by discretization, dividing the domain $[a, b]$ in n edges having length $h = \frac{b-a}{n}$ and so $x_k = a + kh$. By founding the discrete solution y_k of the system we can approximate also the continuous solution as

$$y_k \approx y(x_k)$$

Considering the integral approximation

$$\int_{x_{k-1}}^{x_k} y(x)\,\mathrm{d}x \approx h y_{k-\frac{1}{2}} = h\frac{y_k + y_{k-1}}{2}$$

and the derivative

$$y'(x_k) \approx y'_{k-\frac{1}{2}} = \frac{y_k - y_{k-1}}{2}$$

With this being stated the initial isoperimetric problem can be rewritten (with some analytical simplification for the calculus) in the discretized form as

$$\begin{aligned} \text{minimize:} \quad & -\frac{h}{2}\big(y_0 + y_n\big) - h\sum_{k=1}^{n-1} y_k \\ \text{subject to:} \quad & y_0 = 0, \quad y_n = 0 \\ & l - h\sum_{k=1}^{n}\sqrt{1 + \left(\frac{y_k - y_{k-1}}{h}\right)^2} \end{aligned}$$

As here stated, this problems become a constrained minimization one that can so be solved using the Lagrange multiplier method on which the unknowns is the vector of the discretized function $\boldsymbol{y} = (y_1, \dots, y_n)$ and the 3 Lagrange multipliers λ_0, λ_n, λ associated to the constraints:

$$\begin{aligned} \mathcal{L}\big(\boldsymbol{y}, \lambda_0, \lambda_n, \lambda\big) = & -\frac{h}{2}\sum_{k=1}^{n}\big(y_k + y_{k-1}\big) - \lambda_0 y_0 - \lambda_n y_n \\ & - \lambda\left(l - h\sum_{k=1}^{n}\sqrt{1 + \left(\frac{y_k - y_{k-1}}{h}\right)^2}\right) \end{aligned}$$

By calculating the gradient of the lagrangian and setting it to zero, we satisfy the first necessary condition for the solutions that relates to the following non-linear system that can be solved by approximate solutions:

for $k = 0$:

$$-\frac{h}{2} - \lambda_0 + \lambda\frac{\frac{y_1 - y_0}{h}}{\sqrt{1 - \left(\frac{y_1 - y_0}{h}\right)^2}} = 0$$

for $k = 1, 2, \dots, n-1$

$$-h + \frac{\lambda\frac{y_k - y_{k+1}}{h}}{\sqrt{1 + \left(\frac{y_k - y_{k-1}}{h}\right)^2}} - \frac{\lambda\frac{y_{k+1} - y_k}{h}}{\sqrt{1 + \left(\frac{y_{k+1} - y_k}{h}\right)^2}} = 0$$

for $k = n$:

$$-\frac{h}{2} - \lambda_n - \lambda\frac{\frac{y_n - y_0}{h}}{\sqrt{1 + \left(\frac{y_1 - y_0}{h}\right)^2}} = 0$$

multipliers conditions:

$$\begin{cases} y_0 = 0, \quad y_n = 0 \\ l - h\sum_{k=1}^{n}\sqrt{1 + \left(\frac{y_k - y_{k-1}}{h}\right)^2} = 0 \end{cases}$$

21.3 Continuous interpretation

Considering now the limit for $h \to 0$ the discretized systems seems to become continuous; in particular the 4-th conditions become

$$y_0 \to y(a) = 0 \quad \text{and} \quad y_n \to y(b) = 0$$

The first condition (associated to $k = 0$) and the third ($k = n$) instead becomes

$$-\frac{h}{2} - \lambda_0 + \lambda \frac{\frac{y_1 - y_0}{h}}{\sqrt{1 - \left(\frac{y_1 - y_0}{h}\right)^2}} \quad \to \quad -\lambda_0 - \lambda \frac{y'(a)}{\sqrt{1 - y'(a)^2}} = 0$$

$$-\frac{h}{2} - \lambda_n - \lambda \frac{\frac{y_n - y_0}{h}}{\sqrt{1 + \left(\frac{y_1 - y_0}{h}\right)^2}} \quad \to \quad -\lambda(b) - \lambda \frac{y'(b)}{\sqrt{1 - y'(b)^2}} = 0$$

With some analytical manipulation the second lagrange condition can be stated as

$$-1 - \lambda \frac{\mathrm{d}}{\mathrm{d}x} \left. \frac{y'(x)}{\sqrt{1 + y'(x)^2}} \right|_{x = x_k} = 0$$

21.4 Observation

By pushing the limit $h \to 0$ the discretized problem becomes continuous and can relate to the general formulation of minimization of a functional $\mathcal{F}$ subjected to an equality constraints (depending on $\mathcal{G}$):

$$\begin{aligned} \text{minimize:} \quad & \mathcal{F}(y) = -\int_a^b y(x)\,\mathrm{d}x \qquad \Longrightarrow \int_a^b L(y, y', t)\,\mathrm{d}t \\ \text{subject to:} \quad & y(a) = 0, \quad y(b) = 0 \\ & \int_a^b \sqrt{1 + y'(x)^2}\,\mathrm{d}x - l = 0 \quad \Longrightarrow \int_a^b \mathcal{G}(y, y', y)\,\mathrm{d}t = 0 \end{aligned}$$

In the isoperimetric problem we can determine the functions $L(y, y', x) = -y$ whose derivative are

$$\frac{\partial L}{\partial y} = -1 \quad \text{and} \quad \frac{\partial L}{\partial y'} = 0$$

Using so the Euler-Lagrange approach we can see that it fails because

$$\frac{\partial L}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}\frac{\partial L}{\partial y'} = -1 \neq 0$$

Considering instead now the definition of the constraint

$$\mathcal{G}(y, y', x) = \frac{l}{b - a} - \sqrt{1 + y'^2}$$

we can build another lagrangian $\tilde{L}$ that also considers (with a Lagrange multiplier λ) the constraint:

$$\tilde{L}(y, y', x, \lambda) = L(y, y', x) - \lambda \mathcal{G}(y, y', x)$$

Applying the Euler-Lagrange method on this expression it's possible to get the same result achieved with discretization, in fact

$$\frac{\partial \tilde{L}}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}\frac{\partial \tilde{L}}{\partial y'} = -1 - \lambda \frac{y'}{\sqrt{1 + y'^2}}$$

21.5 General formulation from Euler-Lagrange

Considering the optimal control problem as

$$\begin{aligned} \text{minimize:} \quad & \int_a^b L(x, x', t)\,\mathrm{d}t \\ \text{subject to:} \quad & x(a) = x_a, \quad x(b) = x_b \\ & \int_a^b \mathcal{G}(x, x', t)\,\mathrm{d}t = 0 \end{aligned}$$

a way to reach a solution is by constructing the lagrangian $\mathcal{L}$ defined as

$$\tilde{L}(x, x', t, \lambda) = L(x, x', t) - \lambda \mathcal{G}(x, x', t)$$

$$\mathcal{L}(x, \lambda, \lambda_a, \lambda_b) = \int_a^b \tilde{L}(x, x', t, \lambda)\,\mathrm{d}t - \lambda_a x(a) - \lambda_b x(b)$$

At this point the solution of the problem is the stationary *point* (that in reality is a function) x of the lagrangian, and this means setting to zero the Gateaux derivative of $\mathcal{L}$:

$$\begin{aligned}\delta\mathcal{L} = &\int_a^b \left(\frac{\partial \tilde{L}}{\partial x}\delta x + \frac{\partial \tilde{L}}{\partial x'}\delta x' + \frac{\partial \tilde{L}}{\partial \lambda}\delta\lambda\right)\mathrm{d}t \\ &- \delta\lambda_a\big(x(a) - x_a\big) - \lambda_a \delta x(a) - \delta\lambda_b\big(x(b) - x_b\big) - \lambda_b \delta x(b)\end{aligned}$$

Performing the integration by part in order to remove the term $\delta x'$ the derivative so becomes

$$\begin{aligned}\delta\mathcal{L} = &\int_a^b A\,\delta x\,\mathrm{d}x - \delta\lambda_a\big(x(a) - x_a\big) - \delta\lambda_b\big(x(b) - x_b\big) \\ &+ B\delta x(a) + C\delta x(b)\end{aligned}$$

where

$$A = \frac{\partial \tilde{L}}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \tilde{L}}{\partial x'} \quad B = -\lambda_a - \frac{\partial \tilde{L}}{\partial y'}\bigg|_a \quad C = -\lambda_b + \frac{\partial \tilde{L}}{\partial y'}\bigg|_b$$

At this point the overall solution of the minimization problem becomes the following ordinary differential equation system:

$$\begin{cases} \left(\frac{\partial L}{\partial x} \lambda \frac{\partial \mathcal{G}}{\partial x}\right) - \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial L}{\partial x'} - \lambda \frac{\partial \mathcal{G}}{\partial x'}\right) = 0 \\ x(a) = x_a, \quad x(b) = x_b \\ -\lambda_a - \left(\frac{\partial L}{\partial x'}\Big|_a - \lambda \frac{\partial \mathcal{G}}{\partial x'}\Big|_a\right) = 0 \\ -\lambda_b + \left(\frac{\partial L}{\partial x'}\Big|_b - \lambda \frac{\partial \mathcal{G}}{\partial x'}\Big|_b\right) = 0 \end{cases}$$

21.5.1 Ordinary differential equation constraint

Let's consider the general formulation of the isoperimetric problem as

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(x) = \int_a^b L(x, x', t)\,\mathrm{d}t \\ &\text{subject to:} && w\big(x(a), x(b)\big) = 0 \\ & && \int_a^b \mathcal{G}(x, x', t)\,\mathrm{d}t = l \end{aligned}$$

the problem can be solved introducing the function $z(t)$ such that $z' = \mathcal{G}(x, x', t)$: this means that $z(a) = 0$ and $z(b) = l$ and so the problem if transformed, considering also the introduction of the relation $y = x'$, to the form

$$\begin{aligned} &\text{minimize:} && \mathcal{F}(x, y) = \int_a^b L(x, y, t)\,\mathrm{d}t \\ &\text{subject to:} && w\big(x(a), x(b)\big) = 0 \\ & && z(a) = 0, \quad z(b) = l \\ & && x' = y \\ & && z' = \mathcal{G}(x, y, t) \end{aligned}$$

With the problem as here stated we refer to x and z as **states**, while y is the **control**. We can so see that an optimal control problem can be easily transform into a boundary value problem that can be solved using the technique shown on functional minimization.

22. Optimal Control Problems

22.1 Bolza form and first-order necessary conditions

Consider the finite-dimensional optimal control problem in Bolza form

$$\begin{aligned} \underset{\boldsymbol{x},\boldsymbol{u}}{\text{minimize}} \quad & J[\boldsymbol{x},\boldsymbol{u}] = \Phi\big(\boldsymbol{x}(a),\boldsymbol{x}(b)\big) + \int_a^b L\big(\boldsymbol{x}(t),\boldsymbol{u}(t),t\big)\,\mathrm{d}t, \\ \text{subject to} \quad & \boldsymbol{x}'(t) = \boldsymbol{f}\big(\boldsymbol{x}(t),\boldsymbol{u}(t),t\big), \\ & \boldsymbol{b}\big(\boldsymbol{x}(a),\boldsymbol{x}(b)\big) = \boldsymbol{0}, \\ & \boldsymbol{u}(t) \in \mathcal{U} \quad \text{a.e. on } [a,b]. \end{aligned}$$

Here $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state, $\boldsymbol{u}(t) \in \mathbb{R}^m$ is the control, and $\boldsymbol{b} \in \mathbb{R}^q$ represents endpoint constraints. Unless stated otherwise, the presentation below uses the **normal minimization convention**

$$H(\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{u},t) = L(\boldsymbol{x},\boldsymbol{u},t) + \boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},t).$$

22.1.1 Integral constraints

An integral constraint

$$\boldsymbol{\Psi}\big(\boldsymbol{x}(a),\boldsymbol{x}(b)\big) + \int_a^b \boldsymbol{g}(\boldsymbol{x},\boldsymbol{u},t)\,\mathrm{d}t = \boldsymbol{0}$$

can be converted into endpoint constraints by adding an auxiliary state $\boldsymbol{z}$,

$$\boldsymbol{z}' = \boldsymbol{g}(\boldsymbol{x},\boldsymbol{u},t), \qquad \boldsymbol{z}(a) = \boldsymbol{0}, \qquad \boldsymbol{z}(b) + \boldsymbol{\Psi}\big(\boldsymbol{x}(a),\boldsymbol{x}(b)\big) = \boldsymbol{0}.$$

This conversion is exact. In particular, **both endpoint conditions on $\boldsymbol{z}$ are essential**; adding only the differential equation does not enforce the original integral constraint.

22.1.2 Augmented functional and transversality

Define the endpoint Lagrangian

$$B(\boldsymbol{x}_a,\boldsymbol{x}_b,\boldsymbol{\omega}) = \Phi(\boldsymbol{x}_a,\boldsymbol{x}_b) + \boldsymbol{\omega}^{\mathsf{T}}\boldsymbol{b}(\boldsymbol{x}_a,\boldsymbol{x}_b),$$

and the augmented functional

$$\mathcal{J}_a = B\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{\omega}\big) + \int_a^b \left[H(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u}, t) - \boldsymbol{\lambda}^\mathsf{T} \boldsymbol{x}'\right] \mathrm{d}t.$$

Integration by parts of $-\boldsymbol{\lambda}^\mathsf{T} \delta \boldsymbol{x}'$ gives

$$\begin{aligned}\delta \mathcal{J}_a = &\int_a^b \big(\boldsymbol{\lambda}' + H_{\boldsymbol{x}}\big)^\mathsf{T} \delta \boldsymbol{x} \,\mathrm{d}t + \int_a^b H_{\boldsymbol{u}}^\mathsf{T} \delta \boldsymbol{u} \,\mathrm{d}t \\ &+ \int_a^b \big(H_{\boldsymbol{\lambda}} - \boldsymbol{x}'\big)^\mathsf{T} \delta \boldsymbol{\lambda} \,\mathrm{d}t + \boldsymbol{b}^\mathsf{T} \delta \boldsymbol{\omega} \\ &+ \big(B_{\boldsymbol{x}_a} + \boldsymbol{\lambda}(a)\big)^\mathsf{T} \delta \boldsymbol{x}(a) + \big(B_{\boldsymbol{x}_b} - \boldsymbol{\lambda}(b)\big)^\mathsf{T} \delta \boldsymbol{x}(b).\end{aligned}$$

Therefore, for an unconstrained smooth control, a normal extremal satisfies

$$\begin{cases} \boldsymbol{x}' = H_{\boldsymbol{\lambda}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \\ \boldsymbol{\lambda}' = -H_{\boldsymbol{x}}, \\ \boldsymbol{b}\big(\boldsymbol{x}(a), \boldsymbol{x}(b)\big) = \boldsymbol{0}, \\ B_{\boldsymbol{x}_a} + \boldsymbol{\lambda}(a) = \boldsymbol{0}, \\ B_{\boldsymbol{x}_b} - \boldsymbol{\lambda}(b) = \boldsymbol{0}, \\ H_{\boldsymbol{u}} = \boldsymbol{0}. \end{cases} \tag{22.1}$$

The stationarity condition $H_{\boldsymbol{u}} = \boldsymbol{0}$ is only a **first-order necessary condition** for an interior unconstrained control. It does not by itself establish that the Hamiltonian is minimized; second-order conditions or convexity may be needed.

▪ **Example 22.1 — Minimum-Energy Transfer of a Point Mass.** Let's consider the problem of a mass m sliding on a plane (coordinate x) subjected only to an external applied force F, the optimal control problem is

$$\begin{aligned} &\text{minimize:} && \int_0^1 F^2 \,\mathrm{d}t \\ &\text{subject to:} && x' = v, \qquad v' = \frac{F}{m} \\ &&& x(0) = 0 \qquad x(1) = 1 \\ &&& v(0) = 0 \qquad v(1) = 0 \end{aligned}$$

In this case the description of the dynamical system is known from the physical domain, in fact the velocity is the derivative of the position and the acceleration (derivative of velocity) is equal to the force applied divided by the mass; the boundaries conditions are set by the problem.

At this point to solve the problem we have to compute the two function $\mathcal{B}, \mathcal{H}$ defined as

$$\begin{aligned} \mathcal{B} &= \mu_1\big(x(a) - 0\big) + \mu_2\big(x(b) - 1\big) + \mu_3\big(v(a) - 0\big) + \mu_4\big(v(b) - 0\big) \\ \mathcal{H} &= F^2 + \lambda_1 v + \lambda_2 \frac{F}{m} \end{aligned}$$

Following the results of Equation 22.1 the boundary valued problem that has to be solved to find the minimum point is

$$\begin{cases} x' = v \\ v' = F/m \\ \lambda_1' = 0 \\ \lambda_2' = -\lambda_1 \\ x(0) = 0 \qquad x(1) = 1 \\ v(0) = 0 \qquad v(1) = 0 \\ \mu_1 + \lambda_1(0) = 0 \\ \mu_3 + \lambda_2(0) = 0 \\ \mu_2 - \lambda_1(1) = 0 \\ \mu_4 - \lambda_2(1) = 0 \\ 2F + \frac{\lambda_2}{m} = 0 \end{cases}$$

In this case the adjoin boundary conditions are trivially solved (in fact will exists μ_i that will satisfy the equation); considering the last equation in the expression of the velocity the system becomes

$$\begin{cases} x' = v \\ v' = -\frac{\lambda_2}{2m^2} \\ \lambda_1' = 0 \\ \lambda_2' = -\lambda_1 \\ x(0) = 0 \qquad x(1) = 1 \\ v(0) = 0 \qquad v(1) = 0 \end{cases}$$

Since $\lambda_1(t) = c_1$ is constant and $\lambda_2' = -\lambda_1$, we have $\lambda_2(t) = -c_1 t + c_2$. Therefore

$$\begin{aligned} v' &= \frac{c_1 t - c_2}{2m^2} \\ \xrightarrow{\int} \quad v &= \frac{c_1}{4m^2} t^2 - \frac{c_2}{2m^2} t + c_3 \\ \xrightarrow{\int} \quad x &= \frac{c_1}{12m^2} t^3 - \frac{c_2}{4m^2} t^2 + c_3 t + c_4 \end{aligned}$$

Considering the boundary conditions than the integration constant are $c_3 = c_4 = 0$ and the endpoint conditions determine $c_1 = -24m^2$ and $c_2 = -12m^2$, giving

$$\begin{aligned} x(t) &= 3t^2 - 2t^3, \\ v(t) &= 6t - 6t^2, \\ \lambda_1(t) &= -24m^2, \\ \lambda_2(t) &= 24m^2 t - 12m^2. \end{aligned}$$

and so the control history to determine this minimal solution is

$$F = -\frac{\lambda_2}{2m} = 6\big(1 - 2t\big)m$$

▪

22.2 Pontryagin minimum principle

When the control is constrained to an admissible set $\mathcal{U}$, the interior stationarity equation $H_{\boldsymbol{u}} = \mathbf{0}$ must be replaced by the pointwise Hamiltonian minimum condition. Under the usual regularity hypotheses, a normal optimal process admits a nontrivial costate $\boldsymbol{\lambda}$ such that, almost everywhere,

$$\begin{aligned} \boldsymbol{x}' &= H_{\boldsymbol{\lambda}}, \\ \boldsymbol{\lambda}' &= -H_{\boldsymbol{x}}, \\ \boldsymbol{u}^\star(t) &\in \arg\min_{v \in \mathcal{U}} H\big(\boldsymbol{x}^\star(t), \boldsymbol{\lambda}(t), \boldsymbol{v}, t\big), \end{aligned}$$

with the endpoint constraints and transversality conditions in Equation 22.1. Compactness of $\mathcal{U}$ is a common sufficient assumption ensuring that a minimum exists when H is continuous in the control; **convexity of $\mathcal{U}$ is not required by the Pontryagin principle itself**.

If $\boldsymbol{u}^\star(t)$ lies in the interior of $\mathcal{U}$ and H is differentiable with respect to $\boldsymbol{u}$, the minimum condition implies

$$H_{\boldsymbol{u}}\big(\boldsymbol{x}^\star, \boldsymbol{\lambda}, \boldsymbol{u}^\star, t\big) = \mathbf{0}.$$

At a boundary point this equation need not hold. For convex $\mathcal{U}$, the corresponding first-order variational inequality is

$$H_{\boldsymbol{u}}(\boldsymbol{x}^\star, \boldsymbol{\lambda}, \boldsymbol{u}^\star, t)^{\mathsf{T}}(\boldsymbol{v} - \boldsymbol{u}^\star) \geq 0, \qquad \forall \boldsymbol{v} \in \mathcal{U}.$$

> **Note**
>
> The general Pontryagin principle also allows **abnormal extremals**, in which the multiplier of the objective vanishes. The chapters here use the normal scaling. When abnormality is possible, one introduces a scalar multiplier $\lambda_0 \geq 0$ and writes $H = \lambda_0 L + \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{f}$, with $(\lambda_0, \boldsymbol{\lambda})$ not identically zero.

22.2.1 Examples of the Pontryagin minimum principle

▪ **Example 22.2 — Maximum-Travel Problem with Bounded Force.** Let's consider the system in example Example 22.1 where in this case the goal is to maximize the travel of the mass m having a force $F \leq |1|$; this means solving the following optimal control problem

$$\begin{aligned} \text{minimize:} \quad & -x(1) \\ \text{subject to:} \quad & x' = v, \qquad v' = \frac{F}{m} \\ & x(0) = 0 \\ & v(0) = 0 \qquad v(1) = 0 \\ & F \in [-1, 1] \end{aligned}$$

In order to solve this problem it's necessary to determine the Hamiltonian

$$\mathcal{H}(x, v, \lambda_1, \lambda_2, F, t) = \lambda_1 v + \lambda_2 \frac{F}{m}$$

and the endpoint Lagrangian

$$\mathcal{B}(x_a, v_a, x_b, v_b, \mu_1, \mu_2, \mu_3) = -x_b + \mu_1 x_a + \mu_2 v_a + \mu_3 v_b$$

The adjoint equation so becomes

$$\lambda_1' = -\frac{\partial \mathcal{H}}{\partial x} = 0$$

and

$$\lambda_2' = -\frac{\partial \mathcal{H}}{\partial v} = -\lambda_1$$

for the the adjoint boundary conditions only $-1-\lambda_1(1)=0$ is reported (the other equation trivilly solves the cofficients μ_i that gives no information for the final solution). The boundary valued problem so become

$$\begin{cases} x' = v \\ v' = F/m \\ \lambda_1' = 0 \\ \lambda_2' = -\lambda_1 \\ -1-\lambda_1(1)=0 \\ x(0)=v(0)=v(1)=0 \\ F(t) = \underset{\overline{F}\in[-1,1]}{\operatorname{argmin}} \left\{ \lambda_1(t)v + \lambda_2(t)\frac{\overline{F}}{m} \right\} \end{cases}$$

In this case in order to compute the argument minimum associated to the Pontryagin principle we have to consider only the term related to $\overline{F}$, and in particular this means

$$\begin{aligned} F(t) &= \underset{\overline{F}\in[-1,1]}{\operatorname{argmin}} \left\{ \lambda_2(t)\frac{\overline{F}}{m} \right\} \\ &= \begin{cases} +1 & \text{if } \lambda_2(t) < 0 \\ -1 & \text{if } \lambda_2(t) > 0 \\ [-1,1] & \text{if } \lambda_2(t) = 0 \end{cases} \\ &= -\operatorname{sign}\lambda_2(t) \end{aligned}$$

and so the simplified system that determines the solution becomes

$$\begin{cases} x' = v \\ v' = -\dfrac{\operatorname{sign}\lambda_2}{m} \\ \lambda_1' = 0 \\ \lambda_2' = -\lambda_1 \\ \lambda_1(1) = -1 \\ x(0)=v(0)=v(1)=0 \end{cases}$$

By integration it can be seen that λ_1 can be regarded as a constant c_1 that, for the adjoint boundary condition, is equal to -1 and so λ_2 is in the form

$$\lambda_2 = t + c_2$$

From now on the solution becomes analytically complex, however it can be found that $c_2 = -\frac{1}{2}$ determining the solution

$$F(t) = \begin{cases} 1 & t \le \frac{1}{2} \\ -1 & t > \frac{1}{2} \end{cases}$$

▪

▪ **Example 22.3 — A Cautionary Example: Stationarity Is Not Sufficient.** Given the formal optimal control problem

$$\begin{aligned} \text{minimize:} \quad & \mathcal{F}(x,u) = \int_0^2 2x - u^2 \, \mathrm{d}t \\ \text{subject to:} \quad & x' = 1 - 2u \\ & x(0) = 1 \qquad x(2) = 0 \end{aligned}$$

As stated, this problem **does not possess a minimum** over an unconstrained control class such as $L^2(0,2)$. Indeed, integrating the dynamics gives the single mean-value constraint

$$\int_0^2 u(t) \, \mathrm{d}t = \frac{3}{2}.$$

The negative quadratic term $-\int_0^2 u(t)^2 \, \mathrm{d}t$ can then be made arbitrarily negative by adding zero-mean oscillations of increasing L^2 norm while preserving the endpoint condition. Hence $\inf J = -\infty$.

The equation $H_u = 0$ would only identify a stationary extremal; because H is concave in u, it cannot certify a pointwise minimum. This example is therefore useful as a warning: **Pontryagin stationarity is necessary only under the relevant hypotheses and is never, by itself, a sufficiency test**.

To obtain a well-posed minimization problem one may, for example, replace $-u^2$ by $+u^2$ or impose a compact control bound $u \in [u_{\min}, u_{\max}]$.

▪

▪ **Example 22.4 — Integral Constraint and Bounded Control.** To solve the optimal control problem

$$\begin{aligned} \text{minimize:} \quad & \int_0^1 t + x - y^2 \, \mathrm{d}t \\ \text{subject to:} \quad & x' = x + y \qquad y' = x + yu \\ & \int_0^1 \sin y \, \mathrm{d}t = \gamma \\ & x(0) = 1 \qquad y(1) = 0 \\ & u \in [-2, 3] \end{aligned}$$

where necessarily $|\gamma| \le 1$, because $|\sin y| \le 1$ on an interval of length one. The value 3 sometimes appearing in earlier versions of this example makes the problem infeasible. We remove the integral constraint by introducing one auxiliary state:

$$\begin{aligned} \text{minimize:} \quad & \int_0^1 t + x - y^2 \, \mathrm{d}t \\ \text{subject to:} \quad & x' = x + y \\ & y' = x + yu \\ & z' = \sin y \\ & x(0) = 1, \qquad z(0) = 0 \\ & y(1) = 0, \qquad z(1) = \gamma \\ & u \in [-2, 3] \end{aligned}$$

The associated Hamiltonian $\mathcal{H}$ and endpoint Lagrangian $\mathcal{B}$ of the problem are

$$\begin{aligned}\mathcal{H}(x, y, z, \lambda_1, \lambda_2, \lambda_3) &= t + x - y^2 \\ &\quad + \lambda_1(x + y) + \lambda_2(x + yu) + \lambda_3 \sin y \\ \mathcal{B}(x_0, y_1, z_0, z_1, \mu_{1\ldots4}) &= -\mu_1\big(x_0 - 1\big) - \mu_2 y_1 \\ &\quad - \mu_3 z_0 - \mu_4\big(z_1 - \gamma\big)\end{aligned}$$

With that stated the co-equations are computed as

$$\begin{aligned}\lambda_1' &= -\frac{\partial \mathcal{H}}{\partial x} = -1 - \lambda_1 - \lambda_2, \\ \lambda_2' &= -\frac{\partial \mathcal{H}}{\partial y} = 2y - \lambda_1 - \lambda_2 u - \lambda_3 \cos y \\ \lambda_3' &= -\frac{\partial \mathcal{H}}{\partial z} = 0\end{aligned}$$

The unconstrained endpoint components are $x(1)$ and $y(0)$, hence transversality gives $\lambda_1(1) = 0$ and $\lambda_2(0) = 0$. The bounded control is obtained directly from the Hamiltonian minimum condition. The resulting BVP is

$$\begin{cases} x' = x + y, \\ y' = x + yu, \\ z' = \sin y, \\ \lambda_1' = -1 - \lambda_1 - \lambda_2, \\ \lambda_2' = 2y - \lambda_1 - \lambda_2 u - \lambda_3 \cos y, \\ \lambda_3' = 0, \\ x(0) = 1, \qquad y(1) = 0, \\ z(0) = 0, \qquad z(1) = \gamma, \\ \lambda_1(1) = 0, \qquad \lambda_2(0) = 0, \\ u^\star(t) \in \arg\min_{\overline{u}\in[-2,3]} \{\lambda_2(t) y(t)\, \overline{u}\}. \end{cases}$$

Therefore

$$u^\star(t) = \begin{cases} -2, & \lambda_2(t)y(t) > 0, \\ [-2, 3], & \lambda_2(t)y(t) = 0, \\ 3, & \lambda_2(t)y(t) < 0. \end{cases}$$

The set-valued middle case is a candidate singular situation and requires further analysis; it cannot be assigned arbitrarily to one endpoint of the control interval.

▪

▪ **Example 22.5 — Bang-Bang Control with a Mayer Term.** Given the optimal control problem

$$\begin{aligned} \text{minimize:} \quad & \mathcal{F}(x, y, u) = x(0) + \int_0^1 x^2 + y^2 + u \, \mathrm{d}t \\ \text{subject to:} \quad & x' = y - u, \qquad y' = x + u \\ & x(1) = 2 \\ & u(t) \in [-1, 2] \end{aligned}$$

the solution can be obtained by firstly computing the Hamiltonian $\mathcal{H}$ and the endpoint Lagrangian $\mathcal{B}$ of the problems resulting in

$$\begin{aligned}\mathcal{H}\big(x, y, u, \lambda_1, \lambda_2\big) &= x^2 + y^2 + u + \lambda_1\big(y - u\big) + \lambda_2\big(x + u\big)\\ &= x^2 + y^2 + u\big(1 - \lambda_1 + \lambda_2\big) + \lambda_1 y + \lambda_2 x\\ \mathcal{B}\big(x_0, x_1, \mu\big) &= x_0 + \mu\big(x_1 - 2\big)\end{aligned}$$

We can so explicit: the adjoint ordinary differential equations

$$\begin{aligned}\lambda_1' &= -\frac{\partial \mathcal{H}}{\partial x} = -2x - \lambda_2\\ \lambda_2' &= -\frac{\partial \mathcal{H}}{\partial y} = -2y - \lambda_1\end{aligned}$$

the adjoint boundary conditions

$$\begin{aligned}\lambda_1(0) &= -\frac{\partial \mathcal{B}}{\partial x_0} = -1\\ \lambda_1(1) &= \frac{\partial \mathcal{B}}{\partial x_1} = \mu\\ \lambda_2(0) &= -\frac{\partial \mathcal{B}}{\partial y_0} = 0\\ \lambda_2(1) &= \frac{\partial \mathcal{B}}{\partial y_1} = 0\end{aligned}$$

The switching function is

$$\sigma(t) = \frac{\partial \mathcal{H}}{\partial u} = 1 - \lambda_1(t) + \lambda_2(t).$$

Because the control is bounded, $\sigma = 0$ must **not** be imposed identically. The control is obtained from the Hamiltonian minimum condition. With the original differential equations, the BVP becomes

$$\begin{cases} x' = y - u\\ y' = x + u\\ \lambda_1' = -2x - \lambda_2\\ \lambda_2' = -2y - \lambda_1\\ \lambda_1(0) = -1\\ \cancel{\lambda_1(1) = \mu} \qquad \text{: trivially satisfied}\\ \lambda_2(0) = \lambda_2(1) = 0\\ u = \underset{\overline{u}\in[-1,2]}{\operatorname{argmin}} \big\{\, \overline{u}\big(1 - \lambda_1 + \lambda_2\big)\big\} \end{cases}$$

In this case the solution of the Pontryagin minimum principle is reduced to the form

$$u^\star(t) = \begin{cases} -1, & \sigma(t) > 0,\\ [-1, 2], & \sigma(t) = 0,\\ 2, & \sigma(t) < 0. \end{cases}$$

▪

22.3 What the Necessary Conditions Do Not Prove

The Pontryagin conditions characterize extremals; they do not by themselves prove existence, uniqueness, or optimality. Existence typically requires compactness/coercivity and appropriate

continuity and convexity hypotheses. Sufficiency can follow in special convex problems, or from a verification theorem based on the Hamilton–Jacobi–Bellman equation. Second-order conditions are needed to distinguish minima from maxima or saddle extremals when convexity is absent.

The Hamilton–Jacobi–Bellman approach uses the value function $V(\boldsymbol{x}, t)$ and, formally, the PDE

$$-V_t(\boldsymbol{x}, t) = \min_{\boldsymbol{u}\in\mathcal{U}} \left\{ L(\boldsymbol{x}, \boldsymbol{u}, t) + \nabla_{\boldsymbol{x}} V(\boldsymbol{x}, t)^{\mathsf{T}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right\}.$$

Where V is differentiable along an optimal trajectory, the costate is related to the value-function gradient. This provides a useful conceptual bridge between indirect methods based on PMP and dynamic programming.

22.4 Linear Controls and Singular Arcs

In this section, we examine problems in which the Hamiltonian is a linear function of the control variable u. Although linear problems are generally expected to be easier to handle, this is not always the case. The complexity arises from the fact that the necessary condition for minimizing the Hamiltonian with respect to the control along an optimal trajectory does not yield a well-defined expression for synthesizing the optimal control.

Another challenge associated with this class of problems is the occurrence of discontinuities in the control, which can be difficult to identify. Additionally, there may be an infinite number of switching points that can accumulate at a specific time instant. Given that general results and theorems regarding existence and uniqueness are limited in this context, it becomes necessary to introduce additional relationships by manipulating other necessary conditions.

Consequently, problems with a Hamiltonian that is linear in terms of the control u represent an almost independent area of research within optimal control theory. Most theoretical results in this domain are derived using a geometric approach, involving fiber bundles and symmetry groups rather than traditional variational techniques.

To maintain clarity and simplicity in notation, we will focus on problems with a single control variable and fixed endpoints. The formulation of these problems is as follows:

Definition 22.4.1 Minimize the cost functional J subject to the dynamic $\boldsymbol{x}'$ with the control u constrained in $|u| \leq 1$ and the initial state $\boldsymbol{x}(0) = \boldsymbol{x}_0$, where

$$\begin{aligned} J(u) &= \int_0^T f_0(\boldsymbol{x}) + b_0(\boldsymbol{x})u \, \mathrm{d}t \\ \boldsymbol{x}'(t) &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})u \qquad \text{i.e.} \end{aligned} \tag{22.2}$$

$f_0(\boldsymbol{x})$ and $b_0(\boldsymbol{x})$ are scalar functions of the state $\boldsymbol{x}$, u is the scalar control.

To further simplify the computations, we consider the scalar equation of the functional as an extra state variable posing

$$x_0' = f_0(\boldsymbol{x}) + b_0(\boldsymbol{x})u, \qquad x_0(0) = 0,$$

(we have already seen that this is not a loss of generality), but the key point here is that the control appears only linearly in both dynamics and cost functional. We can write the Hamiltonian for this system, namely

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{\lambda}, t) = \sum_{i=0}^{n} f_i(\boldsymbol{x})\lambda_i + u \sum_{i=0}^{n} b_i(\boldsymbol{x})\lambda_i.$$

It is convenient to rename the two addends as

$$A(\boldsymbol{x}, \boldsymbol{\lambda}, t) = \sum_{i=0}^{n} f_i(\boldsymbol{x})\lambda_i,$$
$$B(\boldsymbol{x}, \boldsymbol{\lambda}, t) = \sum_{i=0}^{n} b_i(\boldsymbol{x})\lambda_i,$$

so that the Hamiltonian becomes

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{\lambda}, t) = A(\boldsymbol{x}, \boldsymbol{\lambda}, t) + B(\boldsymbol{x}, \boldsymbol{\lambda}, t)u. \tag{22.3}$$

The adjoint variables of the costate are given by the usual relation

$$\lambda_i' = -\frac{\partial \mathcal{H}}{\partial x_i} = -\sum_{j=0}^{n} \lambda_j \frac{\partial f_j(\boldsymbol{x})}{\partial x_i} - u \sum_{j=0}^{n} \lambda_j \frac{\partial b_j(\boldsymbol{x})}{\partial x_i}.$$

We state now the necessary conditions for problem Equation 22.2.

Theorem 22.4.1 If $u^\star(t)$ is an optimal control and if $\boldsymbol{x}^\star(t)$ is the corresponding optimal trajectory, then there are a costate $\lambda_i^\star(t)$ such that

- for $i = 1, \dots, n$

$$x_i'^\star = f_i(\boldsymbol{x}^\star) + b_i(\boldsymbol{x}^\star)u^\star,$$
$$\lambda_i'^\star = -\sum_{j=0}^{n} \lambda_j^\star \frac{\partial f_j(\boldsymbol{x}^\star)}{\partial x_i^\star} - u^\star \sum_{j=0}^{n} \lambda_j^\star(t) \frac{\partial b_j(\boldsymbol{x}^\star)}{\partial x_i^\star};$$

- for all $t \in [0, T]$ and all $u(t)$ s.t. $|u(t)| \leq 1$, holds

$$A(\boldsymbol{x}^\star, \lambda^\star, t) + u^\star(t)B(\boldsymbol{x}^\star, \lambda^\star, t) \leq A(\boldsymbol{x}^\star, \lambda^\star, t) + u(t)B(\boldsymbol{x}^\star, \lambda^\star, t);$$

- If T is free, then for all $t \in [0, T]$

$$\mathcal{H}^\star(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t) := A(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t) + u^\star(t)B(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t) = 0, \tag{22.4}$$

while, if T is fixed, then

$$\mathcal{H}^\star(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t) := A(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t) + u^\star(t)B(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t) = c, \tag{22.5}$$

for a real constant c.

It follows easily from Equation 22.3 with the PMP, that the optimal control is given by

$$u^\star(t) = -\text{sign}\{B(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t)\} = -\text{sign}\left\{\sum_{i=0}^{n} b_i(\boldsymbol{x}^\star(t))\lambda_i^\star(t)\right\},$$

as long as the function $B(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star, t)$ is not zero. If B becomes zero, then the sign function is not defined. If B is zero only in a point we speak of *bang-bang* controls, if B is identically zero for an entire interval $t \in (t_1, t_2]$, the controls are called **singular**. In this case, the associated trajectory $\boldsymbol{x}(t)$ is called a **singular arc**.

22.4.1 Singular controls

If problem Equation 22.2 results singular, controls, costate and trajectory have the following property: there is at least one half-open interval $(t_1, t_2] \subset [0, T]$ such that

$$\sum_{i=0}^{n} \lambda_i^\star(t) b_i(\boldsymbol{x}^\star(t)) = 0, \qquad t \in (t_1, t_2]. \tag{22.6}$$

As before, the existence of an extremal singular control, does not imply that the **optimal** control is singular: we need additional information as uniqueness to conclude its optimality. The function Equation 22.6 is often called **switching function**.

For a singular arc, vanishing of the switching function and its time derivatives is only part of the analysis. If the control first appears explicitly in the $2r$-th derivative $\sigma^{(2r)}$, a standard higher-order necessary condition for a minimizing singular arc is the generalized Legendre–Clebsch condition

$$(-1)^r \frac{\partial}{\partial u}\left(\frac{\mathrm{d}^{2r}}{\mathrm{d}t^{2r}} H_u\right) \geq 0.$$

Failure of this condition rules out local optimality of the singular extremal.

22.4.2 Test for singular controls

Suppose we are in the free terminal time case, so that Equation 22.4 holds, we test if it is possible to have a singular control as follows. First we need to assume that the switching function is zero in an unknown interval $(t_1, t_2]$, then, because of the free end time hypothesis, for $t \in (t_1, t_2]$,

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{\lambda}, t) = \sum_{i=0}^{n} f_i(\boldsymbol{x})\lambda_i + u \sum_{i=0}^{n} b_i(\boldsymbol{x})\lambda_i = 0$$

$$\Downarrow$$

$$\sum_{i=0}^{n} f_i(\boldsymbol{x})\lambda_i = 0.$$

Moreover, for each $k \in \mathbb{N}$, we have that

$$\frac{\mathrm{d}^k}{\mathrm{d}t^k} \sum_{i=0}^{n} \lambda_i(t) b_i(\boldsymbol{x}) = 0, \quad \begin{cases} k = 1, 2, 3, \dots, \\ t \in (t_1, t_2], \end{cases} \tag{22.7}$$

and similarly,

$$\frac{\mathrm{d}^k}{\mathrm{d}t^k} \sum_{i=0}^{n} \lambda_i f_i(\boldsymbol{x}) = 0, \quad \begin{cases} k = 1, 2, 3, \dots, \\ t \in (t_1, t_2], \end{cases} \tag{22.8}$$

However, the canonical equations (we omit explicit time dependence) are

$$x_i' = f_i + u b_i$$

$$\lambda_i' = -\sum_{j=0}^{n} \lambda_j(t) \frac{\partial f_j}{\partial x_i} - u \sum_{j=0}^{n} \lambda_j \frac{\partial b_j}{\partial x_i}, \quad \begin{cases} i, j = 1, 2, \dots, n \\ k \in \mathbb{N}, \end{cases}$$

Let $k = 1$ in Equation 22.7, then by the chain rule we have

$$\frac{\mathrm{d}}{\mathrm{d}t} \sum_{i=0}^{n} \lambda_i b_i = \sum_{i=0}^{n} \left(\lambda_i' b_i + \lambda_i \sum_{j=0}^{n} \frac{\partial b_j}{\partial x_i} x_j' \right) = 0.$$

Substituting the canonical equations in the previous one, yields

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\left(f_j\lambda_i\frac{\partial b_i}{\partial x_j}-b_i\lambda_j\frac{\partial f_j}{\partial x_i}\right)+u\sum_{i=0}^{n}\sum_{j=0}^{n}\left(\lambda_i b_j\frac{\partial b_i}{\partial x_j}-\lambda_j b_i\frac{\partial b_j}{\partial x_i}\right)=0.$$

We notice that the coefficient of u is zero, and we conclude that

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\lambda_i\left(b_j\frac{\partial f_i}{\partial x_j}-f_j\frac{\partial b_i}{\partial x_j}\right)=0. \tag{22.9}$$

Applying the same argument to Equation 22.8, gives

$$\begin{aligned}
0&=\frac{\mathrm{d}}{\mathrm{d}t}\sum_{i=0}^{n}\lambda_i f_i(\boldsymbol{x})\\
&=\sum_{i=0}^{n}\left(\lambda_i' f_i+\lambda_i\sum_{j=0}^{n}\frac{\partial f_i}{\partial x_j}x_j'\right)\\
&=\sum_{i=0}^{n}\sum_{j=0}^{n}\left(\lambda_i f_i\frac{\partial f_i}{\partial x_j}-\lambda_j f_i\frac{\partial f_j}{\partial x_i}\right)+u\sum_{i=0}^{n}\sum_{j=0}^{n}\left(\lambda_i b_j\frac{\partial f_i}{\partial x_j}-\lambda_j f_i\frac{\partial b_j}{\partial x_i}\right)\\
&=u\sum_{i=0}^{n}\sum_{j=0}^{n}\lambda_i\left(b_j\frac{\partial f_i}{\partial x_j}-f_j\frac{\partial b_i}{\partial x_j}\right)=0.
\end{aligned}$$

This last line implies that either $u=0$ or

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\lambda_i\left(b_j\frac{\partial f_i}{\partial x_j}-f_j\frac{\partial b_i}{\partial x_j}\right)=0.$$

Noticing that the previous line is equal to Equation 22.9, we can not conclude that $u=0$, and this is true for all $k\in\mathbb{N}.\backslash$ In conclusion, a necessary but not sufficient test for an extremal singular control is that in the interval $(t_1,t_2]$ the following relations are satisfied:

$$\left.\begin{aligned}
\sum_{i=0}^{n}b_i\lambda_i&=0\\
\sum_{i=0}^{n}f_i\lambda_i&=0\\
\sum_{i=0}^{n}\sum_{j=0}^{n}\lambda_i\left(b_j\frac{\partial f_i}{\partial x_j}-f_j\frac{\partial b_i}{\partial x_j}\right)&=0
\end{aligned}\right\}$$

We show now how to compute the explicit expression for the singular control, we return back to the equation of the switching function Equation 22.7 and let $k=2,3,\ldots$. It is shown in [2:pag.487] that those derivatives for $k=2,3,\ldots$ require extensive manipulations but have all the following same structure:

$$\sum_{i=0}^{n}\lambda_i\psi_{ki}(\boldsymbol{x})+u\sum_{i=0}^{n}\lambda_i\phi_{ki}(\boldsymbol{x})=0.$$

If $\sum\lambda_i\phi_{ki}(\boldsymbol{x})=0$ we can perform successive derivatives of Equation 22.7, in general there will be a finite value of $k=m$ for which $\sum\lambda_i\phi_{mi}(\boldsymbol{x})\neq 0$ and then the singular control u can be solved:

$$u=-\frac{\displaystyle\sum_{i=0}^{n}\lambda_i\psi_{mi}(\boldsymbol{x})}{\displaystyle\sum_{i=0}^{n}\lambda_i\phi_{mi}(\boldsymbol{x})}.$$

Up to now we have dealt with free final time, if the final time is fixed the we have to take in account the presence of the constant c in the Hamiltonian, but Equation 22.7 and Equation 22.8 still hold and hence also the conclusions.\ In practical problems, we must check if

$$\sum \lambda_i \phi_{ki}(\boldsymbol{x}) = 0$$

for all $k = 1, \dots, m-1$ and

$$\sum \lambda_i \phi_{mi}(\boldsymbol{x}) \neq 0$$

If any of the relations are violated then this represent a violation of the necessary conditions and so singular controls can not occur, if these conditions are all met, then there may be a singular extremal control.

23. Transformations of Optimal Control Prob

23.1 Reference Problem and Sign Convention

Consider the Bolza problem

$$\begin{aligned} \underset{\boldsymbol{x},\boldsymbol{u},\boldsymbol{p}}{\text{minimize}} \quad & J = \Phi\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) + \int_a^b L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t)\,\mathrm{d}t, \\ \text{subject to} \quad & \boldsymbol{x}' = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t), \\ & \boldsymbol{b}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) = \boldsymbol{0}, \\ & \boldsymbol{u}(t) \in \mathcal{U}. \end{aligned}$$

Here $\boldsymbol{x}(t) \in \mathbb{R}^n$, $\boldsymbol{u}(t) \in \mathbb{R}^m$, and $\boldsymbol{p} \in \mathbb{R}^r$ denotes a vector of constant optimization parameters when present.

Throughout these notes we use the normal minimization convention

$$H(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u}, \boldsymbol{p}, t) = L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t) + \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t),$$

and the endpoint Lagrangian

$$B(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}, \boldsymbol{\omega}) = \Phi(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}) + \boldsymbol{\omega}^{\mathsf{T}} \boldsymbol{b}(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}).$$

The augmented functional contains

$$\int_a^b \left(H - \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{x}'\right)\,\mathrm{d}t.$$

Therefore the endpoint transversality conditions are

$$B_{\boldsymbol{x}_a} + \boldsymbol{\lambda}(a) = \boldsymbol{0}, \qquad B_{\boldsymbol{x}_b} - \boldsymbol{\lambda}(b) = \boldsymbol{0}.$$

> **!** Important
>
> These signs depend on the Hamiltonian and augmented-functional convention. Mixing the convention $H = L + \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{f}$ with endpoint signs derived from $H = L - \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{f}$ gives an incorrect BVP.

For a constrained control, Pontryagin's condition is

$$\boldsymbol{u}^\star(t) \in \arg\min_{v \in \mathcal{U}} H\big(\boldsymbol{x}^\star(t), \boldsymbol{\lambda}(t), v, \boldsymbol{p}, t\big).$$

Only when $\boldsymbol{u}^\star$ is interior to $\mathcal{U}$ and H is differentiable in $\boldsymbol{u}$ may this be reduced to $H_{\boldsymbol{u}} = \mathbf{0}$.

23.2 Lagrange to Mayer Transformation

Consider a pure Lagrange objective

$$J = \int_a^b L(\boldsymbol{x}, \boldsymbol{u}, t)\,\mathrm{d}t.$$

Introduce a scalar auxiliary state z:

$$z' = L(\boldsymbol{x}, \boldsymbol{u}, t), \qquad z(a) = 0.$$

Then

$$z(b) = \int_a^b L(\boldsymbol{x}, \boldsymbol{u}, t)\,\mathrm{d}t,$$

so the equivalent Mayer problem is

$$\begin{aligned} \text{minimize} \quad & z(b), \\ \text{subject to} \quad & \boldsymbol{x}' = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \\ & z' = L(\boldsymbol{x}, \boldsymbol{u}, t), \\ & \boldsymbol{b}\big(\boldsymbol{x}(a), \boldsymbol{x}(b)\big) = \mathbf{0}, \\ & z(a) = 0, \\ & \boldsymbol{u}(t) \in \mathcal{U}. \end{aligned}$$

With the augmented state

$$\boldsymbol{w} = (\boldsymbol{x}^\mathsf{T}, z)^\mathsf{T},$$

this can be written as a standard Mayer OCP.

> **Note**
>
> For a Bolza objective, the same construction gives the Mayer cost $\Phi(\boldsymbol{x}(a), \boldsymbol{x}(b)) + z(b)$.

23.3 Mayer to Lagrange Transformation

Consider

$$J = \Phi\big(\boldsymbol{x}(a), \boldsymbol{x}(b)\big).$$

A general Mayer term depends on both endpoints. Consequently, it cannot in general be represented by a running cost depending only on the instantaneous variables $(\boldsymbol{x}(t), \boldsymbol{u}(t), t)$ unless the initial endpoint is fixed or is carried along as an additional constant state.

General construction

Introduce

$$\boldsymbol{p}' = \mathbf{0}, \qquad \boldsymbol{p}(a) = \boldsymbol{x}(a).$$

Thus $\boldsymbol{p}(t) \equiv \boldsymbol{x}(a)$. Define

$$\widetilde{L}(\boldsymbol{p}, \boldsymbol{x}, \boldsymbol{u}, t) = \nabla_{\boldsymbol{x}} \Phi(\boldsymbol{p}, \boldsymbol{x})^{\mathsf{T}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) + \frac{\Phi(\boldsymbol{p}, \boldsymbol{p})}{b - a}.$$

Since $\boldsymbol{p}' = \boldsymbol{0}$,

$$\frac{\mathrm{d}}{\mathrm{d}t} \Phi(\boldsymbol{p}, \boldsymbol{x}(t)) = \nabla_{\boldsymbol{x}} \Phi(\boldsymbol{p}, \boldsymbol{x})^{\mathsf{T}} \boldsymbol{x}'.$$

Hence

$$\begin{aligned} \int_a^b \widetilde{L} \, \mathrm{d}t &= \Phi\big(\boldsymbol{p}, \boldsymbol{x}(b)\big) - \Phi\big(\boldsymbol{p}, \boldsymbol{x}(a)\big) + \Phi(\boldsymbol{p}, \boldsymbol{p}) \\ &= \Phi\big(\boldsymbol{x}(a), \boldsymbol{x}(b)\big). \end{aligned}$$

Therefore an equivalent pure Lagrange formulation is

$$\begin{aligned} \text{minimize} \quad & \int_a^b \widetilde{L}(\boldsymbol{p}, \boldsymbol{x}, \boldsymbol{u}, t) \, \mathrm{d}t, \\ \text{subject to} \quad & \boldsymbol{x}' = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \\ & \boldsymbol{p}' = \boldsymbol{0}, \\ & \boldsymbol{p}(a) - \boldsymbol{x}(a) = \boldsymbol{0}, \\ & \boldsymbol{b}\big(\boldsymbol{x}(a), \boldsymbol{x}(b)\big) = \boldsymbol{0}. \end{aligned}$$

If $\boldsymbol{x}(a)$ is fixed, the state $\boldsymbol{p}$ is unnecessary because its value is already known.

23.4 Constant Parameters as States

Suppose the OCP contains an unknown constant vector $\boldsymbol{p} \in \mathbb{R}^r$. It can be represented as a constant state:

$$\boldsymbol{p}'(t) = \boldsymbol{0}.$$

The transformed dynamics are

$$\begin{aligned} \boldsymbol{x}' &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t), \\ \boldsymbol{p}' &= \boldsymbol{0}. \end{aligned}$$

If $\boldsymbol{p}$ is an optimization variable, its endpoint value must not be fixed. Equivalently, one may keep $\boldsymbol{p}$ as a finite-dimensional parameter. The two formulations lead to the parameter stationarity condition

$$B_{\boldsymbol{p}} + \int_a^b H_{\boldsymbol{p}} \, \mathrm{d}t = \boldsymbol{0}.$$

> **Warning**
>
> A prescribed physical parameter is not an optimization variable. Converting a known constant into a state is usually unnecessary and introduces redundant variables.

23.5 Integral Constraints

Consider the isoperimetric constraint

$$\boldsymbol{\Psi}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) + \int_a^b \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t) \, \mathrm{d}t = \boldsymbol{0}.$$

Introduce $\boldsymbol{z}$ by

$$z' = g(x, u, p, t), \qquad z(a) = 0.$$

Then

$$z(b) = \int_a^b g(x, u, p, t)\,\mathrm{d}t,$$

and the integral constraint becomes

$$z(b) + \Psi\big(x(a), x(b), p\big) = 0.$$

Thus the equivalent augmented OCP contains

$$\begin{aligned} x' &= f(x, u, p, t), \\ z' &= g(x, u, p, t), \\ z(a) &= 0, \\ z(b) + \Psi(x(a), x(b), p) &= 0. \end{aligned}$$

For the special condition

$$\int_a^b g(x, u, t)\,\mathrm{d}t = c,$$

use $z(a) = 0$ and $z(b) = c$.

> **!** Important
>
> Introducing only the differential equation $z' = g$ does **not** enforce the integral constraint. The corresponding terminal condition on $z(b)$ is essential.

23.6 Fixed-Interval Time Transformation

A general interval $t \in [a, b]$ can be mapped to $s \in [0, 1]$ by

$$t = a + (b - a)s.$$

Then

$$\frac{\mathrm{d}x}{\mathrm{d}s} = (b - a) f\big(x, u, a + (b - a)s\big),$$

and

$$\int_a^b L(x, u, t)\,\mathrm{d}t = \int_0^1 (b - a) L\big(x, u, a + (b - a)s\big)\,\mathrm{d}s.$$

This simple scaling is useful for numerical solvers that assume a normalized independent-variable interval.

23.7 Free Terminal Time

Consider

$$\begin{aligned} \underset{T, x, u}{\text{minimize}} \quad & \Phi\big(x(0), x(T), T\big) + \int_0^T L(x, u, t)\,\mathrm{d}t, \\ \text{subject to} \quad & x' = f(x, u, t), \\ & b\big(x(0), x(T), T\big) = 0, \\ & u(t) \in \mathcal{U}, \qquad T > 0. \end{aligned}$$

Time normalization

Set

$$t = sT, \qquad s \in [0, 1],$$

and define

$$\widetilde{\boldsymbol{x}}(s) = \boldsymbol{x}(sT), \qquad \widetilde{\boldsymbol{u}}(s) = \boldsymbol{u}(sT).$$

Then

$$\widetilde{\boldsymbol{x}}' = T\boldsymbol{f}\big(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{u}}, sT\big)$$

and

$$\int_0^T L(\boldsymbol{x}, \boldsymbol{u}, t)\,\mathrm{d}t = \int_0^1 TL\big(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{u}}, sT\big)\,\mathrm{d}s.$$

The normalized problem is therefore

$$\begin{aligned} \text{minimize} \quad & \Phi\big(\widetilde{\boldsymbol{x}}(0), \widetilde{\boldsymbol{x}}(1), T\big) + \int_0^1 TL\big(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{u}}, sT\big)\,\mathrm{d}s, \\ \text{subject to} \quad & \widetilde{\boldsymbol{x}}' = T\boldsymbol{f}\big(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{u}}, sT\big), \\ & \boldsymbol{b}\big(\widetilde{\boldsymbol{x}}(0), \widetilde{\boldsymbol{x}}(1), T\big) = \boldsymbol{0}, \\ & \widetilde{\boldsymbol{u}}(s) \in \mathcal{U}, \qquad T > 0. \end{aligned}$$

The scalar T can either remain an optimization parameter or be represented by a constant state $T' = 0$.

Free-time transversality

Let

$$B(\boldsymbol{x}_0, \boldsymbol{x}_T, T, \boldsymbol{\omega}) = \Phi(\boldsymbol{x}_0, \boldsymbol{x}_T, T) + \boldsymbol{\omega}^{\mathsf{T}}\boldsymbol{b}(\boldsymbol{x}_0, \boldsymbol{x}_T, T).$$

A free terminal time produces the additional condition

$$H\big(\boldsymbol{x}(T), \boldsymbol{\lambda}(T), \boldsymbol{u}(T), T\big) + B_T = 0.$$

If the problem is autonomous and B has no explicit dependence on T, then

$$H(T) = 0.$$

In the normalized formulation, the same information appears as stationarity with respect to the parameter T.

23.8 Free Initial and Final Times

If both endpoints in time are unknown, use

$$t = t_0 + (t_f - t_0)s, \qquad s \in [0, 1].$$

Then

$$\widetilde{\boldsymbol{x}}' = (t_f - t_0)\boldsymbol{f}\big(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{u}}, t_0 + (t_f - t_0)s\big).$$

Both t_0 and t_f become optimization parameters and each has its own stationarity/transversality equation. This case should not be collapsed into a single free-duration parameter when the dynamics or endpoint data depend explicitly on absolute time.

23.9 Path and Mixed Constraints

A path constraint has the form

$$\boldsymbol{c}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \le \boldsymbol{0}.$$

Unlike an integral constraint, it cannot generally be replaced by a finite number of endpoint conditions. It requires pointwise multipliers and complementarity conditions. For a mixed state-control constraint, one works formally with

$$H_a = H + \boldsymbol{\nu}^{\mathsf{T}} \boldsymbol{c},$$

with

$$\boldsymbol{\nu}(t) \ge \boldsymbol{0}, \qquad \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}, t) \le \boldsymbol{0}, \qquad \nu_i c_i = 0.$$

Pure state constraints are more delicate because the associated multipliers may be measures and costates may have jump conditions at entry or exit points. They should be treated separately from ordinary control bounds.

23.10 Equivalence and Numerical Conditioning

The transformations in this chapter preserve the mathematical problem when all auxiliary dynamics and endpoint conditions are included. They do **not** necessarily preserve numerical conditioning. In particular:

- converting every integral to an auxiliary state increases BVP dimension;
- introducing constant parameters as states adds zero dynamics and may worsen conditioning;
- time normalization introduces products with the unknown duration T;
- Mayer and Lagrange formulations can produce different sparsity patterns after discretization.

For analysis, choose the formulation that makes the necessary conditions transparent. For computation, choose the formulation that gives the best-scaled and sparsest nonlinear system.

24. Summary of Optimal Control Conditions

24.1 Canonical Bolza problem

Consider

$$\begin{aligned}
\underset{\boldsymbol{x},\boldsymbol{u},\boldsymbol{p}}{\text{minimize}} \quad & J = \Phi\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) + \int_a^b L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t)\,\mathrm{d}t, \\
\text{subject to} \quad & \boldsymbol{x}' = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t), \\
& \boldsymbol{b}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) = \boldsymbol{0}, \\
& \boldsymbol{\Psi}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) + \int_a^b \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t)\,\mathrm{d}t = \boldsymbol{0}, \\
& \boldsymbol{u}(t) \in \mathcal{U}.
\end{aligned}$$

Here $\boldsymbol{p}$ denotes constant optimization parameters, if present.

24.2 Integral-constraint augmentation

Introduce z by

$$z' = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t), \qquad z(a) = 0.$$

The isoperimetric constraint becomes the endpoint condition

$$\boldsymbol{\Psi}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) + z(b) = \boldsymbol{0}.$$

Thus the integral constraint is not removed merely by introducing z'; its terminal boundary condition is essential.

24.3 Hamiltonian and endpoint Lagrangian

Using the normal minimization convention, define

$$H(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\eta}, \boldsymbol{u}, \boldsymbol{p}, t) = L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t) + \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t) + \boldsymbol{\eta}^{\mathsf{T}} \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t),$$

$$B(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}, \boldsymbol{\omega}, \boldsymbol{\eta}) = \Phi(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}) + \boldsymbol{\omega}^{\mathsf{T}} \boldsymbol{b}(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}) + \boldsymbol{\eta}^{\mathsf{T}} \boldsymbol{\Psi}(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{p}).$$

The augmented functional may be written as

$$\mathcal{J}_a = B + \int_a^b \left[H - \boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{x}' - \boldsymbol{\mu}^{\mathsf{T}}\boldsymbol{z}'\right] \mathrm{d}t + \boldsymbol{\eta}^{\mathsf{T}}\boldsymbol{z}(b).$$

Variation with respect to $\boldsymbol{z}$ gives $\boldsymbol{\mu}' = \mathbf{0}$ and $\boldsymbol{\mu}(t) = \boldsymbol{\eta}$, so the auxiliary costate is constant and can be eliminated.

24.4 Necessary conditions

A normal extremal satisfies

$$\begin{aligned}
\boldsymbol{x}' &= \frac{\partial H}{\partial \boldsymbol{\lambda}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t), \\
\boldsymbol{\lambda}' &= -\frac{\partial H}{\partial \boldsymbol{x}}, \\
\boldsymbol{b}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) &= \mathbf{0}, \\
\boldsymbol{\Psi}\big(\boldsymbol{x}(a), \boldsymbol{x}(b), \boldsymbol{p}\big) + \int_a^b \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, t)\,\mathrm{d}t &= \mathbf{0}, \\
\frac{\partial B}{\partial \boldsymbol{x}_a} + \boldsymbol{\lambda}(a) &= \mathbf{0}, \\
\frac{\partial B}{\partial \boldsymbol{x}_b} - \boldsymbol{\lambda}(b) &= \mathbf{0}, \\
\frac{\partial B}{\partial \boldsymbol{p}} + \int_a^b \frac{\partial H}{\partial \boldsymbol{p}}\,\mathrm{d}t &= \mathbf{0}, \\
\boldsymbol{u}^\star(t) &\in \arg\min_{\boldsymbol{v}\in\mathcal{U}} H\big(\boldsymbol{x}^\star(t), \boldsymbol{\lambda}(t), \boldsymbol{\eta}, \boldsymbol{v}, \boldsymbol{p}, t\big).
\end{aligned}$$

If the optimal control is interior to $\mathcal{U}$ and H is differentiable in $\boldsymbol{u}$, the last condition implies

$$\frac{\partial H}{\partial \boldsymbol{u}} = \mathbf{0}.$$

It should **not** be imposed as an additional equation for a boundary control. For convex $\mathcal{U}$, the equivalent first-order variational inequality is

$$H_{\boldsymbol{u}}(\boldsymbol{u}^\star)^{\mathsf{T}}(\boldsymbol{v} - \boldsymbol{u}^\star) \geq 0, \qquad \forall \boldsymbol{v} \in \mathcal{U}.$$

24.5 Free terminal time

If the terminal time T is free, define

$$B(\boldsymbol{x}_0, \boldsymbol{x}_T, T, \boldsymbol{\omega}) = \Phi(\boldsymbol{x}_0, \boldsymbol{x}_T, T) + \boldsymbol{\omega}^{\mathsf{T}}\boldsymbol{b}(\boldsymbol{x}_0, \boldsymbol{x}_T, T).$$

The additional transversality condition is

$$H(T) + \frac{\partial B}{\partial T} = 0.$$

For an autonomous problem with no explicit T-dependence in B, this reduces to $H(T) = 0$.

24.6 Singular controls

For a scalar control appearing linearly,

$$H = A(\boldsymbol{x}, \boldsymbol{\lambda}, t) + \sigma(\boldsymbol{x}, \boldsymbol{\lambda}, t)u,$$

where $\sigma = H_u$ is the switching function. With $u \in [u_{\min}, u_{\max}]$,

$$u^\star = \begin{cases} u_{\min}, & \sigma > 0, \\ u_{\max}, & \sigma < 0. \end{cases}$$

If $\sigma \equiv 0$ on an interval, the arc is singular. Differentiate σ along the extremal until the control appears explicitly; then verify the appropriate higher-order necessary condition, such as the generalized Legendre–Clebsch condition, before claiming local optimality.

24.7 Numerical solution viewpoint

Indirect methods solve the state–costate boundary-value problem generated by the necessary conditions. Direct methods instead parameterize or collocate the trajectories and solve the resulting sparse nonlinear program. In either case, a numerical result should be checked for dynamic consistency, endpoint and path feasibility, optimality residuals, and stability under mesh refinement. The formulations and diagnostic procedures are developed in Numerical Methods for Optimal Control.

> **Note**
>
> The equations above are **necessary conditions**. Existence, uniqueness, and sufficiency require additional hypotheses. Abnormal extremals require the more general Hamiltonian $H = \lambda_0 L + \boldsymbol{\lambda}^\top \boldsymbol{f}$ with $\lambda_0 \geq 0$ and $(\lambda_0, \boldsymbol{\lambda})$ nontrivial.

A. Gaussian Elimination and Echelon Forms

This appendix collects the linear-algebra machinery used throughout the book, and in particular by the index-reduction algorithms of the DAE part. It is placed outside the main text because it is a prerequisite rather than a result: readers who are comfortable with rank-revealing elimination, echelon forms, and the effect of pivoting on conditioning can skip it and refer back only for notation.

Three facts are used repeatedly in the main text:

- any matrix, square or rectangular, singular or not, can be reduced to echelon form by elementary row operations (Theorem A.1.1);
- the reduction identifies a full-rank leading block and a zero block, which is exactly the separation between differential and algebraic equations in a DAE (Section 8.1);
- the numerical rank, and therefore that separation, depends on the pivoting strategy and on the conditioning of the matrix.

The *Weierstrass canonical form* is quite valuable from a theoretical standpoint. However, it can be computationally intensive since it necessitates the use of several Jordan canonical forms. In the following section, we will present a more numerically efficient algorithm for reducing a Differential Algebraic Equation (DAE) to an Ordinary Differential Equation (ODE). This method leverages the well-known Gaussian elimination technique, enhanced with differentiation to achieve the desired reduction.

A.1 Elimination Process

Before diving into DAE reduction to ODE, we will revisit some fundamental concepts and definitions that are pertinent to this process.

Definition A.1.1 Given a matrix

$$M \in \mathbb{R}^{n \times m}$$

we define elementary operations as follows:

- **Permutation of two rows** (or columns),
- **Linear combinations of rows** (or columns).

These operations can also be viewed as specific forms of matrix multiplication.

Definition A.1.2 A non-singular matrix

$$\boldsymbol{P}_{[i,j]} \in \mathbb{R}^{n\times n}$$

is referred to as a **permutation matrix** (**row** or **column** depend on the application) if it is derived from the identity matrix $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ by swapping column i with column j.

Permutation matrices exhibit several noteworthy properties:

- $\boldsymbol{P}_{[i,j]}^{-1} = \boldsymbol{P}_{[i,j]}^{\mathsf{T}}$: This shows that $\boldsymbol{P}_{[i,j]}$ is an orthogonal matrix.
- $\boldsymbol{P}_{[i,j]}^{2} = \boldsymbol{I}$: This indicates that $\boldsymbol{P}_{[i,j]}$ is an involution matrix.
- $\boldsymbol{P}_{[i,j]}\boldsymbol{M}$: This operation exchanges row i with row j.
- $\boldsymbol{M}\boldsymbol{P}_{[i,j]}$: This operation exchanges column i with column j.

Elimination Example

Consider the permutation matrix

$$\boldsymbol{P}_{[1,2]} \in \mathbb{R}^{3\times 3} \quad \text{and the matrix} \quad \boldsymbol{M} \in \mathbb{R}^{3\times 3}$$

The product $\boldsymbol{P}_{[1,2]}M$ exchanges rows 1 and 2 of matrix $\boldsymbol{M}$:

$$\boldsymbol{P}_{[1,2]}\boldsymbol{M} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} = \begin{bmatrix} 4 & 5 & 6 \\ 1 & 2 & 3 \\ 7 & 8 & 9 \end{bmatrix}$$

Similarly, the product $\boldsymbol{M}\boldsymbol{P}_{[1,2]}$ exchanges columns 1 and 2 of matrix $\boldsymbol{M}$:

$$\boldsymbol{M}\boldsymbol{P}_{[1,2]} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 3 \\ 5 & 4 & 6 \\ 8 & 7 & 9 \end{bmatrix}$$

In a similar manner, we can define matrices that perform linear combinations of rows (or columns). These matrices are classified as **elementary matrices**.

Definition A.1.3 A matrix

$$\boldsymbol{F}_{[i,j]}(\alpha) \in \mathbb{R}^{n\times n}$$

is referred to as a **row (or column) elementary matrix** if it has the following structure:

$$\boldsymbol{F}_{[i,j]}(\alpha) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & & 0 & & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & \alpha & \cdots & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix}$$

where the (i, j) entry equals α.

R An elementary matrix $F_{[i,j]}(\alpha) \in \mathbb{R}^{n\times n}$ is identical to the identity matrix of dimension n in all entries except for the (i, j) position, which is set to α.

Elementary matrices have several key properties:

- The operation $F_{[i,j]}(\alpha)M$ adds row j multiplied by α to row i.
- The operation $MF_{[i,j]}(\alpha)$ adds column i multiplied by α to column j.

Theorem A.1.1 Rank normal form (Gaussian elimination with row and column operations).

Let $M \in \mathbb{R}^{n\times m}$ be a matrix of rank r ($0 \le r \le \min(n, m)$). Then there exist invertible matrices $L \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ such that

$$LMR = \begin{bmatrix} I_r & 0_{r\times(m-r)} \\ 0_{(n-r)\times r} & 0_{(n-r)\times(m-r)} \end{bmatrix}.$$

In particular, by omitting the normalization of the pivots, I_r may be replaced by an invertible diagonal matrix $\Lambda \in \mathbb{R}^{r\times r}$. Empty blocks are understood when $r = 0$, $r = n$, or $r = m$.

Proof. We prove the result constructively, using only **elementary row and column operations**. Each such operation is multiplication by an invertible matrix: row operations act on the **left**, column operations on the **right**.

A.1.1 Step 1: Choose a nonzero pivot

If $M = 0$, then $r = 0$ and the result holds with $L = I_n$ and $R = I_m$. Otherwise, choose an entry $M_{ij} \neq 0$. Exchange row i with row 1 and column j with column 1. The resulting matrix has the block form

$$M_1 = \begin{bmatrix} a & b^\top \\ c & D \end{bmatrix}, \qquad a \neq 0,$$

where a is a scalar, $b \in \mathbb{R}^{m-1}$, $c \in \mathbb{R}^{n-1}$, and $D \in \mathbb{R}^{(n-1)\times(m-1)}$.

A.1.2 Step 2: Eliminate the entries below and to the right of the pivot

For every $i = 2, \dots, n$, perform the row operation

$$R_i \longleftarrow R_i - \frac{(M_1)_{i1}}{a} R_1.$$

This eliminates the entries below a and gives

$$\begin{bmatrix} a & b^\top \\ 0 & D - \dfrac{cb^\top}{a} \end{bmatrix}.$$

Next, for every $j = 2, \dots, m$, perform the column operation

$$C_j \longleftarrow C_j - \frac{b_{j-1}}{a} C_1.$$

We obtain

$$M_2 = \begin{bmatrix} a & 0 \\ 0 & S \end{bmatrix}, \qquad S = D - \frac{cb^\top}{a}.$$

The matrix S is the **Schur complement** of the pivot a. Because the operations are invertible, they preserve rank. Hence

$$\operatorname{rank}(\boldsymbol{M}) = \operatorname{rank}(\boldsymbol{M}_2) = 1 + \operatorname{rank}(\boldsymbol{S}),$$

so $\operatorname{rank}(\boldsymbol{S}) = r - 1$.

A.1.3 Step 3: Repeat on the remaining submatrix

Apply the same procedure to $\boldsymbol{S}$, without modifying the already eliminated first row and column. At each step, a nonzero pivot is isolated and the rank of the remaining submatrix decreases by one.

After **exactly r nonzero pivots**, the remaining submatrix has rank zero and is therefore the zero matrix. We obtain

$$\boldsymbol{L}_0\boldsymbol{M}\boldsymbol{R}_0 = \begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \qquad \boldsymbol{\Lambda} = \operatorname{diag}(a_1, \dots, a_r), \quad a_k \neq 0.$$

A.1.4 Step 4: Normalize the pivots

Multiply the first r rows by $1/a_1, \dots, 1/a_r$, respectively. In matrix form, use the invertible diagonal matrix

$$\boldsymbol{D}_0 = \operatorname{diag}(a_1^{-1}, \dots, a_r^{-1}, \underbrace{1, \dots, 1}_{n-r}).$$

Then, setting $\boldsymbol{L} = \boldsymbol{D}_0\boldsymbol{L}_0$ and $\boldsymbol{R} = \boldsymbol{R}_0$,

$$\boldsymbol{L}\boldsymbol{M}\boldsymbol{R} = \begin{bmatrix} \boldsymbol{I}_r & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}.$$

All the matrices used to construct $\boldsymbol{L}$ and $\boldsymbol{R}$ are invertible (permutations, additions of multiples of rows or columns, and nonzero scalings). Their products are therefore invertible, as required. ∎

To illustrate the algorithm clearly, we provide the following example.

Elimination Example with Row Operations

Consider the matrix $\boldsymbol{M} \in \mathbb{R}^{3\times4}$ with rank rank $\boldsymbol{M} = r = 2$, defined as follows:

$$\boldsymbol{M} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 4 & 2 & 0 \end{bmatrix}.$$

Step 1: Initialization

We begin by setting

$$\boldsymbol{M}^{(1)} = \boldsymbol{M} \quad \text{and} \quad k = 1.$$

We search for the first non-zero element, starting from the position $(k, k) = (1, 1)$ and moving down the first column, then across the rows.

At the first iteration, we find the first non-zero element at position $(p, q) = (2, 2)$. To move this element to the $(1, 1)$ position, we use the following permutation matrices:

- $\boldsymbol{P}_{[1,2]} \in \mathbb{R}^{3\times3}$
- $\boldsymbol{P}_{[2,1]} \in \mathbb{R}^{4\times4}$.

Applying these matrices yields:

$$\begin{aligned}\overline{\boldsymbol{M}}^{(1)} &= \boldsymbol{P}_{[1,2]}\boldsymbol{M}^{(1)}\boldsymbol{P}_{[2,1]} \\ &= \begin{bmatrix}0 & 1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 1\end{bmatrix}\boldsymbol{M}^{(1)}\begin{bmatrix}0 & 1 & 0 & 0\\ 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \\ &= \begin{bmatrix}2 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 4 & 0 & 2 & 0\end{bmatrix}.\end{aligned}$$

Step 2: Zeroing Out

Next, we need to set all non-zero elements in the first row and column to zero. The only non-zero element in the first column is $\overline{M}^{(1)}_{3,1} = 4$. We use the row combination matrix:

$$\boldsymbol{C}_{[3,1]}(-2) = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ -2 & 0 & 1\end{bmatrix} \in \mathbb{R}^{3\times 3}$$

to achieve the goal:

$$\begin{aligned}\boldsymbol{M}^{(2)} &= \boldsymbol{C}_{[3,1]}(-2)\overline{\boldsymbol{M}}^{(1)} \\ &= \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ -2 & 0 & 1\end{bmatrix}\begin{bmatrix}2 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 4 & 0 & 2 & 0\end{bmatrix} \\ &= \begin{bmatrix}2 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 2 & 0\end{bmatrix}.\end{aligned}$$

Step 3: Next Iteration

Now, we set $k = 2$ and search for the first non-zero element in $\boldsymbol{M}^{(2)}$. The first non-zero element is at position $(p, q) = (2, 3)$. Thus, we can use the permutation matrices:

- $\boldsymbol{P}_{[2,2]} = \boldsymbol{I} \in \mathbb{R}^{3\times 3}$
- $\boldsymbol{P}_{[3,1]} \in \mathbb{R}^{4\times 4}$.

We apply these matrices:

$$\begin{aligned}\overline{\boldsymbol{M}}^{(2)} &= \boldsymbol{M}^{(2)}\boldsymbol{P}_{[3,1]} \\ &= \begin{bmatrix}2 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 2 & 0\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \\ &= \begin{bmatrix}2 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 2 & 0 & 0\end{bmatrix}.\end{aligned}$$

Step 4: Final Zeroing

Finally, we apply the row combination matrix:

$$C_{[3,2]}(-2) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}$$

to complete the process:

$$\begin{aligned} M^{(3)} &= C_{[3,2]}(-2)\overline{M}^{(2)} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \end{aligned}$$

The diagonal block is given by:

$$\Lambda \in \mathbb{R}^{2\times 2} = \operatorname{diag}(2, 1).$$

This clearly shows that the rank of matrix M is equal to $r = 2$.

A.2 Echelon Form

A matrix is in **row echelon form** if:

- All nonzero rows are above rows of all zeros.
- Each leading entry (first nonzero number from the left) of a row is strictly to the right of the leading entry of the row above it.
- All entries in a column below a leading entry are zeros.

Example of a matrix in row echelon form:

$$\begin{bmatrix} 1 & 2 & 0 & 3 \\ 0 & 4 & 1 & 5 \\ 0 & 0 & 0 & 6 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Another example of a matrix in row echelon form:

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{bmatrix}$$

A.2.1 Elementary Row Operations

To transform a matrix into echelon form, we use three elementary row operations, each of which corresponds to multiplying on the left by an **elementary matrix**:

1. **Swapping two rows** ⇒ Multiply by a **permutation matrix**.
2. **Multiplying a row by a nonzero scalar** ⇒ Multiply by a **diagonal matrix**.
3. **Adding or subtracting a multiple of one row to another row** ⇒ Multiply by a **lower triangular matrix**.

Each elementary matrix is obtained by applying the corresponding row operation to the identity matrix.

Reduction Procedure

Step 1: Identify the Leftmost Nonzero Column (Pivot Column)

- Scan the columns from left to right to find the first column that contains a nonzero entry.
- The first nonzero number in that column will be our **pivot**.

Step 2: Ensure the Pivot is in the First Row of the Submatrix

- If necessary, swap rows to move a row with a nonzero pivot to the top.
- This corresponds to multiplying by a **permutation matrix** P.

Step 3: Zero Out All Entries Below the Pivot

- For each row below the pivot row, subtract an appropriate multiple of the pivot row to make the entries below the pivot **zero**.
- This operation is equivalent to multiplying by a **lower triangular elementary matrix** L.

Step 4: Normalize the Pivot (Optional but Useful)

- The pivot can be scaled to 1 by dividing the row by the pivot value.
- This step is equivalent to multiplying by a **diagonal matrix** D.

Step 5: Move to the Next Column and Repeat

- Find the next pivot column (to the right of the current pivot).
- Repeat Steps 2-4 for the remaining sub-matrix.
- Continue this process until all pivot positions are correctly set.

At each step, the transformation can be written as:

$$\boldsymbol{L}_k\boldsymbol{D}_k\boldsymbol{P}_k \cdots \boldsymbol{L}_2\boldsymbol{D}_2\boldsymbol{P}_2\boldsymbol{L}_1\boldsymbol{D}_1\boldsymbol{P}_1\boldsymbol{A} = \boldsymbol{U}$$

where $\boldsymbol{U}$ is the echelon form of $\boldsymbol{A}$, and the matrices $\boldsymbol{P}$, $\boldsymbol{D}$, $\boldsymbol{L}$ represent elementary row operations.

A.2.2 Echelon-Form Example

Given the matrix:

$$\boldsymbol{A} = \begin{bmatrix} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 3 & 7 & 8 \end{bmatrix}$$

Step 1: Make the First Pivot 1

Swap row 1 and row 2 using permutation matrix $\boldsymbol{P}$:

$$\boldsymbol{PA} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \\ 3 & 7 & 8 \end{bmatrix}$$

Step 2: Zero Out Below the Pivot

Subtract 2 times row 1 from row 2 and 3 times row 1 from row 3 using lower triangular matrix L_1:

$$L_1PA = \begin{bmatrix} 1 & 3 & 5 \\ 0 & -2 & -4 \\ 0 & -2 & -7 \end{bmatrix}$$

Step 3: Normalize the Second Pivot

Multiply row 2 by $-\frac{1}{2}$ using diagonal matrix D_1:

$$D_1L_1PA = \begin{bmatrix} 1 & 3 & 5 \\ 0 & 1 & 2 \\ 0 & -2 & -7 \end{bmatrix}$$

Step 4: Zero Out Below the New Pivot

Add 2 times row 2 to row 3 using lower triangular matrix L_2:

$$L_2D_1L_1PA = \begin{bmatrix} 1 & 3 & 5 \\ 0 & 1 & 2 \\ 0 & 0 & -3 \end{bmatrix}$$

This matrix is now in **row echelon form**.

A.2.3 Singular-Matrix Example

Consider the singular matrix:

$$A = \begin{bmatrix} 2 & 4 & 6 \\ 1 & 2 & 3 \\ 3 & 6 & 9 \end{bmatrix}$$

Step 1: Normalize the First Pivot

Swap row 1 and row 2:

$$PA = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{bmatrix}$$

Step 2: Zero Out Below the Pivot

Subtract 2 times row 1 from row 2 and 3 times row 1 from row 3:

$$L_1PA = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Since the last two rows are all zeros, the **rank** of the matrix is **1**, meaning:

- The matrix has **linearly dependent** rows.
- The determinant is **zero**, confirming singularity.
- The system $Ax = b$ may have either infinitely many solutions or no solution, depending on b.

Thus, the matrix does not have an inverse.

A.2.4 Reduction Algorithm

To transform a matrix into row echelon form:

1. Identify the leftmost **pivot** column.
2. Swap rows if necessary to bring a nonzero pivot to the top $\boldsymbol{PA}$.
3. Zero out all entries below the pivot $\boldsymbol{L}_1, \boldsymbol{L}_2, \ldots$.
4. Normalize pivots if needed $\boldsymbol{D}_1, \boldsymbol{D}_2, \ldots$.
5. Move to the next pivot column and repeat.
6. Continue until all pivots are set correctly.

Each row operation corresponds to multiplication by an **elementary matrix**, leading to the factorization:

$$\boldsymbol{E}_k \cdots \boldsymbol{E}_2 \boldsymbol{E}_1 \boldsymbol{A} = \boldsymbol{U}$$

where $\boldsymbol{U}$ is in echelon form, and $\boldsymbol{E}_i$ are elementary matrices.

This method is a **systematic** approach to solving linear systems, computing ranks, and checking singularity conditions.

Algorithm. Row Echelon Form Reduction

Input: A matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$
Output: Matrix in row echelon form
$k \leftarrow 1, j \leftarrow 1$ `// Initialize`
while $k \leq m$ ***and*** $j \leq n$ **do**
  $p \leftarrow k$
  while $p \leq m$ ***and*** $A_{pj} = 0$ **do**
    $p \leftarrow p + 1$
  end
  if $p > m$ **then**
    $j \leftarrow j + 1$ `// No pivot in this column, move right`
  else
    if $p \neq k$ **then**
      Swap rows k and p `// Bring pivot to current row`
    end
    for $i \leftarrow k + 1$ **to** m **do**
      $t \leftarrow A_{ij}/A_{kj}$
      $A_{i,\bullet} \leftarrow A_{i,\bullet} - t \cdot A_{k,\bullet}$ `// Eliminate below pivot`
    end
    $k \leftarrow k + 1, j \leftarrow j + 1$ `// Move to next pivot position`
  end
end
return $\boldsymbol{A}$

Figure A.1: **Algorithm:** Row Echelon Form Reduction (with row swaps only). Transform a matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ into row echelon form.

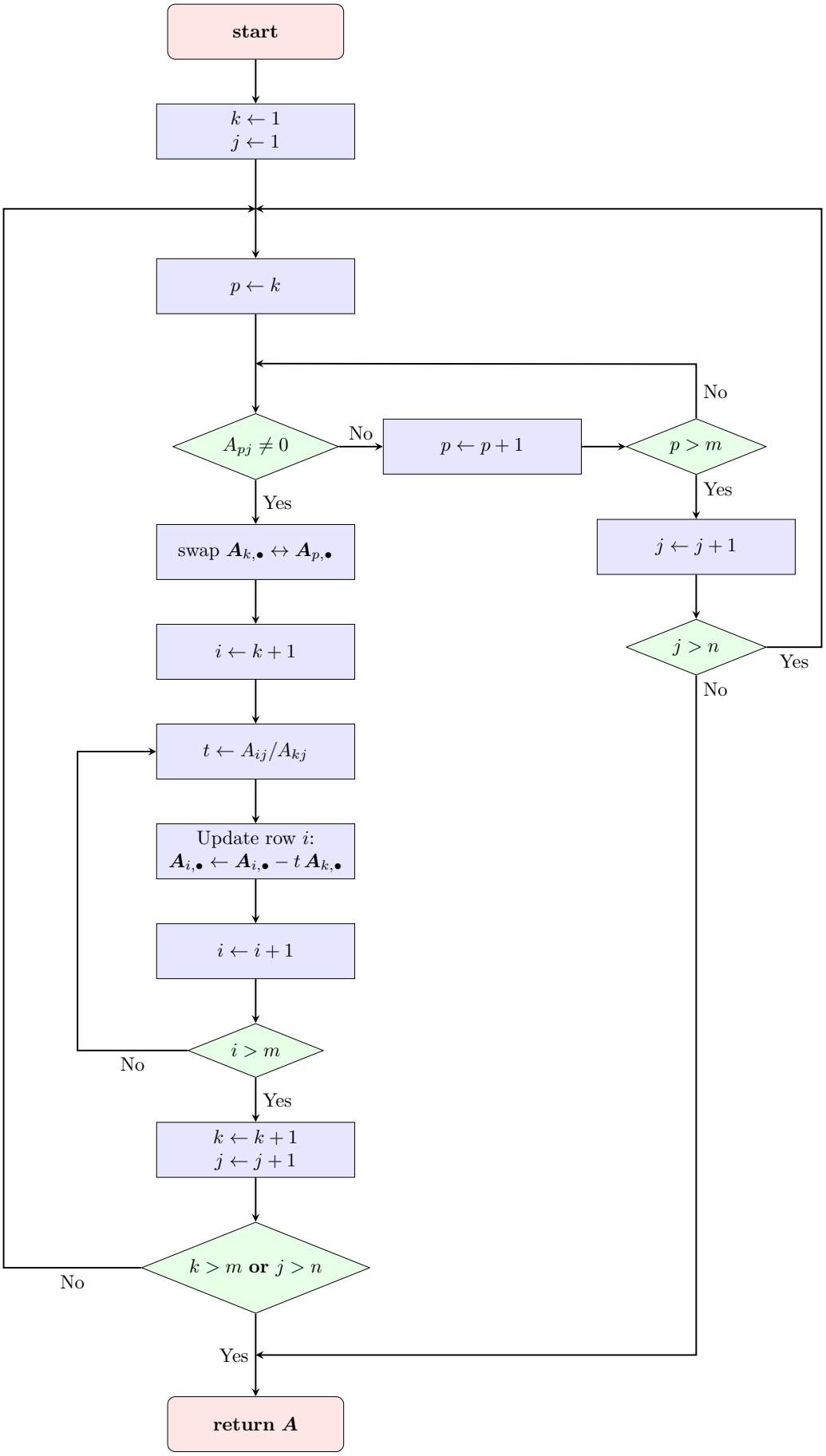


Figure A.2: Algorithm: Reduction to Row Echelon Form (flow chart)

A.3 Equivalence transformations for a linear DAE

Consider $E\dot{x} + Ax = f(t)$. Multiplication by a constant invertible matrix P preserves its solution set:

$$PE\dot{x} + PAx = Pf.$$

Column operations require a **matching change of variables** $x = Qz$, with Q constant and invertible:

$$(PEQ)\dot{z} + (PAQ)z = Pf.$$

These are *strict equivalence transformations* of the pencil. Applying column operations to E without transforming A and the state does not preserve the DAE. If $Q = Q(t)$ depends on time, an additional term $PE\dot{Q}z$ appears.

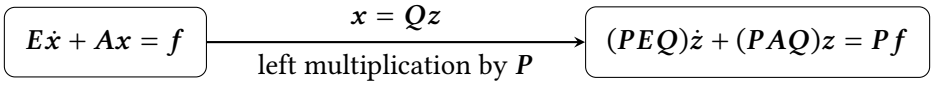


Figure A.3: Row and column operations must transform the complete DAE.

A.4 Numerical rank, pivoting, and conditioning

Gaussian elimination with partial pivoting is preferable to unpivoted elimination for floating-point computations. Rank decisions require a scale-dependent tolerance: a tiny pivot is not a proof of exact singularity. For example,

$$M_\varepsilon = \begin{bmatrix} 1 & 1 \\ 1 & 1+\varepsilon \end{bmatrix}, \qquad \det M_\varepsilon = \varepsilon.$$

For every nonzero ε this matrix has exact rank two, but when $|\varepsilon|$ is comparable with rounding error its numerical rank can be one. Rank-revealing QR with column pivoting or SVD can be more reliable than plain elimination, particularly when estimating the DAE index. In the SVD, singular values below a threshold of the form $\tau \approx \max(m, n)\epsilon_{\text{mach}}\sigma_{\max}$ are often classified as numerically zero; the correct threshold remains problem-dependent.

Important distinction: Gaussian elimination is an algebraic tool; differentiating a constraint changes the regularity requirements on the forcing and must not be confused with an invertible row operation. A differentiated system must retain the original undifferentiated constraints to preserve equivalence.

References

[1] Vladimir Igorevich Arnold. 1989. *Mathematical methods of classical mechanics*. Springer Science & Business Media.

[2] Michael Athans and Peter L. Falb. 1966. *Optimal control : An introduction to the theory and its applications*. McGraw-Hill, New York, Saint Louis, San Francisco.

[3] Mokhtar S. Bazaraa, J. J. Goode, and C. M. Shetty. 1972. Constraint qualifications revisited. *Management Science* 18, 9 (1972), 567–573. https://doi.org/10.1287/mnsc.18.9.567

[4] Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. 2006. *Nonlinear programming: Theory and algorithms* (3rd ed.). Wiley.

[5] Dimitri P. Bertsekas. 2016. *Nonlinear programming* (3rd ed.). Athena Scientific.

[6] J. Frédéric Bonnans and Alexander Shapiro. 2000. *Perturbation analysis of optimization problems*. Springer.

[7] S. P. Boyd and L. Vandenberghe. 2004. *Convex optimization*. Cambridge University Press.

[8] Earl A Coddington and Norman Levinson. 1955. *Theory of ordinary differential equations*. Tata McGraw-Hill Education.

[9] Feliks R Gantmacher. 1964. *The theory of matrices*. Taylor & Francis.

[10] Gene H. Golub and Charles F. Van Loan. 2013. *Matrix computations* (4th ed.). Johns Hopkins University Press.

[11] Joao P Hespanha. 2009. *Linear systems theory*. Princeton university press.

[12] Arieh Iserles. 2009. *A first course in the numerical analysis of differential equations* (2nd ed.). Cambridge University Press. https://doi.org/10.1017/CBO9780511995569

[13] F. John. 1948. Extremum Problems with Inequalities as Subsidiary Conditions. In *Studies and essays: Courant anniversary volume*, K. O. Friedrichs, O. E. Neugebauer and J. J. Stoker (eds.). Wiley-Interscience, New York, 187–204.

[14] H. W. Kuhn and A. W. Tucker. 1951. Nonlinear programming. In *Proceedings of the second berkeley symposium on mathematical statistics and probability*, 1951. University of California Press, Berkeley, California, 481–492.

[15] René Lamour and Dagmar Monett. 2016. A new algorithm for index determination in DAEs using algorithmic differentiation. *Numerical Algorithms* 72, 3 (2016), 667–682. https://doi.org/10.1007/s11075-015-0062-y

[16] Jerrold E Marsden and Tudor Ratiu. 2013. *Introduction to mechanics and symmetry: A basic exposition of classical mechanical systems*. Springer Science & Business Media.

[17] Sven Erik Mattsson and Gustaf Söderlind. 1993. Index reduction in differential-algebraic equations using dummy derivatives. *SIAM Journal on Scientific Computing* 14, 3 (1993), 677–692. https://doi.org/10.1137/0914043

[18] G. McCormick. 1967. Second order conditions for constrained minima. *SIAM Journal on Applied Mathematics* 15, 3 (1967), 641–652. https://doi.org/10.1137/0115056

[19] Jorge Nocedal and Stephen J. Wright. 2006. *Numerical optimization* (2nd ed.). Springer Science & Business Media. https://doi.org/10.1007/978-0-387-40065-5

[20] Linda Petzold. 1982. Differential/algebraic equations are not ODE's. *SIAM Journal on Scientific and Statistical Computing* 3, 3 (1982), 367–384.

[21] R. Tyrrell Rockafellar and Roger J.-B. Wets. 1998. *Variational analysis*. Springer.

[22] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Oriolo. 2009. *Robotics: Modelling, planning and control*. Springer Science & Business Media.

[23] David S. Watkins. 2004. *Fundamentals of matrix computations* (2nd ed.). Wiley.