

# Advanced Control Engineering

Davide Bagnara

August 6, 2025

# Contents

<b>1 Linear System Theory and Design</b>	<b>12</b>
1.1 State Space Representation . . . . .	12
1.1.1 Solution of the Homogeneous State Space Equation in Continuous-time Domain . . . . .	14
1.1.2 Computation of $e^{\tilde{A}t}$ . . . . .	16
1.1.3 Solution of the Non-Homogeneous State Space Equation in Continuous-time Domain . . . . .	18
1.2 Discretization . . . . .	23
1.2.1 Discrete-time Transform Methods . . . . .	30
1.3 Discrete-time Representation of State Space . . . . .	35
1.3.1 Solutions of the Non Homogeneous State Space Equation in Discrete-time Domain . . . . .	37
1.3.2 $\mathcal{Z}$ -Transform Approach to the Solution of Discrete-time State Equations. . . . .	38
1.3.3 Non-Uniqueness of State Space Representation. . . . .	39
1.4 Canonical State Space Representation . . . . .	40
1.4.1 Controllable Canonical Form. . . . .	40
1.4.2 Partial-Fraction-Expansion Programming Method. . . . .	41
1.5 Discretization of Continuous-time State Space System . . . . .	45
1.6 Controllability . . . . .	53
1.7 State Feedback Control . . . . .	55
1.8 Observability . . . . .	59
1.9 Principle of Duality . . . . .	61
1.10 State Observer . . . . .	63
1.11 State Feedback with Integrator . . . . .	76
1.11.1 Continuous-time Domain Case . . . . .	76
1.11.2 Discrete-time Representation of an Integrator . . . . .	78
1.11.3 Discrete-time Domain Case . . . . .	81
<b>2 Ljapunov Stability</b>	<b>89</b>
2.1 Some backgrounds . . . . .	89
2.1.1 Equilibrium Points in Continuous-time Domain . . . . .	92
2.1.2 Equilibrium Points in Discrete-time Domain . . . . .	93
2.2 Fundamentals of Ljapunov Theory. . . . .	94
2.3 Ljapunov Stability. . . . .	102
2.4 First Ljapunov Criterion . . . . .	103
2.4.1 Continuous-time domain . . . . .	103
2.4.2 Discrete-time domain . . . . .	103
2.5 Second Ljapunov Criterion - Direct Method . . . . .	104

2.6 Ljapunov Analysis of Linear Time-Invariant Systems . . . . .	104
<b>3 Optimal Control</b>	<b>108</b>
3.1 Some Background . . . . .	108
3.2 Solution of the General Continuous-time Optimization Problem . . . . .	110
3.2.1 Problem formulation and solution . . . . .	110
3.3 Introduction to Linear Quadratic Optimal Control . . . . .	118
3.4 Steady State Quadratic Optimal Control . . . . .	126
3.4.1 Steady state quadratic optimal control using matlab . . . . .	127
3.5 Optimal Control of the Mass-Spring-Mass System . . . . .	129
3.5.1 Simulation Results . . . . .	132
3.6 Linear Quadratic Integral (LQI) Control . . . . .	136
3.6.1 LQI Control of Mass-Spring-Mass . . . . .	137
<b>4 Least Squares Parameters Estimation for Dynamical Systems</b>	<b>139</b>
4.1 Non-Recursive Method of Least Squares (LS) . . . . .	139
4.2 Recursive Method of Least Squares (LS) . . . . .	143
<b>5 The Kalman Theory</b>	<b>148</b>
5.1 The linear least-squares problem . . . . .	148
5.2 The Kalman-filter problem . . . . .	151
<b>6 Model Predictive Control</b>	<b>157</b>
6.1 Predictive Current Control of a Single-Phase <b>RL</b> Load . . . . .	157
6.1.1 Control Problem . . . . .	157
6.1.2 Prediction of Current Trajectories . . . . .	158
6.1.3 Optimization Problem . . . . .	159
6.1.4 Control Algorithm . . . . .	159
6.2 Theory of Model Predictive Control . . . . .	162
<b>7 Adaptive Control</b>	<b>167</b>
7.1 Model Reference Adaptive Control. . . . .	167
7.2 Simple Adaptive Control Systems . . . . .	168
7.2.1 Direct Adaptive Control . . . . .	168
7.2.1.1 Lyapunov Design . . . . .	168
7.3 Adaptive Control for Active Suspension System . . . . .	175
7.3.1 Adaptive Control Design . . . . .	178
<b>8 Dynamical System Models Derivation</b>	<b>184</b>
8.1 Permanent-magnet Linear Actuator - Moving Coil . . . . .	184
8.2 Permanent Magnet Synchronous Motor - Mathematical Model Derivation . . . . .	190
8.2.1 PMSM model respect to a stationary reference frame . . . . .	193
8.2.2 PMSM model respect to a rotating reference frame . . . . .	196
8.2.3 PMSM Torque Derivation . . . . .	200
8.2.4 Equivalent Circuit of a PMSM . . . . .	203
8.3 Induction Motor Model Derivation . . . . .	204
8.4 Lithium-ion Battery Model . . . . .	214
8.4.1 Principle of Operation . . . . .	214
8.4.2 Dynamical Model of Lithium-ion Battery . . . . .	216
8.4.3 Open Circuit Voltage Model . . . . .	223

---

8.5	LRCL Circuit Modelization . . . . .	224
8.6	Generic Thermal Model . . . . .	228
8.6.1	Preliminary background . . . . .	228
8.6.2	Equivalent mathematical model . . . . .	231
<b>9</b>	<b>Case Studies</b>	<b>236</b>
9.1	Control of a Wheel with C-Code Implementation . . . . .	236
9.1.1	Rotor Speed and Load Torque Observer . . . . .	236
9.1.2	C-Code Implementation Using C-caller . . . . .	241
9.1.3	Simulation results . . . . .	246
9.2	Mass-Spring-Mass Control Problem . . . . .	246
9.2.1	Model Derivation . . . . .	247
9.2.2	Control Design . . . . .	251
9.2.3	Integral control without load estimation . . . . .	253
9.2.3.1	Simulation results . . . . .	256
9.2.4	Integral control with load estimation . . . . .	256
9.2.4.1	Simulation results . . . . .	258
9.3	Control of a Single Phase Inverter . . . . .	259
9.3.1	Model Derivation . . . . .	259
9.3.2	Control Design . . . . .	262
9.3.2.1	Simulation Results . . . . .	264
9.4	Permanent-Magnet Linear Actuator - Moving Coil . . . . .	265
9.4.1	Model Derivation . . . . .	266
9.4.2	Control Problem . . . . .	268
9.4.3	Control Design without Load Estimator . . . . .	268
9.4.3.1	Simulation results . . . . .	270
9.4.4	Control Design with Load Estimator . . . . .	271
9.4.4.1	Simulation Results . . . . .	273
9.5	Pendulum on Cart . . . . .	274
9.5.1	Model Derivation . . . . .	274
9.5.2	Control Implementation . . . . .	278
9.5.3	Control Implementation with load observer . . . . .	280
9.5.4	Simulation Results . . . . .	283
9.6	Segway . . . . .	285
9.6.1	Model Derivation . . . . .	285
9.7	Lithium-ion Battery Management by the use of Extended Kalman Filter . . . . .	289
9.7.1	State Observer Implementation with EKF . . . . .	291
9.7.2	Simulation results . . . . .	294
9.7.3	Adaptive State Observer Implementation with EKF . . . . .	295
9.8	PMSM MPC with EKF . . . . .	295
9.8.1	Introduction . . . . .	295
9.8.2	Control design . . . . .	297
9.8.2.1	Model discretization . . . . .	298
9.8.3	Extended Kalman Filter (EKF) Observer . . . . .	299
9.8.4	EKF Observer for nonlinear PMSM . . . . .	303
9.8.5	Space Vector (SV) and Model Predictive (MP) Modulators . . . . .	305
9.8.6	Current Reference Generator . . . . .	309
9.8.7	MPC with EKF Implementation . . . . .	310
9.8.7.1	MPC algorithm implementation . . . . .	311

---

9.8.8	Simulation Results . . . . .	315
9.9	Induction Motor Control with Kalman Filter Observer . . . . .	316
9.9.1	Rotor Flux Observer . . . . .	320
9.9.2	Torque Control . . . . .	321
9.9.3	Simulation Results . . . . .	323
9.10	Control of an Harmonic in Harsh Environment . . . . .	324
9.10.1	Magnitude control design . . . . .	325
9.10.2	Simulation results . . . . .	327
9.10.3	Magnitude and phase control design . . . . .	329
9.10.4	Simulation results . . . . .	330
9.11	Magnetic Bearing Modelization and Control . . . . .	330
9.11.1	Nomenclature . . . . .	331
9.11.2	Model Derivation . . . . .	331
9.11.3	Model Linearization . . . . .	334
9.11.4	State-Feedback with Integrator applied to the linearized magnetic bearing model. . . . .	339
9.11.5	Simulation Results . . . . .	340
<b>10</b>	<b>Appendices</b>	<b>343</b>
10.1	Moving Reference Frames . . . . .	343
10.2	Propaedeutic Fundamentals for Magnetic Bearings . . . . .	349
10.2.1	Magnetic Field and Induction . . . . .	349
10.2.2	Properties of Ferromagnetic Material . . . . .	353
10.2.3	Magnetic Circuit . . . . .	354
10.2.4	Magnetic Force . . . . .	357
10.2.5	Linearization of the Force-Current Relation . . . . .	358

# List of Figures

1.1	Pendulum.	19
1.2	Phase portrait of the harmonic oscillator for different initial conditions.	21
1.3	Phase portrait of the harmonic damped oscillator.	22
1.4	Pendulum.	22
1.5	Description of a discrete time system.	24
1.6	Sampler and zero-order hold.	25
1.7	(a) System with a sampler and zero order hold; (b) equivalent system (mathematical model) which consists of an impulse sampler and transfer function $(1 - e^{-sT})/s$ .	26
1.8	Right Riemann Integration	29
1.9	Continuous time model.	30
1.10	Mapping of the left half of the s plane into z plane by $s = (1 - z^{-1})/t_s$ (backward difference method).	33
1.11	Area approximation bt the bilinear transformation method (also called Tustin).	34
1.12	Mapping of the left half of the s plane into z plane by $s = 2/t_s(1 - z^{-1})/(1 + z^{-1})$ (bilinear transformation method).	35
1.13	State space representation of a given dynamical system in both continuous and discrete-time domain.	36
1.14	Proportional Resonant Controller structure.	49
1.15	Proportional Resonant Control performance in the case of backward discretization. $i^{ref}(t)$ -dashed, $i(t)$ -solid.	49
1.16	Proportional Resonant Control performance in the case of pre-warped trapezoidal discretization. $i^{ref}(t)$ -dashed, $i(t)$ -solid.	50
1.17	Implementation of the proportional resonant control in the case of discrete time state space representation.	52
1.18	Proportional resonant control performance in the case of discrete time state space implementation. $i^{ref}(t)$ -dashed, $i(t)$ -solid.	53
1.19	Closed loop control system with $u = -\mathbf{K}\vec{x}$ .	56
1.20	State feedback evolution for an generic asymptotic stable system. The effect of the state feedback control is to change the <i>nature</i> of the system. The new system will behave, in closed loop, has designed. The design of the state feedback matrix $\mathbf{K}$ is here called: <i>regulator problem</i> .	57
1.21	Spring mass system with force input $u(t)$ .	57
1.22	Phase portraits of the harmonic oscillator with and without state feedback compensation.	59
1.23	Full-order state observer.	65
1.24	Observed state feedback control system.	69
1.25	Double Integrator Observer.	73
1.26	Double Integrator Observer in Discrete Time Domain.	74

1.27 Phase generation. . . . .	74
1.28 Double Integrator Observer in Discrete Time Domain with bounder block. . . . .	75
1.29 Double integrator observer used as derivative applied to a rotating phase signal. .	75
1.30 Servo system with state feedback and integral control - continuous time case . .	76
1.31 PI-controller architecture in continuous time domain. . . . .	79
1.32 PI-controller architecture in discrete time domain. . . . .	80
1.33 PI-controller architecture in discrete time domain - alternative representation. .	80
1.34 Servo system with state feedback and integral control . . . . .	81
1.35 Servo system with observed state feedback and integral control . . . . .	87
1.36 Servo system with observed state feedback and integral control. SISO case. . . .	87
1.37 Servo system with observed state feedback and integral control. SISO case. . . .	88
2.1 Pendulum. . . . .	89
2.2 Phase portrait of the harmonic oscillator for different initial conditions. . . . .	91
2.3 Phase portrait of the harmonic damped oscillator. . . . .	92
2.4 Ljapunov stability criterion. . . . .	95
2.5 Ljapunov function $V(\vec{x})$ and the Ljapunov level surface $V(\vec{x}) = c$ (dashed). . . .	97
2.6 Plotting results. . . . .	107
3.1 Maximum and minimum of the function $f(x, y) = x + y$ subject to the constraint $\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0$ . . . . .	109
3.2 . . . . .	115
3.3 Informal visualization of the evolution of a state vector in case of autonomous system (dashed) and in the case of optimal state feedback (solid). . . . .	120
3.4 Implementation of the finite time optimal regulator. . . . .	121
3.5 System representation for the case of autonomous system. The symbol $\vec{x}^a(k)$ for the autonomous system is used. . . . .	131
3.6 System representation for the case with optimal state feedback where matrix $\mathbf{K}(k)$ , $k = 0, 1, \dots, N - 1$ has been pre-calculated. . . . .	131
3.7 System representation for the case with steady state optimal state feedback. . . .	132
3.8 Simulation results. . . . .	132
3.9 Simulation results. . . . .	133
3.10 Simulation results. . . . .	133
3.11 Simulation results. . . . .	134
3.12 Simulation results. . . . .	135
3.13 Evolution of the control input $u(k)$ for the case of state feedback. For the au- tonomous, obviously, $u(k) = 0$ . . . . .	135
3.14 LQI controller structure. . . . .	136
3.15 Simulation results. . . . .	138
3.16 Control input $u(k)$ . . . . .	138
4.1 Non-recursive scheme for least squares method. . . . .	139
5.1 The solution to the least squares problem. The matrix $\mathbf{F}$ is $3 \times 2$ ; its two column vectors $\vec{f}_1$ and $\vec{f}_2$ span a two-dimensional plane in a three-dimensional space. The vector $\vec{y}$ does not lie in this plane. The projection of $\vec{y}$ onto the plane spanned by $\vec{f}_1$ and $\vec{f}_2$ is the vector $\hat{\vec{y}} = \mathbf{F}\hat{\vec{x}}$ . The residual $\vec{e} = \vec{y} - \hat{\vec{y}}$ is orthogonal to the plane spanned by $\vec{f}_1$ and $\vec{f}_2$ . . . . .	150
5.2 Illustration of the orthogonality principle. . . . .	153

5.3	Kalman observer implementation. . . . .	156
6.1	H-Bridge inverter with $RL$ load. . . . .	157
6.2	Predictive current control. . . . .	158
6.3	Prediction of the current at the next time step $i(k+1)$ as a function of the output voltage $u(k)$ . The current reference $i^{ref}$ is indicated by the dashed line. . . . .	161
6.4	Simulation results. . . . .	162
6.5	Block diagram for MPC of a linear system (without observer). . . . .	166
7.1	Block diagram of a model-reference adaptive control (MRAC). . . . .	168
7.2	Control architecture. . . . .	173
7.3	Simulation results. . . . .	174
7.4	Simulation results. . . . .	175
7.5	Sky-hook model reference. . . . .	176
7.6	Active suspension model. . . . .	177
7.7	Control architecture for direct MRAC. . . . .	181
7.8	Simulation results. . . . .	182
7.9	Simulation results. . . . .	183
8.1	Moving coil actuator. . . . .	185
8.2	Evaluation of the $d\vec{B}/dt$ . . . . .	186
8.3	Lorentz force applied to the coil charge due to effect of the speed. . . . .	188
8.4	Stationary and rotating reference frame . . . . .	191
8.5	Stationary and rotating reference frame . . . . .	192
8.6	Stationary $\alpha\beta$ reference frame . . . . .	194
8.7	Rotating $dq$ reference frame . . . . .	196
8.8	Rotating and rotor reference frame representation of the pmsm equivalent circuit. . . . .	203
8.9	Machine reference frames description: stationary $(\alpha, \beta)$ , rotor $(d, q)$ and synchronous speed $(x, y)$ reference frames. . . . .	206
8.10	Different point of views regarding different reference frames: stationary $(\alpha, \beta)$ can be considered in still condition; rotor $(d, q)$ can be thought rotating at the rotor speed; synchronous reference frame $(x, y)$ can be thought rotating at electrical stator pulsation. . . . .	207
8.11	Internal view of a typical induction motor with squirrel cage rotor. . . . .	208
8.12	Schematic energy diagram of a lithium cell at open circuit. . . . .	215
8.13	Schematic energy diagram of a lithium cell at open circuit. . . . .	216
8.14	Open circuit voltage as a function of state of charge for different lithium-ion cell chemistries. . . . .	217
8.15	Equivalent circuit of the cell. . . . .	218
8.16	Behavior of the cell polarization, when a cell is subjected to a discharge pulse followed by a rest. . . . .	219
8.17	Circuit that models the diffusion voltage process. . . . .	220
8.18	Measuring parameter values from a pulse response. . . . .	220
8.19	The enhanced self-correcting cell model equivalent circuit. . . . .	223
8.20	Lithium-ion battery open circuit cell as function of the state of charge. . . . .	224
8.21	LRCL electrical schematic. . . . .	224
8.22	Control volume for internal flow in a heat-sink. . . . .	230
8.23	Fundamental thermal circuit used as benchmark. . . . .	230
8.24	Schematic representation of the fundamental thermal circuit. . . . .	231

8.25 Generic thermal structure for a generic electromechanical system. . . . .	232
8.26 Generic thermal equivalent circuit for a generic electromechanical system. . . . .	233
8.27 Generic thermal structure for a cascade of electromechanical systems - the flow of the thermal fluid is created by a hydraulic pump. . . . .	235
9.1 Description of the system. . . . .	236
9.2 Control system implementation with state observer in continuous time domain. .	239
9.3 Control system implementation with state observer in discrete time domain. .	240
9.4 C-caller control architecture. . . . .	241
9.5 Simplified crankshaft model. . . . .	248
9.6 Open loop response of the system applying an impulse to the torque input $\tau_1$ . The natural frequency of the system is $f_0 \approx \frac{1}{2\pi} \sqrt{k_\theta \frac{J_1+J_2}{J_1 J_2}} = 1.424 Hz$ . . . . .	252
9.7 A generic representation of the Proportional Integrator + State Feedback control architecture. This structure can be used when inner properties of the plant need to be modified. . . . .	253
9.8 Description of the control strategy in discrete-time domain without $\tau_2(t)$ estimator.	255
9.9 Simulation results. . . . .	256
9.10 Description of the control strategy with integration and compensation of the load.	257
9.11 Step response of the state feedback control with observer and load compensation. $\tau_2$ is the external load. . . . .	258
9.12 Observer performance. . . . .	259
9.13 Single phase inverter circuit with load disturbance. . . . .	260
9.14 Equivalent model of the single phase inverter with load disturbance. . . . .	260
9.15 Fluid level control in a tank. . . . .	262
9.16 Control architecture for harmonic compensator. . . . .	263
9.17 Simulation results. . . . .	265
9.18 Performance estimation of the disturbance $d(t)$ . . . . .	265
9.19 Linear actuator. . . . .	266
9.20 Control architecture. . . . .	269
9.21 Simulation results. . . . .	270
9.22 Simulation results. . . . .	271
9.23 Control architecture with load estimator. . . . .	272
9.24 Simulation results. . . . .	273
9.25 Pendulum on cart. . . . .	275
9.26 Control architecture for <i>pendulum on cart</i> problem - discrete time domain. .	279
9.27 Control architecture for <i>pendulum on cart</i> problem with load estimator - discrete time domain. . . . .	282
9.28 Simulation results for the case with no load estimation. . . . .	283
9.29 Simulation results for the case with load estimation and compensation. . . . .	284
9.30 Segway. . . . .	286
9.31 The enhanced self-correcting cell model equivalent circuit. . . . .	291
9.32 Extended Kalman Filter algorithm for the lithium ion battery. . . . .	294
9.33 Lithium-ion battery manager - simulation results. . . . .	294
9.34 Lithium-ion battery manager - simulation results. . . . .	295
9.35 PMSM control architecture. . . . .	298
9.36 Kalman observer with time varying matrix transition. . . . .	303
9.37 Representation of the space vector modulation and relation with inverter switches state. . . . .	306

9.38 Space vector representation of a generic voltage vector $\vec{u}_{\alpha\beta}$ .	306
9.39 Inverter state representation of the voltages $v_1$ and $v_2$ .	307
9.40 Inverter state representation of the voltages $v_3$ and $v_4$ .	307
9.41 Inverter state representation of the voltages $v_5$ and $v_6$ .	308
9.42 Inverter state representations of the voltage $v_0$ .	308
9.43 PMSM control architecture.	315
9.44 Simulation results.	316
9.45 Simulation results.	316
9.46 Induction motor torque control architecture with flux observer.	317
9.47 Kalman observer.	321
9.48 Simulation results.	324
9.49 Harmonic magnitude extraction block.	325
9.50 Global control layout.	327
9.51 Simulation results.	328
9.52 Simulation results.	328
9.53 Global control layout.	329
9.54 Simulation results.	330
9.55 Simulation results.	330
9.56 Two poles pairs bearing magnets.	332
9.57 Functional graph for the implementation of a control based on the linearized model.	339
9.58 Magnetic bearing control architecture.	340
9.59 Simulation results - case study of linear control based on the linearized model of the magnetic bearing.	341
9.60 Simulation results - case study of linear control based on the linearized model of the magnetic bearing.	342
10.1 Inertial reference frames in relative motion along one axis.	343
10.2 Inertial reference frames in relative motion.	345
10.3 Basic magnetic bearing structure, showing the fundamental components.	349
10.4 Lorentz force.	350
10.5 Magnetic fields around conductors.	351
10.6 Induction law in a turn..	352
10.7 $B - H$ diagram, hysteresis loop.	353
10.8 Magnetic circuit.	355
10.9 Graphical determination of the magnetization.	356
10.10 Force and geometry of a magnetic bearing portion of circuit.	357
10.11 Differential driving mode of the bearing magnets.	359

# List of Tables

6.1	Set of admissible output voltage $\mathcal{U}(k)$ that meets the switching constraint $ \Delta u(k)  \leq 1.160$	160
6.2	Cost corresponding to the three admissible $u(k)$ .	161
9.1	Flexible shaft model parameters.	251
9.2	Components data of the filter and load.	264
9.3	Set of admissible output voltage $\mathcal{U}(k)$ from a given voltage vector $(u_\alpha \ u_\beta)$ .	309
9.4	IM model parameters.	323
9.5	Magnetic bearing data.	341

# Chapter 1

## Linear System Theory and Design

### 1.1 State Space Representation

Dynamical systems are in general represented by  $n$ -order differential equations. The most famous example of mathematical representation of a dynamic system can be found into the principles of mass movement derived by Sir Newton, e.g.

$$\begin{cases} \ddot{x}(t) = 0 \\ x(0) = x_0 & \text{initial condition of the position} \\ \dot{x}(0) = v_0 & \text{initial condition of the speed} \end{cases} \quad (1.1.1)$$

From mathematical results we know that a  $n$ -order differential equation can be represented by a set of  $n$  first-order differential equations. That means the dynamic system represented by Eq. (1.1.1) can be modelized by the following set of equations

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = 0 \\ x_1(0) = x_0 & \text{initial condition of the first state variable} \\ x_2(0) = v_0 & \text{initial condition of the second state variable} \end{cases} \quad (1.1.2)$$

Eq. (1.1.2) represents the state space representation of a dynamical system.

The model represented in Eq. (1.1.2) can be rewritten using the matrix notation. The use of the matrix notation permits to use the set of rules and results achieved in the branch of *matrix analysis* and helps to achieve additional results that the  $n$ -order differential equation representation could hide. Let's see how the system of Eq. (1.1.2) can be represented in matrix

form, that gives

$$\begin{cases} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \\ \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} x_0 \\ v_0 \end{bmatrix} \end{cases} \quad (1.1.3)$$

or in a more compact way

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) \\ \vec{x}(0) = \vec{x}_0 \end{cases} \quad (1.1.4)$$

where  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$ ,  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ . The system  $\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t)$  is also called *autonomous* and represents the set of equation which describes the evolution of the system when it starts from the initial condition  $\vec{x}(0) = \vec{x}_0$  and without any perturbations or inputs.

The properties of the system  $\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t)$ , in term of stability as well as instability, can be derived analysing the eigenvalues of the matrix  $\tilde{\mathbf{A}}$ . Additional methods, like *Ljapunov's stability* and corollaries, exist and they permit to extend and generalize the concept of *stability* and *asymptotic stability*. These theorems will be encountered in the next lectures.

Coming back to our transition matrix  $\tilde{\mathbf{A}}$ , eigenvalues can be find calculating the roots of the polynomial characteristic:

$$|\lambda \mathbf{I} - \tilde{\mathbf{A}}| = 0. \quad (1.1.5)$$

The eigenvalues are also called characteristics roots. Eigenvalues are complex numbers, and when all eigenvalues have negative real part it brings the system to the asymptotic stability.

One of the most powerful property of the state space representation is its **non-unique representation**. That means, for a given dynamical system, an infinite number of state space representation exists, and all of these state space representations will have the same polynomial characteristic, which means, any state space representation of a given dynamical system will behave in the same way.

Suppose the following linear transformation  $\vec{x} = \mathbf{P}\vec{z}$  ( $\dot{\vec{x}} = \mathbf{P}\dot{\vec{z}}$ ) applied to the system of Eq. (1.1.4). The system of Eq. (1.1.4) can be rewritten (in term of the new state space vector  $\vec{z}$ ) as follows

$$\mathbf{P}\dot{\vec{z}} = \tilde{\mathbf{A}}\mathbf{P}\vec{z} \Rightarrow \dot{\vec{z}} = \mathbf{P}^{-1}\tilde{\mathbf{A}}\mathbf{P}\vec{z} \quad (1.1.6)$$

The polynomial characteristic of the new state space representation  $\dot{\vec{z}} = \mathbf{P}^{-1}\tilde{\mathbf{A}}\mathbf{P}\vec{z}$  can be

written as well

$$\begin{aligned}
 |\lambda\mathbf{I} - \mathbf{P}^{-1}\tilde{\mathbf{A}}\mathbf{P}| &= |\lambda\mathbf{P}^{-1}\mathbf{P} - \mathbf{P}^{-1}\tilde{\mathbf{A}}\mathbf{P}| \\
 &= |\mathbf{P}^{-1}(\lambda\mathbf{I} - \tilde{\mathbf{A}})\mathbf{P}| \\
 &= |\mathbf{P}^{-1}|\lambda\mathbf{I} - \tilde{\mathbf{A}}||\mathbf{P}| \\
 &= |\mathbf{P}^{-1}|\mathbf{P}|\lambda\mathbf{I} - \tilde{\mathbf{A}}| \\
 &= |\lambda\mathbf{I} - \tilde{\mathbf{A}}|.
 \end{aligned} \tag{1.1.7}$$

where we have proved that the eigenvalues of  $\tilde{\mathbf{A}}$  are **invariant under a linear transformation\***.

### 1.1.1 Solution of the Homogeneous State Space Equation in Continuous-time Domain

Let's now take a look how equation  $\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t)$  can be solved.

Consider the state space equation

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) \tag{1.1.16}$$

---

\* or recalling that

$$\mathbf{A}^{-1}(\mathbf{BC})^{-1} = (\mathbf{BCA})^{-1} \quad \text{or} \quad \mathbf{A}(\mathbf{BC})^{-1} = (\mathbf{BCA}^{-1})^{-1} \tag{1.1.8}$$

consider the system

$$\dot{\vec{x}} = \tilde{\mathbf{A}}\vec{x} + \tilde{\mathbf{B}}\vec{u} \tag{1.1.9}$$

$$\vec{y} = \mathbf{C}\vec{x} \tag{1.1.10}$$

and the linear transformation  $\vec{x} = \mathbf{T}\vec{z}$  and the equivalent system

$$\dot{\vec{z}} = \tilde{\mathbf{M}}\vec{z} + \tilde{\mathbf{N}}\vec{u} \tag{1.1.11}$$

$$\vec{y} = \mathbf{P}\vec{x} \tag{1.1.12}$$

where

$$\tilde{\mathbf{M}} = \mathbf{T}^{-1}\tilde{\mathbf{A}}\mathbf{T} \tag{1.1.13}$$

$$\tilde{\mathbf{N}} = \mathbf{T}^{-1}\tilde{\mathbf{B}} \tag{1.1.14}$$

$$\mathbf{P} = \mathbf{CT} \tag{1.1.15}$$

The transfer function of the system of Eq. (1.1.11) results as follows

$$\begin{aligned}
 \mathbf{P}(s\mathbf{I} - \tilde{\mathbf{M}})^{-1}\tilde{\mathbf{N}} &= \mathbf{CT}(s\mathbf{I} - \mathbf{T}^{-1}\tilde{\mathbf{A}}\mathbf{T})^{-1}\mathbf{T}^{-1}\tilde{\mathbf{B}} \\
 &= \mathbf{C}(s\mathbf{TT}^{-1} - \tilde{\mathbf{A}})^{-1}\tilde{\mathbf{B}} \\
 &= \mathbf{C}(s\mathbf{I} - \tilde{\mathbf{A}})^{-1}\tilde{\mathbf{B}}
 \end{aligned}$$

which proof the two equivalent systems have the same transfer function.

where  $\vec{x}$  is the state vector ( $n$ -vector) and  $\tilde{\mathbf{A}}$  is an  $n \times n$  constant matrix. Let us assume that the solution of Eq. (1.1.16) is in the form of a vector power series<sup>†</sup> in  $t$ , or

$$\vec{x}(t) = \vec{b}_0 + \vec{b}_1 t + \vec{b}_2 t^2 + \cdots + \vec{b}_k t^k + \cdots \quad (1.1.17)$$

By substituting Eq. (1.1.17) into Eq. (1.1.16), we obtain

$$\begin{aligned} & \vec{b}_1 + 2\vec{b}_2 t + 3\vec{b}_3 t^2 + \cdots + k\vec{b}_k t^{k-1} + \cdots \\ &= \tilde{\mathbf{A}} \left( \vec{b}_0 + \vec{b}_1 t + \vec{b}_2 t^2 + \cdots + \vec{b}_k t^k + \cdots \right) \end{aligned} \quad (1.1.18)$$

$\vec{x}(t)$  is the solution of the Eq. (1.1.18) if and only if the coefficients of the two polynomials (right and left hand side) must be identical for every  $t \in \mathbb{R}$ , which is equivalent to say that a solution of a linear system is a linear function, that means

$$\begin{aligned} \vec{b}_1 &= \tilde{\mathbf{A}} \vec{b}_0 \\ \vec{b}_2 &= \frac{1}{2} \tilde{\mathbf{A}} \vec{b}_1 = \frac{1}{2} \tilde{\mathbf{A}}^2 \vec{b}_0 = \frac{1}{2!} \tilde{\mathbf{A}}^2 \vec{b}_0 \\ \vec{b}_3 &= \frac{1}{3} \tilde{\mathbf{A}} \vec{b}_2 = \frac{1}{6} \tilde{\mathbf{A}}^3 \vec{b}_0 = \frac{1}{3!} \tilde{\mathbf{A}}^3 \vec{b}_0 \\ &\vdots \\ \vec{b}_k &= \frac{1}{k!} \tilde{\mathbf{A}}^k \vec{b}_0 \end{aligned} \quad (1.1.19)$$

By substituting  $t = 0$  in Eq. (1.1.17), we obtain

$$\vec{x}(0) = \vec{b}_0 \quad (1.1.20)$$

From Eq. (1.1.19) we know that every term  $\vec{b}_k$  can be expressed as series of power of  $\tilde{\mathbf{A}}$  namely  $\vec{b}_k = \frac{1}{k!} \tilde{\mathbf{A}}^k \vec{x}(0)$ . The solution  $\vec{x}(t)$  can be written as well

$$\vec{x}(t) = \left( \mathbf{I} + \tilde{\mathbf{A}}t + \frac{1}{2!} \tilde{\mathbf{A}}^2 t^2 + \cdots + \frac{1}{k!} \tilde{\mathbf{A}}^k t^k + \cdots \right) \vec{x}(0) \quad (1.1.21)$$

The expression in parentheses in the right-hand side of Eq. Eq. (1.1.21) is an  $n \times n$  matrix. Because of its similarity to the infinite power series for a scalar exponential, we call it the *matrix exponential* and results in

$$\mathbf{I} + \tilde{\mathbf{A}}t + \frac{1}{2!} \tilde{\mathbf{A}}^2 t^2 + \cdots + \frac{1}{k!} \tilde{\mathbf{A}}^k t^k + \cdots = e^{\tilde{\mathbf{A}}t} \quad (1.1.22)$$

<sup>†</sup>It is often convenient search a solution of the form of a bounded power series as follows:

$$\vec{x}(t) = \sum_{n=0}^{\infty} \vec{b}_n t^n \quad \text{where } \vec{b}_n \in \mathbb{R}^n$$

In term of the matrix exponential, the solution of Eq. (1.1.16) can be written as follows:

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) \quad (1.1.23)$$

**Matrix exponential.** It can be proved that the matrix exponential, namely  $n \times n$  matrix  $\tilde{\mathbf{A}}$ ,

$$e^{\tilde{\mathbf{A}}t} = \sum_{k=0}^{\infty} \frac{\tilde{\mathbf{A}}^k t^k}{k!} \quad (1.1.24)$$

converges absolutely for all finite  $t$ . Hence computer computations for evaluating the elements of  $e^{\tilde{\mathbf{A}}t}$  by using series expansion can easily be carried out.

### 1.1.2 Computation of $e^{\tilde{\mathbf{A}}t}$

Now we investigate on a method for computing  $e^{\tilde{\mathbf{A}}t}$ . Obviously exist different methods, we just propose one method which is useful for engineering approach.

We consider the Laplace transformation approach to the solution of the equation

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) \quad (1.1.25)$$

Taking the Laplace transform of both sides of Eq. (1.1.25), we obtain

$$s\mathbf{X}(s) - \vec{x}(0) = \tilde{\mathbf{A}}\mathbf{X}(s) \quad (1.1.26)$$

Where  $\mathbf{X}(s) = \mathcal{L}[\vec{x}(t)]$ . Hence

$$(s\mathbf{I} - \tilde{\mathbf{A}})\mathbf{X}(s) = \vec{x}(0). \quad (1.1.27)$$

The inverse Laplace transform of  $\mathbf{X}(s) = (s\mathbf{I} - \tilde{\mathbf{A}})^{-1}\vec{x}(0)$  gives the solution of  $\vec{x}(t)$ . Thus

$$\vec{x}(t) = \mathcal{L}^{-1} \left[ (s\mathbf{I} - \tilde{\mathbf{A}})^{-1} \right] \vec{x}(0) \quad (1.1.28)$$

Note that<sup>†</sup>

$$(s\mathbf{I} - \tilde{\mathbf{A}})^{-1} = \frac{\mathbf{I}}{s} + \frac{\tilde{\mathbf{A}}}{s^2} + \frac{\tilde{\mathbf{A}}^2}{s^3} + \dots \quad (1.1.29)$$

the inverse Laplace transform of  $(s\mathbf{I} - \tilde{\mathbf{A}})^{-1}$  is<sup>§</sup>

$$\mathcal{L}^{-1} \left[ (s\mathbf{I} - \tilde{\mathbf{A}})^{-1} \right] = \mathbf{I} + \mathbf{A}t + \frac{\tilde{\mathbf{A}}^2 t^2}{2!} + \frac{\tilde{\mathbf{A}}^3 t^3}{3!} + \dots = e^{\tilde{\mathbf{A}}t} \quad (1.1.30)$$

<sup>†</sup>from geometric series:

$$\sum_{k=1}^n \frac{a^{k-1}}{x^k} = \frac{1 - (a/x)^n}{x - a} \quad \xrightarrow{n \rightarrow +\infty} \quad \frac{1}{x - a} \quad \text{when } |a| < |x|$$

<sup>§</sup>Remind that  $\mathcal{L}^{-1} \left[ \frac{1}{s} \right] = 1 \quad \forall t \in \mathbb{R}^+$

In some case we will use the following notation:  $\Phi(t) = e^{\tilde{\mathbf{A}}t}$ .

**Example 1.1.** Calculate the state transition matrix  $\Phi(t) = e^{\tilde{\mathbf{A}}t}$  of the following state equation:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (1.1.31)$$

The state transition matrix is given as follows:

$$\Phi(t) = e^{\tilde{\mathbf{A}}t} = \mathcal{L}^{-1} \left[ \left( s\mathbf{I} - \tilde{\mathbf{A}} \right)^{-1} \right] \quad (1.1.32)$$

Since

$$\left( s\mathbf{I} - \tilde{\mathbf{A}} \right) = \begin{bmatrix} s & -1 \\ 2 & s+3 \end{bmatrix} \quad (1.1.33)$$

the inverse of  $(s\mathbf{I} - \tilde{\mathbf{A}})$  is

$$\begin{aligned} \left( s\mathbf{I} - \tilde{\mathbf{A}} \right)^{-1} &= \frac{1}{(s+1)(s+2)} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix} = \\ &= \begin{bmatrix} \frac{2}{s+1} - \frac{1}{s+2} & \frac{1}{s+1} - \frac{1}{s+2} \\ -\frac{2}{s+1} + \frac{2}{s+2} & -\frac{1}{s+1} + \frac{2}{s+2} \end{bmatrix} \end{aligned} \quad (1.1.34)$$

Hence

$$\begin{aligned} \Phi(t) &= e^{\tilde{\mathbf{A}}t} = \mathcal{L}^{-1} \left[ \left( s\mathbf{I} - \tilde{\mathbf{A}} \right)^{-1} \right] = \\ &= \begin{bmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix} \end{aligned} \quad (1.1.35)$$

and

$$\Phi^{-1}(t) = e^{-\tilde{\mathbf{A}}t} = \begin{bmatrix} 2e^t - e^{2t} & e^t - e^{2t} \\ -2e^t + 2e^{2t} & -e^t + 2e^{2t} \end{bmatrix} \quad (1.1.36)$$

◇

From a generic point of view a dynamical system can have an input (or more inputs) and can have an output (or more outputs).

As example, we can consider a simple mass, where, by applying a force input  $u(t) = f(t)$ , we want to control its position  $y = x_1$ . Thus is the case of a SISO (single input single output) system where, the whole system model,  $\ddot{x}(t) = f(t)$ , can be written as follows

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}} \vec{x}(t) + \tilde{\mathbf{B}} u(t), & \vec{x}(0) = \vec{x}_0 \\ \vec{y}(t) = \mathbf{C} \vec{x}(t) \end{cases} \quad (1.1.37)$$

where  $u(t) = f(t)$ ,  $\vec{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$ ,  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ ,  $\tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$  and  $\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ .

### 1.1.3 Solution of the Non-Homogeneous State Space Equation in Continuous-time Domain

Let's now evaluate the solution of the non homogenous equation

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}\vec{u}(t) \quad (1.1.38)$$

where  $\vec{x}$  is the state vector ( $n$ -vector),  $\vec{u}$  the input ( $r$ -vector),  $\tilde{\mathbf{A}}$  an  $n \times n$  constant matrix and  $\tilde{\mathbf{B}}$  an  $n \times r$  constant matrix.

By writing Eq. (1.1.38) as

$$\dot{\vec{x}}(t) - \tilde{\mathbf{A}}\vec{x}(t) = \tilde{\mathbf{B}}\vec{u}(t) \quad (1.1.39)$$

and premultiplying both sides of this last equation by  $e^{-\tilde{\mathbf{A}}t}$ , we obtain

$$e^{-\tilde{\mathbf{A}}t} [\dot{\vec{x}}(t) - \tilde{\mathbf{A}}\vec{x}(t)] = e^{-\tilde{\mathbf{A}}t} \tilde{\mathbf{B}}\vec{u}(t) \quad (1.1.40)$$

or

$$\frac{d}{dt} [e^{-\tilde{\mathbf{A}}t} \vec{x}(t)] = e^{-\tilde{\mathbf{A}}t} \tilde{\mathbf{B}}\vec{u}(t) \quad (1.1.41)$$

Integrating the preceding equation between 0 and  $t$  gives

$$e^{-\tilde{\mathbf{A}}t} \vec{x}(t) = \vec{x}(0) + \int_0^t e^{-\tilde{\mathbf{A}}\tau} \tilde{\mathbf{B}}\vec{u}(\tau) d\tau \quad (1.1.42)$$

or

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) + \int_0^t e^{\tilde{\mathbf{A}}(t-\tau)} \tilde{\mathbf{B}}\vec{u}(\tau) d\tau \quad (1.1.43)$$

Eq. (1.1.43) can also be written as follows:

$$\vec{x}(t) = \Phi(t)\vec{x}(0) + \int_0^t \Phi(t-\tau) \tilde{\mathbf{B}}\vec{u}(\tau) d\tau \quad (1.1.44)$$

where  $\Phi(t) = e^{\tilde{\mathbf{A}}t}$

**Example 1.2. (Solution of the double integrator)** Consider the system (double integrator)

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ \vec{x}(0) &= \begin{bmatrix} x_0 \\ v_0 \end{bmatrix} \end{aligned} \quad (1.1.45)$$

where  $\vec{x}(0) = \begin{bmatrix} x_0 \\ v_0 \end{bmatrix}$  is the initial condition,  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$  and  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$ .

The solution of the dynamical system is

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) \quad (1.1.46)$$

and can be solved as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1}\left\{\left[s\mathbf{I} - \tilde{\mathbf{A}}\right]^{-1}\right\}\vec{x}(0) = \mathcal{L}^{-1}\left\{\frac{1}{s^2} \begin{bmatrix} s & 1 \\ 0 & s \end{bmatrix}\right\}\vec{x}(0) = \\ &= \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \vec{x}(0) = \begin{bmatrix} x_0 + v_0 t \\ v_0 \end{bmatrix} \end{aligned} \quad (1.1.47)$$

which represents the Newton's first law that states: if a body is at rest or moving at a constant speed in a straight line, it will remain at rest or keep moving in a straight line at constant speed unless it is acted upon by a force. ◀

**Example 1.3. (Pendulum)** Consider the pendulum of Figure 1.1

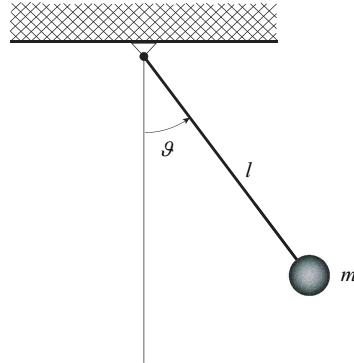


Figure 1.1: Pendulum.

Where the kinetic and potential energy can be expressed as follows

$$E_{kin} = \frac{1}{2}ml^2\dot{\vartheta}^2$$

$$E_{pot} = mgl(1 - \cos \vartheta)$$

Applying the Euler-Lagrange equation we obtain

$$\frac{d}{dt} \left( \frac{\partial E_{kin}}{\partial \dot{q}_i} \right) - \frac{\partial E_{kin}}{\partial q_i} + \frac{\partial E_{pot}}{\partial q_i} = Q_{diss}$$

where  $Q_{diss} = -b\dot{\vartheta}$  which is a force due to the friction and results in the following second order differential equation

$$ml^2\ddot{\vartheta} + mgl \sin \vartheta = -b\dot{\vartheta} \quad (1.1.48)$$

which in state space form results as follows

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{g}{l} \sin x_1 - \frac{b}{ml^2} x_2 \end{cases} \quad (1.1.49)$$

Now we consider the case of small oscillation introducing the approximation  $\sin \vartheta \approx \vartheta$  and we consider the case  $b = 0$  and  $g = l$  which results in the following harmonic oscillator

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ \vec{x}(0) &= \begin{bmatrix} 1 & 0 \end{bmatrix}^T \end{aligned} \quad (1.1.50)$$

where  $\vec{x}(0)$  is the initial condition,  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$  and  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$ .

The solution of the dynamical system is

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) \quad (1.1.51)$$

and can be solved as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1} \left\{ \left[ s\mathbf{I} - \tilde{\mathbf{A}} \right]^{-1} \right\} \vec{x}(0) = \mathcal{L}^{-1} \left\{ \frac{1}{s^2 + 1} \begin{bmatrix} s & 1 \\ -1 & s \end{bmatrix} \right\} \vec{x}(0) = \\ &= \begin{bmatrix} \cos(t) & \sin(t) \\ -\sin(t) & \cos(t) \end{bmatrix} \vec{x}(0) = \begin{bmatrix} \cos(t) & -\sin(t) \end{bmatrix}^T \end{aligned} \quad (1.1.52)$$

The solution  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T = [\cos(t) \ -\sin(t)]^T$  can be plotted in the plane  $(x_1, x_2)$  - which is called *phase-portrait*. The solution  $\vec{x}(t)$  is the parametric representation of a circle with unitary radius. Selecting any initial condition different from zero, we can observe that the trajectory state vector is stationary - there isn't any loss of energy - and the motus is a periodic cycles as in a harmonic oscillator, see also Figure 1.2.

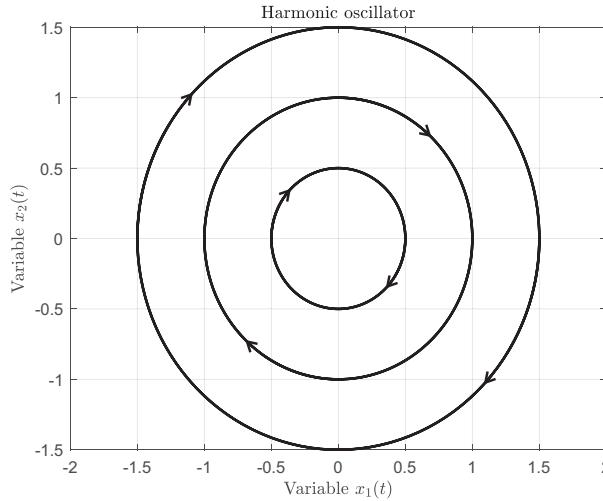


Figure 1.2: Phase portrait of the harmonic oscillator for different initial conditions.

Consider now the damped version of the harmonic oscillator, where  $ml^2 = 1$  and  $b = 1/2$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\vec{x}(0) = [1 \ 0]^T \quad (1.1.53)$$

where  $\vec{x}(0)$  is the initial condition and  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -1 & -\frac{1}{2} \end{bmatrix}$  and  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$ .

The solution of the dynamical system is

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) \quad (1.1.54)$$

and can be solved as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1} \left\{ \left[ s\mathbf{I} - \tilde{\mathbf{A}} \right]^{-1} \right\} \vec{x}(0) = \mathcal{L}^{-1} \left\{ \frac{1}{2s^2 + s + 2} \begin{bmatrix} 2s+1 & 2 \\ -2 & 2s \end{bmatrix} \right\} \vec{x}(0) = \\ &= \begin{bmatrix} e^{-\alpha t} [\gamma \cos(\beta t) + \sin(\beta t)] & e^{-\alpha t} \sin(\beta t) \\ -e^{-\alpha t} \sin(\beta t) & e^{-\alpha t} [\gamma \cos(\beta t) - \sin(\beta t)] \end{bmatrix} \vec{x}(0) \\ &= \begin{bmatrix} e^{-\alpha t} (\gamma \cos(\beta t) + \sin(\beta t)) & -e^{-\alpha t} \sin(\beta t) \end{bmatrix}^T \end{aligned} \quad (1.1.55)$$

where  $\alpha, \beta, \gamma \in \mathbb{R}^+$ .

The plotting of the solution  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$  in the plane  $(x_1, x_2)$  is shown in Figure 1.3, where its sink nature is noticeable. Starting from any initial condition the trajectory

of the state vector goes asymptotically to the origin.

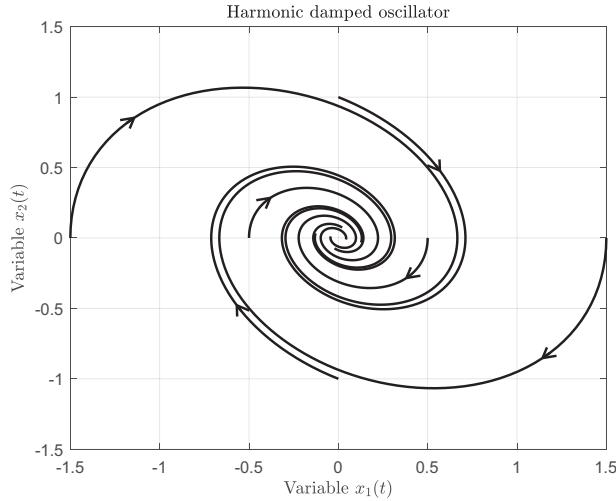


Figure 1.3: Phase portrait of the harmonic damped oscillator.

**Non-Homogeneous case** - Let's now consider the case without friction, and where the pendulum has an input, as follows

$$ml^2\ddot{\vartheta}(t) + mgl \sin \vartheta(t) = \tau_m(t) \quad (1.1.56)$$

or

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{g}{l} \sin x_1 + \frac{1}{ml^2} \tau_m \end{cases} \quad (1.1.57)$$

or

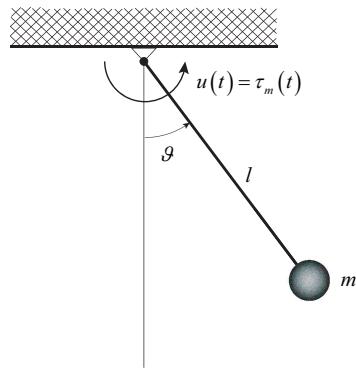


Figure 1.4: Pendulum.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tau_m \quad (1.1.58)$$

$$\vec{x}(0) = [1 \ 0]^T$$

where  $\vec{x}(0)$  is the initial condition,  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ ,  $\tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ ,  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$  and  $u(t) = \tau_m(t)$ .

By Eq. (1.1.59) here reported

$$\vec{x}(t) = \Phi(t)\vec{x}(0) + \int_0^t \Phi(t-\tau)\tilde{\mathbf{B}}\vec{u}(\tau)d\tau \quad (1.1.59)$$

where  $\Phi(t) = \begin{bmatrix} \cos(t) & \sin(t) \\ -\sin(t) & \cos(t) \end{bmatrix}$ , we obtain  $x(t)$  as follows

$$\vec{x}(t) = \begin{bmatrix} \cos(t) \\ -\sin(t) \end{bmatrix} + \int_0^t \begin{bmatrix} \sin(t-\tau) \\ \cos(t-\tau) \end{bmatrix} \tau_m(\tau)d\tau \quad (1.1.60)$$

◇

## 1.2 Discretization

Developing control systems, we can observe that both digital and analog substructures coexist together. On one hand e.g. the plant and the actuators must be designed in continuous time domain because that is their nature. On the other hand controllers, stored into processor units, must be represented in discrete-time domain. Many controllers are also available in continuous-time domain by the use of op-amp and other monolithic electronic devices.

Analysing discrete-time control systems, the concept of **Impulse sampling** shall be investigated.

**Impulse sampling.** A sampler converts a continuous-time signal, e.g.  $x(t)$ , into a train of pulses occurring at equal spaced sampling instant  $t = 0, t_s, 2t_s, \dots$ , where  $t_s$  is the sampling period, resulting in e.g.  $x(kt_s)$  or simply  $x(k)$ .

Mathematically speaking the sample process can be modelized as well

$$\begin{aligned} x^*(t) &= x(0)\delta(t) + x(t_s)\delta(t-t_s) + \dots + x(kt_s)\delta(t-kt_s) + \dots \\ &= \sum_{k=0}^{\infty} x(kt_s)\delta(t-kt_s) \end{aligned} \quad (1.2.1)$$

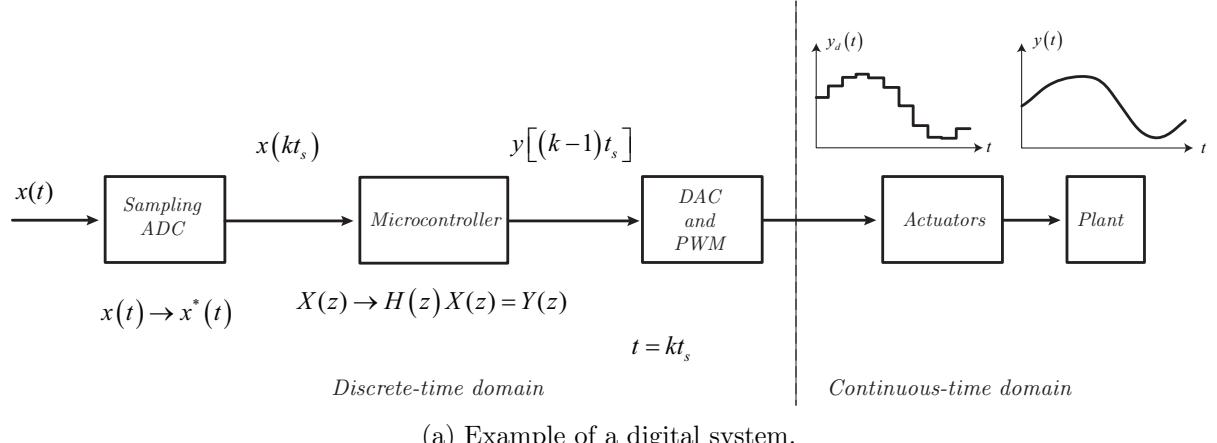
where  $x^*(t)$  is the sampled version of the original function  $x(t)$  and  $\delta(t)$  the delta function.

Once, at time step  $t = kt_s$ , the original signal  $x(t)$  has been sampled, obtaining the quantity  $x(kt_s)$ , the microcontroller (or any other digital controller) **stores and keeps constant** the value of  $x(kt_s)$  along the whole time period  $t_s$ . When the time period  $t_s$  has expired the valued of  $x(kt_s)$  will be updated as

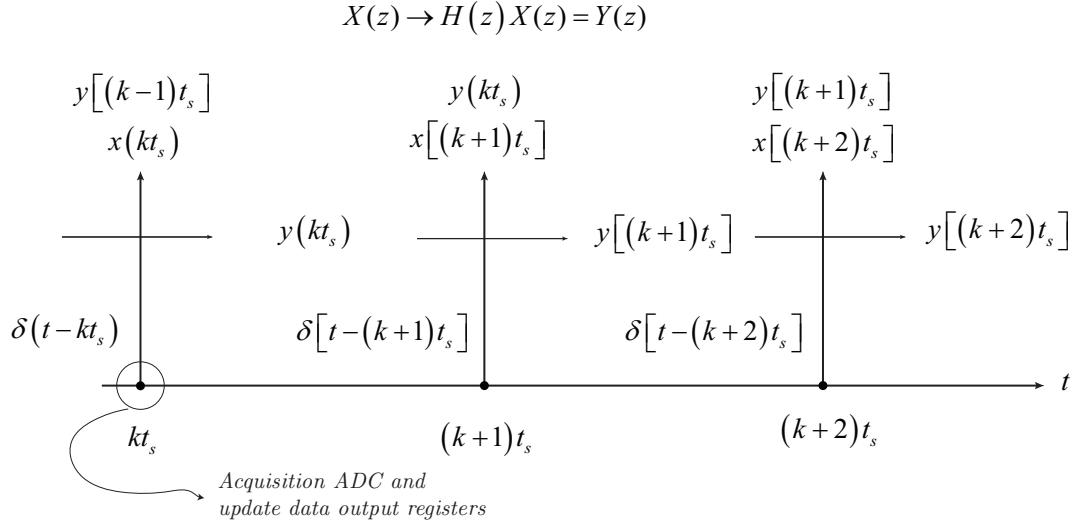
$$x(kt_s) \rightarrow x[(k+1)t_s] \quad (1.2.2)$$

A data hold circuit (basically performed by the **microcontroller** itself) converts the sampled data into a continuous time signal.

The output of the microcontroller unit is called  $y(kt_s)$



(a) Example of a digital system.



(b) Discrete time evolution inside a digital system, where the natural presence of a unit delay step ( $z^{-1}$ ) is shown.

Figure 1.5: Description of a discrete time system.

As shown in Figure 1.5 at every  $t = kt_s$  the microcontroller sample the input  $x(kt_s)$  and release the output calculated at the previous step  $y[(k+1)t_s]$ .

Moreover, as shown in Figure 1.5 between any two consecutive sampling values the **sampler** transmits no information. A sampled data is hold inside the micro-controller while actuators convert from discrete-time data to continuous-time, this part of the process can be identified as a function called **zero-order hold**.

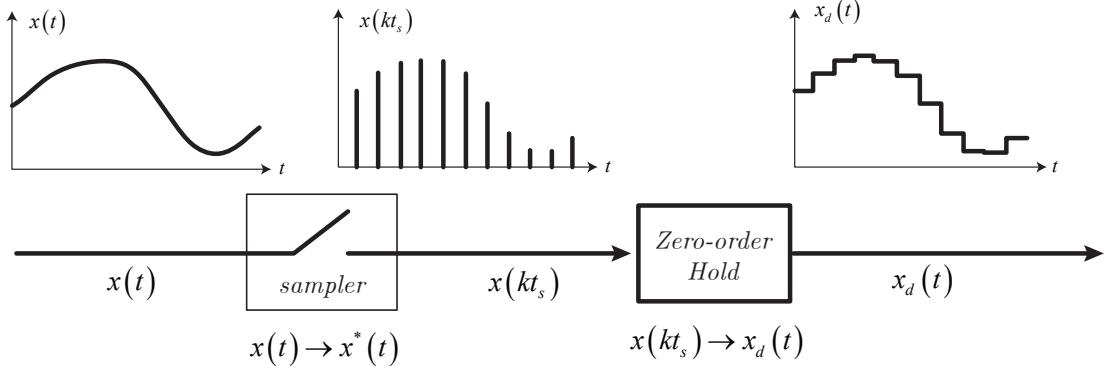


Figure 1.6: Sampler and zero-order hold.

**Zero-order hold**(transfer function). Assume that the signal  $x(t)$  is zero for  $t < 0$ . The output  $x_d(t)$  is related to  $x(t)$  as follows

$$\begin{aligned}
 x_d(t) &= x(0) [h(t) - h(t - t_s)] + x(t_s) [h(t - t_s) - h(t - 2t_s)] + \\
 &\quad + x(2t_s) [h(t - 2t_s) - h(t - 3t_s)] + \dots \\
 &= \sum_{k=0}^{\infty} x(kt_s) [h(t - kt_s) - h[t - (k+1)t_s]]
 \end{aligned} \tag{1.2.3}$$

where  $h(t)$  is the Heaviside function. Since

$$\mathcal{L}[h(t - kt_s)] = \frac{e^{-kt_ss}}{s}$$

the Laplace transform of Eq. (1.2.3) becomes

$$\begin{aligned}
 \mathcal{L}[x_d(t)] &= X_d(s) = \sum_{k=0}^{\infty} x(kt_s) \frac{e^{-kt_ss} - e^{-(k+1)t_ss}}{s} \\
 &= \frac{1 - e^{-t_ss}}{s} \sum_{k=0}^{\infty} x(kt_s) e^{-kt_ss}
 \end{aligned} \tag{1.2.4}$$

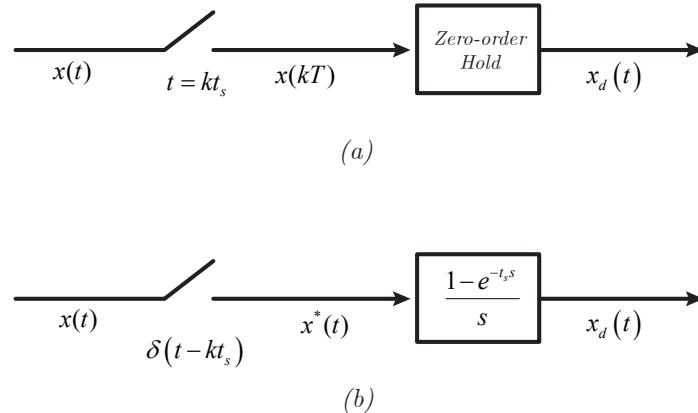


Figure 1.7: (a) System with a sampler and zero order hold; (b) equivalent system (mathematical model) which consists of an impulse sampler and transfer function  $(1 - e^{-sT})/s$ .

where

$$X^*(s) = \sum_{k=0}^{\infty} x(kt_s)e^{-kt_ss} \quad (1.2.5)$$

The **transfer function of the zero-order hold** is given by

$$H_0(s) = \frac{1 - e^{-t_ss}}{s} \quad (1.2.6)$$

considering the term  $x(kt_s)$  constant in the equation Eq. 1.2.5, we obtain

$$\mathcal{L}^{-1}[X^*(s)] = x^*(t) = \sum_{k=-\infty}^{\infty} x(kt_s)\delta(t - kt_s) \quad (1.2.7)$$

Then, the sampled signal  $x^*(t)$  (in time domain) and the original continuous time signal  $x(t)$  can be related as follows:

$$\begin{aligned} x^*(t) &= \dots + x(0)\delta(t) + x(t_s)\delta(t - t_s) + \dots + x(kt_s)\delta(t - kt_s) + \dots \\ &= \sum_{k=-\infty}^{\infty} x(kt_s)\delta(t - kt_s) \end{aligned} \quad (1.2.8)$$

**The  $\mathcal{Z}$ -transform(Derivation).** Consider the Laplace transform equation

$$X^*(s) = \sum_{k=0}^{\infty} x(kt_s)e^{-kt_ss} \quad (1.2.9)$$

We define

$$z = e^{t_s s}$$

or

$$s = \frac{1}{t_s} \ln z$$

Then Eq. 1.2.9 becomes

$$X^*(s)|_{s=\frac{1}{t_s} \ln z} = X(z) = \sum_{k=0}^{\infty} x(kt_s)z^{-k} \quad (1.2.10)$$

Note that the variable  $z$  is a complex variable and  $t_s$  is the sampling period.

When we transform  $e^{t_s s}$  to  $z$ , the concept of impulse sampling (which is purely a mathematical process) enables us to analyse by the  $\mathcal{Z}$ -transform method discrete time control systems which involve samplers and hold circuits. This means that by use the complex variable  $z$ , the techniques developed for the Laplace transform methods (such as the frequency response and root locus methods) can be readily applied to analyse discrete-time control systems involving sampling operations.

**Example 1.4 (Obtain the  $\mathcal{Z}$ -transform of the Heaviside function  $h(t)$ ).** Define

$$x(t) = h(t)$$

The impulse-sampled signal  $x^*(t)$  can be obtained as follows:

$$x^*(t) = \sum_{k=0}^{\infty} x(kt_s)\delta(t - kt_s) = \sum_{k=0}^{\infty} \delta(t - kt_s)$$

Then the Laplace transform of  $x^*(t)$  can be given by

$$X^*(s) = \sum_{k=0}^{\infty} e^{-kt_s s} = \frac{1}{1 - e^{-t_s s}}$$

Note that  $\sum_{k=0}^{\infty} x^{-k} = \frac{1}{1-x^{-1}}$  when  $|x| > 1$ . Hence

$$X(z) = \mathcal{Z}[h(t)] = \frac{1}{1 - z^{-1}}$$

◇

**Example 1.5 (Obtain the  $\mathcal{Z}$ -transform of the sum of a generic function  $x(t)$ ).** Consider

the function  $x_d(kt_s)$  which is a sum of function  $x(nt_s)$ , where  $n = 0, 1, 2, \dots, k$ , such that

$$x_d(kt_s) = \sum_{n=0}^k x(nt_s) \quad k = 0, 1, 2, \dots$$

where  $x_d(kt_s) = 0$  for  $k < 0$ . Obtain the  $\mathcal{Z}$  transform of  $x_d(kt_s)$ .

First note that

$$x_d(kt_s) = x(0) + x(t_s) + \dots + x[(k-1)t_s] + x(kt_s)$$

$$x_d[(k-1)t_s] = x(0) + x(t_s) + \dots + x[(k-1)t_s]$$

Hence

$$x_d(kt_s) - x_d[(k-1)t_s] = x(kt_s) \quad k = 0, 1, 2, \dots$$

Therefore,

$$\mathcal{Z}[x_d(kt_s) - x_d[(k-1)t_s]] = \mathcal{Z}[x(kt_s)]$$

or

$$X_d(z) - z^{-1}X_d(z) = X(z)$$

which yields

$$X_d(z) = \frac{1}{1 - z^{-1}}X(z)$$

or

$$\mathcal{Z}\left[\sum_{n=0}^k x(nt_s)\right] = \frac{1}{1 - z^{-1}}X(z) \quad (1.2.11)$$

◇

**Example 1.6.** PID control representation in discrete-time domain. The **PID** control in continuous time domain is given by:

$$m(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d \dot{e}(t) \quad (1.2.12)$$

where  $e(t)$  is the input to the controller.

In order to obtain the pulse transfer function for the digital **PID** controller, we may discretize the above equation. By approximating the integral term by the right Riemann summation and the derivative term by a two point difference form, we obtain:

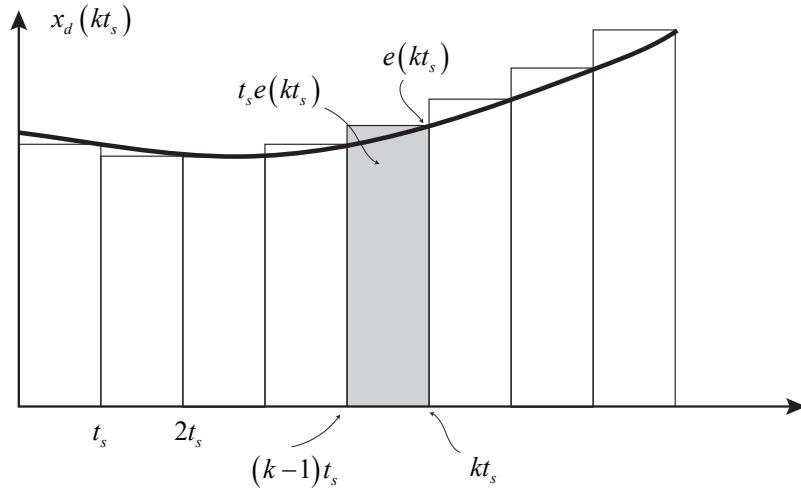


Figure 1.8: Right Riemann Integration

$$m(kt_s) = k_p e(kt_s) + k_i t_s \sum_{n=0}^k e(nt_s) + \frac{k_d}{t_s} \{e(kt_s) - e[(k-1)t_s]\} \quad (1.2.13)$$

Where the  $\mathcal{Z}$ -Transform of the integral term becomes, see Eq. (1.2.11)

$$\mathcal{Z} \left[ \sum_{n=0}^k e(nt_s) \right] = \frac{1}{1 - z^{-1}} E(z) \quad (1.2.14)$$

**The pulse transfer function of the digital PID becomes**

$$\text{PID}(z) = \frac{M(z)}{E(z)} = k_p + t_s \frac{k_i}{1 - z^{-1}} + \frac{k_d}{t_s} (1 - z^{-1}) \quad (1.2.15)$$

Where in discrete time domain becomes  $z^{-1} \rightarrow (k - 1)$

$$m(k) = m_p(k) + m_i(k) + m_d(k)$$

where

$$m_p(k) = k_p e(k) \quad (1.2.16)$$

$$m_i(k) = m_i(k - 1) + t_s k_i e(k)$$

$$m_d(k) = \frac{k_d}{t_s} [e(k) - e(k - 1)]$$

◇

### 1.2.1 Discrete-time Transform Methods

It is common practice to design controller, filter, and general models in continuous time domain, and at the end of the design process, convert them into a discrete-time system using a discretization method. In the following, we are going to consider two different methods of discretization: *backward difference method* and *bilinear (or trapezoidal) transformation method*.

Shall be remembered that discretization of continuous time models, not only the particular method used, but also the sampling frequency chosen, will affect the dynamic characteristic of the resulting model. If the sampling frequency is sufficiently high, the equivalent discrete-time model will yield a good approximation to a given continuous time model.

**Example 1.7. (Backward difference method.)** Digital Implementation of a Cruise Control. Consider the continuous time model

$$m \frac{dv(t)}{dt} + bv(t) = f(t) \quad (1.2.17)$$

or, where  $b/m = \mu$  and  $u(t) = f(t)/m$

$$\frac{dv(t)}{dt} + \mu v(t) = u(t) \quad (1.2.18)$$

where the Laplace transform is

$$\frac{V(s)}{U(s)} = \frac{1}{s + \mu} \quad (1.2.19)$$

and lets try to discretize with backward difference method: Let us integrate both sides from 0 to  $t$ :

$$\int_0^t \frac{dv(t)}{dt} dt = -\mu \int_0^t v(t) dt + \int_0^t u(t) dt \quad (1.2.20)$$

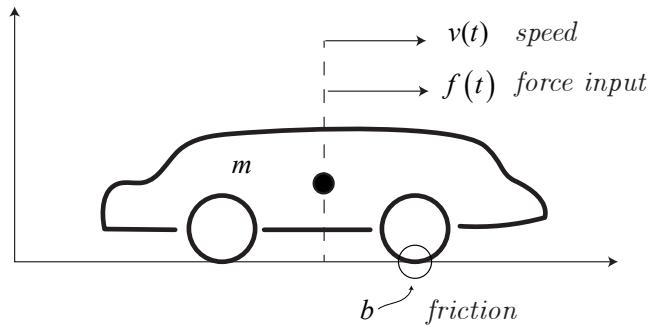


Figure 1.9: Continuous time model.

Substituting  $t \rightarrow kt_s$  (we are in the  $kt_s$  time step)

$$\int_0^{kt_s} \frac{dv(t)}{dt} dt = -\mu \int_0^{kt_s} v(t) dt + \int_0^{kt_s} u(t) dt \quad (1.2.21)$$

which become

$$v(kt_s) - v(0) = -\mu \int_0^{kv} v(t) dt + \int_0^{kv} u(t) dt \quad (1.2.22)$$

Consider now the time step  $(k-1)t_s$  we obtain

$$v[(k-1)t_s] - v(0) = -\mu \int_0^{(k-1)t_s} v(t) dt + \int_0^{(k-1)t_s} u(t) dt \quad (1.2.23)$$

By subtracting Eq. (1.2.23) from Eq (1.2.22) we obtain

$$v(kt_s) - v[(k-1)t_s] = -\mu \int_{(k-1)t_s}^{kt_s} v(t) dt + \int_{(k-1)t_s}^{kt_s} u(t) dt \quad (1.2.24)$$

The integral term are now calculated by the backward difference method then

$$\int_{(k-1)t_s}^{kt_s} v(t) dt = v(kt_s) \int_{(k-1)t_s}^{kt_s} dt = t_s v(kt_s) \quad (1.2.25)$$

and

$$\int_{(k-1)t_s}^{kt_s} u(t) dt = u(kt_s) \int_{(k-1)t_s}^{kt_s} dt = t_s u(kt_s) \quad (1.2.26)$$

Hence we obtain

$$v(kt_s) = v[(k-1)t_s] - \mu t_s v(kt_s) + t_s u(kt_s) \quad (1.2.27)$$

The  $\mathcal{Z}$ -Transform of Eq. (1.2.27) is

$$V(z) = V(z)z^{-1} - \mu t_s V(z) + t_s U(z) \quad (1.2.28)$$

for which we get

$$\frac{V(z)}{U(z)} = \frac{1}{\frac{1-z^{-1}}{t_s} + \mu} \quad (1.2.29)$$

By comparing the  $\mathcal{Z}$ -transform with  $\mathcal{L}$ -transform Eq. (1.2.19) we note that the right hand

sides of these two equations become identical if we let

$$s = \frac{1 - z^{-1}}{t_s} \quad (1.2.30)$$

The Eq. (1.2.30) is the mapping from the  $s$  plane to the  $z$  plane when the backward difference method is used to discretize Eq. (1.2.18).

The stability region in the  $s$  plane,  $(\text{Re}(s) < 0)$ , can be mapped by Eq. 1.2.30 into the  $z$  plane as follows.

Noting that the stable region in the  $s$  plane is given by  $\text{Re}(s) < 0$ , hence we can write

$$\begin{aligned} \text{Re}\left(\frac{1 - z^{-1}}{t_s}\right) &= \frac{1}{t_s} \text{Re}\left(\frac{z - 1}{z}\right) = \\ &= \frac{1}{t_s} \text{Re}\left(\frac{\sigma + j\omega - 1}{\sigma + j\omega}\right) < 0 \end{aligned} \quad (1.2.31)$$

where

$$\begin{aligned} \text{Re}\left(\frac{(\sigma + j\omega - 1)(\sigma - j\omega)}{(\sigma + j\omega)(\sigma - j\omega)}\right) &= \\ &= \text{Re}\left(\frac{\sigma^2 - \sigma + \omega^2 + j\omega}{\sigma^2 + \omega^2}\right) = \\ &= \frac{\sigma^2 - \sigma + \omega^2}{\sigma^2 + \omega^2} < 0 \end{aligned} \quad (1.2.32)$$

which can be written as

$$\left(\sigma - \frac{1}{2}\right)^2 + \omega^2 < \left(\frac{1}{2}\right)^2 \quad (1.2.33)$$

The stable region can thus be mapped into a circle with center at  $\sigma = \frac{1}{2}$ ,  $\omega = 0$  and radius equal to  $\frac{1}{2}$  as shown in Figure 1.10. That mean the discretization method based on *backward difference*, **transform a continuous-time stable system into a discrete-time stable system**.

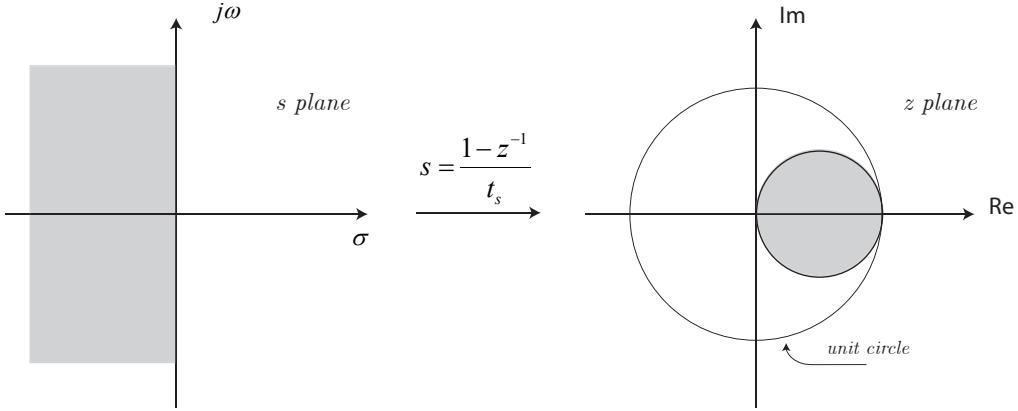


Figure 1.10: Mapping of the left half of the *s* plane into *z* plane by  $s = (1 - z^{-1})/t_s$  (backward difference method).

◇

**Example 1.8. (Bilinear transformation method.)** the bilinear transformation method is also called the trapezoidal integration method or Tustin transformation method. By this method we approximate the areas

$$\int_{(k-1)t_s}^{kt_s} x_d(t)dt \quad \text{and} \quad \int_{(k-1)t_s}^{kt_s} x(t)dt$$

by  $\frac{1}{2}[x_d(kt_s) - x_d((k-1)t_s)]t_s$  and  $\frac{1}{2}[x(kt_s) - x((k-1)t_s)]t_s$ , respectively; Notice that the integration scheme assumes that the variation between successive sampling points is linear, as shown in Figure 1.11. Hence by the use of the bilinear transformation method, Eq. (1.2.27) can be written as follows:

$$v(kt_s) = v[(k-1)t_s] - \mu \frac{t_s}{2} [v(kt_s) + v((k-1)t_s)] + \frac{t_s}{2} [u(kt_s) + u((k-1)t_s)] \quad (1.2.34)$$

The  $\mathcal{Z}$ -transform of Eq, (1.2.34) is

$$V(z) = z^{-1}V(z) - \mu \frac{t_s}{2} [V(z) + z^{-1}V(z)] + \frac{t_s}{2} [U(z) + z^{-1}U(z)] \quad (1.2.35)$$

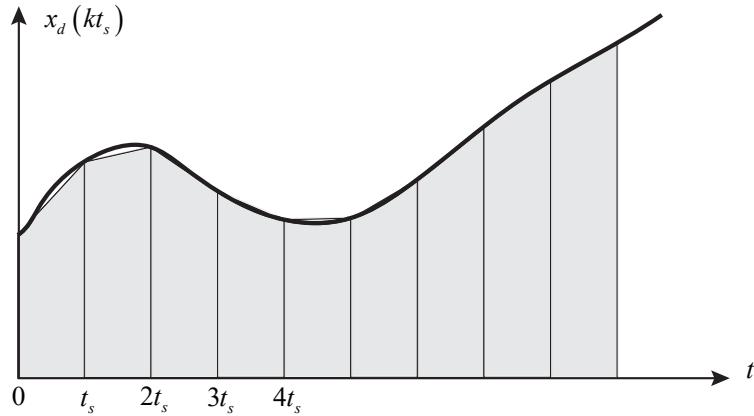


Figure 1.11: Area approximation by the bilinear transformation method (also called Tustin).

for which we get

$$\frac{V(z)}{U(z)} = \frac{1}{\frac{2}{t_s} \frac{1-z^{-1}}{1+z^{-1}} + \mu} \quad (1.2.36)$$

By comparing the  $\mathcal{Z}$ -transform with  $\mathcal{L}$ -transform Eq. (1.2.36) we note that the right hand sides of these two equations become identical if we let

$$s = \frac{2}{t_s} \frac{1-z^{-1}}{1+z^{-1}} \quad (1.2.37)$$

The Eq. (1.2.37) is the mapping from the  $s$  plane to the  $z$  plane when the bilinear transformation method is used to discretize Eq. (1.2.18).

The stability region in the  $s$  plane can be mapped by Eq. (1.2.37) into the  $z$  plane as follows. Noting that the stable region in the  $s$  plane is given by  $\text{Re}(s) < 0$ , hence we can write

$$\begin{aligned} \text{Re} \left( \frac{2}{t_s} \frac{1-z^{-1}}{1+z^{-1}} \right) &= \frac{2}{t_s} \text{Re} \left( \frac{z-1}{z+1} \right) = \\ &= \frac{2}{t_s} \text{Re} \left( \frac{\sigma - 1 + j\omega}{\sigma + 1 + j\omega} \right) < 0 \end{aligned} \quad (1.2.38)$$

where

$$\begin{aligned} \text{Re} \left( \frac{\sigma - 1 + j\omega}{\sigma + 1 + j\omega} \right) &= \\ &= \frac{\sigma^2 - 1 + \omega^2}{(\sigma + 1)^2 + \omega^2} < 0 \end{aligned} \quad (1.2.39)$$

Eq. (1.2.39) is hold when

$$\sigma^2 + \omega^2 < 1 \quad (1.2.40)$$

The stable region can thus be mapped into a circle with center at  $\sigma = 0, \omega = 0$  and radius equal to 1 as shown in Figure 1.12

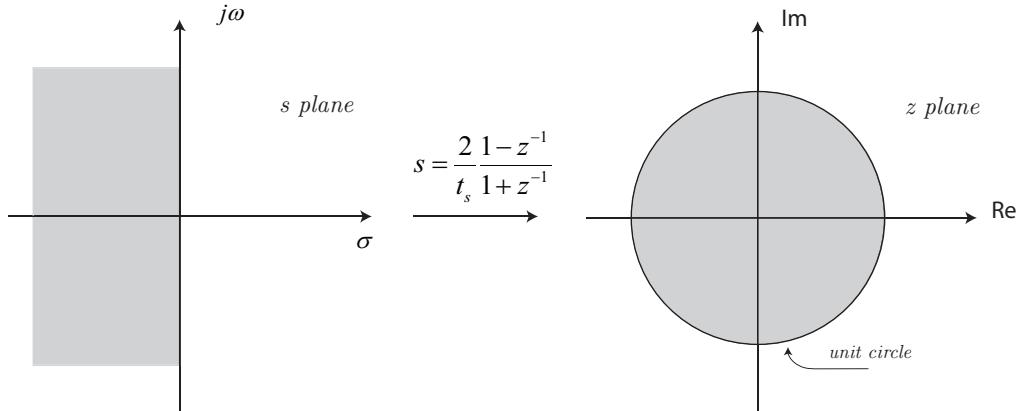


Figure 1.12: Mapping of the left half of the *s* plane into *z* plane by  $s = 2/t_s(1 - z^{-1})/(1 + z^{-1})$  (bilinear transformation method).

That mean the discretization method based on *Tustin*, **transform a continuous-time stable system into a discrete-time stable system.** ◁

### 1.3 Discrete-time Representation of State Space

Up to now we have seen system represented in the contest of continuous time domain where they are described by differential equation. Any continuous-time dynamical system can be rewritten in the form of discrete-time domain. This process is called *discretization of continuous-time systems*. The discretized system is always an approximation of the continuous-time case.

Consider the continuous-time domain representation of Eq. (1.3.1)

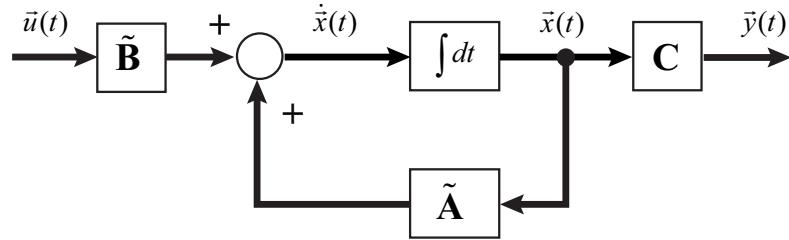
$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}} \vec{x}(t) + \tilde{\mathbf{B}} u(t), & \vec{x}(0) = \vec{x}_0 \\ \vec{y}(t) = \mathbf{C} \vec{x}(t) \end{cases} \quad (1.3.1)$$

and the equivalent discrete-time domain representation of Eq. (1.3.2)

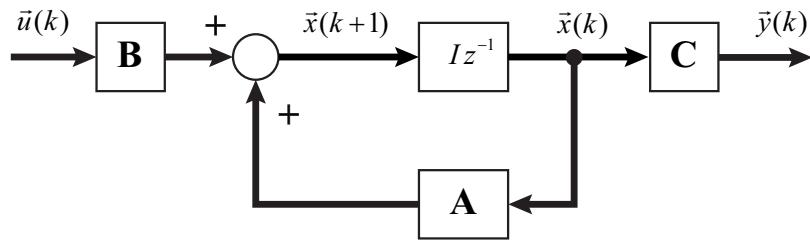
$$\begin{cases} \vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k), & \vec{x}(0) = \vec{x}_0 \\ \vec{y}(k) = \mathbf{C} \vec{x}(k) \end{cases} \quad (1.3.2)$$

they represent the same dynamical system. As we can see the two transition matrices  $\tilde{\mathbf{A}}$  and  $\mathbf{A}$  as well as the two input matrices  $\tilde{\mathbf{B}}$  and  $\mathbf{B}$  are different.

Exist a transformation method which, for a given discrete time step, correlates matrix  $\tilde{\mathbf{A}}$  to matrix  $\mathbf{A}$  and matrix  $\tilde{\mathbf{B}}$  to matrix  $\mathbf{B}$ .



(a) Block diagram of the linear time-invariant continuous-time system represented in state space.



(b) Block diagram of the linear time-invariant discrete-time system represented in state space.

Figure 1.13: State space representation of a given dynamical system in both continuous and discrete-time domain.

Before to derive the relation between  $\tilde{\mathbf{A}}$  and  $\mathbf{A}$ , and between  $\tilde{\mathbf{B}}$  and  $\mathbf{B}$ , we present some relevant and propaedeutic results for discrete time state space representation.

As shown for the continuous time case, we now present the solution for the discrete-time case, namely we present the solution of the system of Eq. (1.3.2)

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}u(k) \quad (1.3.3)$$

by a recursion procedure and then by the  $z$  transform method. Them we discuss methods for computing  $(z\mathbf{I} - \mathbf{A})^{-1}$ .

In general, discrete-time equation are easier to solve than differential equations because the former can be solved easily by means of recursion procedure. The recursion procedure is quite simple and convenient for digital computations.

### 1.3.1 Solutions of the Non Homogeneous State Space Equation in Discrete-time Domain

Consider the following state equation and output equation:

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \quad (1.3.4)$$

$$\vec{y}(k) = \mathbf{C}\vec{x}(k) \quad (1.3.5)$$

The solution, for any positive integer, may be obtained directly by recursion, as follows:

$$\begin{aligned} \vec{x}(1) &= \mathbf{A}\vec{x}(0) + \mathbf{B}\vec{u}(0) \\ \vec{x}(2) &= \mathbf{A}\vec{x}(1) + \mathbf{B}\vec{u}(1) = \mathbf{A}^2\vec{x}(0) + \mathbf{AB}\vec{u}(0) + \mathbf{B}\vec{u}(1) \\ \vec{x}(3) &= \mathbf{A}\vec{x}(2) + \mathbf{B}\vec{u}(2) = \mathbf{A}^3\vec{x}(0) + \mathbf{A}^2\mathbf{B}\vec{u}(0) + \mathbf{AB}\vec{u}(1) + \mathbf{B}\vec{u}(2) \\ &\vdots \end{aligned} \quad (1.3.6)$$

By repeating this procedure, we obtain the solution of Eq. (1.3.4)

$$\vec{x}(k) = \mathbf{A}^k\vec{x}(0) + \sum_{j=0}^{k-1} \mathbf{A}^{k-j-1}\mathbf{B}\vec{u}(j) \quad k = 1, 2, 3, \dots \quad (1.3.7)$$

Clearly,  $\vec{x}(k)$  consists of two parts, one representing the contribution of the initial state  $\vec{x}(0)$ , and the other the contribution of the input  $\vec{u}(j)$ , where  $j = 0, 1, 2, \dots, k-1$ . The output  $\vec{y}(k)$  is given by

$$\vec{y}(k) = \mathbf{C}\mathbf{A}^k\vec{x}(0) + \mathbf{C}\sum_{j=0}^{k-1} \mathbf{A}^{k-j-1}\mathbf{B}\vec{u}(j) \quad (1.3.8)$$

Notice that it is possible to write the solution of the homogeneous state equation

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) \quad (1.3.9)$$

as

$$\vec{x}(k) = \Psi(k)\vec{x}(0) \quad (1.3.10)$$

where  $\Psi(k)$  is a unique  $n \times n$  matrix satisfying the condition

$$\Psi(k+1) = \mathbf{A}\Psi(k) \quad \Psi(0) = \mathbf{I} \quad (1.3.11)$$

Clearly,  $\Psi(k)$  can be given by

$$\Psi(k) = \mathbf{A}^k \quad (1.3.12)$$

From Eq. (1.3.12), we see that the solution of Eq. (1.3.9) is simply a transformation of the initial

state. Therefore, the unique matrix  $\Psi(k)$  is called the *state transition matrix*. It is also called the *the fundamental matrix*. The state transition matrix contains all the information about the free motions of the system define by Eq. (1.3.9).

In terms of the state transition matrix  $\Psi(k)$ , Eq. 1.3.7 can be written in the form

$$\begin{aligned}\vec{x}(k) &= \Psi(k)\vec{x}(0) + \sum_{j=0}^{k-1} \Psi(k-j-1)\mathbf{B}\vec{u}(j) \\ &= \Psi(k)\vec{x}(0) + \sum_{j=0}^{k-1} \Psi(j)\mathbf{B}\vec{u}(k-j-1)\end{aligned}\quad (1.3.13)$$

Substituting Eq. (1.3.13) into second equation of Eq. (1.3.4), we obtain:

$$\vec{y}(k) = \mathbf{C}\Psi(k)\vec{x}(0) + \mathbf{C}\sum_{j=0}^{k-1} \Psi(j)\mathbf{B}\vec{u}(k-j-1) \quad (1.3.14)$$

### 1.3.2 $\mathcal{Z}$ -Transform Approach to the Solution of Discrete-time State Equations.

We now present the solution of discrete-time state equation by the  $\mathcal{Z}$ -transform method. Consider the discrete-time system described by Eq. (1.3.15):

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \quad (1.3.15)$$

Taking the  $\mathcal{Z}$ -transform of both sides of Eq. (1.3.15) gives

$$z\vec{X}(z) - z\vec{x}(0) = \mathbf{A}\vec{X}(z) + \mathbf{B}\vec{U}(z) \quad (1.3.16)$$

where  $\vec{X}(z) = \mathcal{Z}[\vec{x}(k)]$  and  $\vec{U}(z) = \mathcal{Z}[\vec{u}(k)]$ . Then

$$(z\mathbf{I} - \mathbf{A})\vec{X}(z) = z\vec{x}(0) + \mathbf{B}\vec{U}(z) \quad (1.3.17)$$

Premultiplying both sides of this last equation by  $(z\mathbf{I} - \mathbf{A})^{-1}$ , we obtain

$$\vec{X}(z) = (z\mathbf{I} - \mathbf{A})^{-1}z\vec{x}(0) + (z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\vec{U}(z) \quad (1.3.18)$$

Taking the inverse  $\mathcal{Z}$ -transform ( $\mathcal{Z}^{-1}$ ) of both sides of Eq. (1.3.18) gives

$$\vec{x}(k) = \mathcal{Z}^{-1}\left[(z\mathbf{I} - \mathbf{A})^{-1}z\right]\vec{x}(0) + \mathcal{Z}^{-1}\left[(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\vec{U}(z)\right] \quad (1.3.19)$$

Comparing Eq. (1.3.7) with Eq. (1.3.19), we obtain

$$\mathbf{A}^k = \mathcal{Z}^{-1}\left[(z\mathbf{I} - \mathbf{A})^{-1}z\right] \quad (1.3.20)$$

and

$$\sum_{j=0}^{k-1} \mathbf{A}^{k-j-1} \mathbf{B} \vec{u}(j) = \mathcal{Z}^{-1} \left[ (z\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \vec{U}(z) \right] \quad (1.3.21)$$

where  $k = 1, 2, 3, \dots$

Notice that the solution by the  $z$  transform method involves the process of inverting the matrix  $(z\mathbf{I} - \mathbf{A})$ , accomplished by analytical means or by use of a computer routine. The solution also requires the inverse  $\mathcal{Z}$ -transform of  $(z\mathbf{I} - \mathbf{A})^{-1} z$  and  $(z\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \vec{U}(z)$ .

### 1.3.3 Non-Uniqueness of State Space Representation.

For a given pulse-transfer-function system the state space representation is not unique. We have demonstrated that different state space representation for a given pulse-transfer-function system are possible. The state equation, however, are related to each other by the similarity transformation.

Consider the system defined by

$$\begin{cases} \vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \\ \vec{y}(k) = \mathbf{C}\vec{x} \end{cases} \quad (1.3.22)$$

Let us define a new state vector  $\vec{z}$  by

$$\vec{x}(k) = \mathbf{T}\vec{z}(k) \quad (1.3.23)$$

where  $\mathbf{T}$  is a non-singular matrix. Then by substituting Eq. (1.3.23) into (1.3.22) we obtain

$$\mathbf{T}\vec{z}(k+1) = \mathbf{A}\mathbf{T}\vec{z}(k) + \mathbf{B}\vec{u}(k) \quad (1.3.24)$$

Left-multiplying both sides by  $\mathbf{T}^{-1}$  yields

$$\vec{z}(k+1) = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}\vec{z}(k) + \mathbf{T}^{-1}\mathbf{B}\vec{u}(k) \quad (1.3.25)$$

Let us define

$$\mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \mathbf{G}, \quad \mathbf{T}^{-1}\mathbf{B} = \mathbf{H}$$

Then Eq. (1.3.22) can be written as follows:

$$\vec{z}(k+1) = \mathbf{G}\vec{z}(k) + \mathbf{H}\vec{u}(k) \quad (1.3.26)$$

By defining

$$\mathbf{C}\mathbf{T} = \mathbf{M}$$

we can write:

$$\vec{y}(k) = \mathbf{M}\vec{z} \quad (1.3.27)$$

We have thus shown that the state space representation given by Eq. (1.3.22) is equivalent to the state representation given by Eq. (1.3.26) and Eq. (1.3.27)

$$\begin{cases} \vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \\ \vec{y}(k) = \mathbf{C}\vec{x} \end{cases} \iff \begin{cases} \vec{z}(k+1) = \mathbf{G}\vec{z}(k) + \mathbf{H}\vec{u}(k) \\ \vec{y}(k) = \mathbf{M}\vec{z} \end{cases} \quad (1.3.28)$$

## 1.4 Canonical State Space Representation

There are many techniques available for obtaining state space representations of discrete-time systems. Consider the discrete-time system described by the following ARMA model

$$\begin{aligned} y(k) + a_{n-1}y(k-1) + a_{n-2}y(k-2) + \cdots + a_0y(k-n) &= \\ &= b_n u(k) + b_{n-1}u(k-1) + \cdots + b_0u(k-n) \end{aligned} \quad (1.4.1)$$

where  $u(k)$  is the input and  $y(k)$  is the output of the system at the  $k$ th sampling instant. Note that some of the coefficients  $a_i$  ( $i = 1, 2, \dots, n$ ) and  $b_j$  ( $j = 1, 2, \dots, n$ ) may be zero. Eq. 1.4.1 can be written in the form of the pulse transfer function as

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_n z^n + b_{n-1} z^{n-1} + \cdots + b_1 z + b_0}{z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0} \quad (1.4.2)$$

or

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_n + b_{n-1} z^{-1} + \cdots + b_1 z^{-(n-1)} + b_0 z^{-n}}{1 + a_{n-1} z^{-1} + \cdots + a_1 z^{-(n-1)} + a_0 z^{-n}} \quad (1.4.3)$$

### 1.4.1 Controllable Canonical Form.

The state space representation of the transfer function system given by Eq. (1.4.1) in the controllable canonical form is given by the following equations:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \quad (1.4.4)$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (1.4.5)$$

$$\mathbf{C} = \begin{bmatrix} b_0 - a_0 b_n, & b_1 - a_1 b_n, & \dots, & b_{n-1} - a_{n-1} b_n \end{bmatrix} \quad (1.4.6)$$

$$\mathbf{D} = b_n \quad (1.4.7)$$

### 1.4.2 Partial-Fraction-Expansion Programming Method.

The programming method using partial fraction expansion applies when the denominator of the pulse transfer function is in a factored form. Consider the system defined by

$$\begin{aligned} G(z) &= \frac{Y(z)}{U(z)} = \frac{b_n z^n + b_{n-1} z^{n-1} + \dots + b_1 z + b_0}{z^n + a_{n-1} z^{n-1} + \dots + a_1 z + a_0} = \\ &= b_n + \frac{(b_{n-1} - a_{n-1} b_n) z^{n-1} + (b_{n-2} - a_{n-2} b_n) z^{n-2} + \dots + (b_0 - a_0 b_n)}{(z - p_{n-1})(z - p_{n-2}) \dots (z - p_0)} \end{aligned} \quad (1.4.8)$$

we shall first discuss the case where all poles are distinct. Then we shall consider the case where multiple poles are involved.

**Case 1:  $G(z)$  Involves Distinct Poles Only.** In this case Eq. (1.4.8) can be expanded into partial fractions as follows:

$$G(z) = \frac{Y(z)}{U(z)} = b_n + \frac{c_{n-1}}{z - p_{n-1}} + \frac{c_{n-2}}{z - p_{n-2}} + \dots + \frac{c_0}{z - p_0} \quad (1.4.9)$$

where

$$c_i = \lim_{z \rightarrow p_i} \left[ \frac{Y(z)}{X(z)} (z - p_i) \right] \quad (1.4.10)$$

Equation (10.9) can be written in the form

$$Y(z) = b_n U(z) + \frac{c_{n-1}}{z - p_{n-1}} U(z) + \frac{c_{n-2}}{z - p_{n-2}} U(z) + \dots + \frac{c_0}{z - p_0} U(z) \quad (1.4.11)$$

Let us define the state variables as follows:

$$\begin{aligned} X_1(z) &= \frac{1}{z - p_1} U(z) \\ X_2(z) &= \frac{1}{z - p_2} U(z) \\ X_3(z) &= \frac{1}{z - p_3} U(z) \\ &\vdots \\ X_n(z) &= \frac{1}{z - p_n} U(z) \end{aligned} \quad (1.4.12)$$

then can be rewritten as

$$\begin{aligned} zX_1(z) &= p_1X_1 + U(z) \\ zX_2(z) &= p_2X_2 + U(z) \\ &\vdots \\ zX_n(z) &= p_nX_n + U(z) \end{aligned} \tag{1.4.13}$$

Also, Eq. (1.4.11) can be written as

$$Y(z) = b_nU(z) + c_1X_1(z) + c_2X_2(z) + \cdots + c_nX_n(z) \tag{1.4.14}$$

The inverse  $\mathcal{Z}$  transformation of Eqs. (1.4.13)) and (1.4.14) gives

$$\begin{aligned} x_1(k+1) &= p_1x_1(k) + u(k) \\ x_2(k+1) &= p_2x_2(k) + u(k) \\ &\vdots \\ x_n(k+1) &= p_nx_n(k) + u(k) \end{aligned} \tag{1.4.15}$$

and

$$y(k) = c_1x_1(k) + c_2x_2(k) + \cdots + c_nx_n(k) + b_0u(k) \tag{1.4.16}$$

Rewriting the state equation and the output equation in the form of vector-matrix equations, we obtain

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ 0 & p_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & p_n \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} u(k) \tag{1.4.17}$$

and

$$y(k) = \begin{bmatrix} c_1 & c_2 & \cdots & c_n \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{bmatrix} + b_0u(k) \tag{1.4.18}$$

Notice that the  $n \times n$  state matrix is diagonal matrix. The diagonal elements are the poles of  $G(z) = Y(z)/X(z)$ .

**Case 2:  $G(z)$  Involves Multiple Poles.** In the discussion that follows, we assume that  $G(z)$  involves a multiple pole of order  $m$  at  $z = p_1$  and that all other poles are distinct. Consider the

system defined by Eq. (1.4.8). Since  $G(z)$  can be written in the form

$$\begin{aligned}
 G(z) = \frac{Y(z)}{U(z)} &= \frac{b_n z^n + b_{n-1} z^{n-1} + \cdots + b_1 z + b_0}{(z - p_1)^m (z - p_{m+1})(z - p_{m+2}) \cdots (z - p_n)} = \\
 &= b_n + \frac{(b_{n-1} - a_{n-1} b_n) z^{n-1} + (b_{n-2} - a_{n-2} b_n) z^{n-2} + \cdots + (b_0 - a_0 b_n)}{(z - p_1)^m (z - p_{m+1})(z - p_{m+2}) \cdots (z - p_n)} = \\
 &= b_n + \frac{c_1}{(z - p_1)^m} + \frac{c_2}{(z - p_2)^{m-1}} + \cdots + \frac{c_m}{z - p_1} + \\
 &\quad + \frac{c_{m+1}}{z - p_{m+1}} + \frac{c_{m+2}}{z - p_{m+2}} + \cdots + \frac{c_n}{z - p_n}
 \end{aligned} \tag{1.4.19}$$

we obtain

$$\begin{aligned}
 Y(z) &= b_n U(z) + \frac{c_1}{(z - p_1)^m} U(z) + \frac{c_2}{(z - p_2)^{m-1}} U(z) + \cdots + \frac{c_m}{z - p_1} U(z) \\
 &\quad + \frac{c_{m+1}}{z - p_{m+1}} U(z) + \frac{c_{m+2}}{z - p_{m+2}} U(z) + \cdots + \frac{c_n}{z - p_n} U(z)
 \end{aligned} \tag{1.4.20}$$

Let us define the first  $m$  state variables  $X_1(z), X_2(z), \dots, X_m(z)$  by the equations:

$$\begin{aligned}
 X_1(z) &= \frac{1}{(z - p_1)^m} U(z) \\
 X_2(z) &= \frac{1}{(z - p_1)^{m-1}} U(z) \\
 X_3(z) &= \frac{1}{(z - p_1)^{m-2}} U(z) \\
 &\vdots \\
 X_m(z) &= \frac{1}{z - p_1} U(z)
 \end{aligned} \tag{1.4.21}$$

and the remaining  $n - m$  state variables  $X_{m+1}(z), X_{m+2}(z), \dots, X_n(z)$  by the equations

$$\begin{aligned}
 X_{m+1}(z) &= \frac{1}{z - p_{m+1}} U(z) \\
 X_{m+2}(z) &= \frac{1}{z - p_{m+2}} U(z) \\
 X_{m+3}(z) &= \frac{1}{z - p_{m+3}} U(z) \\
 &\vdots \\
 X_n(z) &= \frac{1}{z - p_n} U(z)
 \end{aligned} \tag{1.4.22}$$

Notice that the  $m$  state variables defined by Eq. (1.4.22) are related each to the next by the following equations:

$$\begin{aligned}\frac{X_1(z)}{X_2(z)} &= \frac{1}{z - p_1} \\ \frac{X_2(z)}{X_3(z)} &= \frac{1}{z - p_1} \\ &\vdots \\ \frac{X_{m-1}(z)}{X_m(z)} &= \frac{1}{z - p_1}\end{aligned}\tag{1.4.23}$$

By taking the inverse  $\mathcal{Z}$  transforms of all of Eq. (10.9), the last equation in Eq. (10.9), and all of Eq. (10.9), we obtain

$$\begin{aligned}x_1(k+1) &= p_1 x_1(k) + x_2(k) \\ x_2(k+1) &= p_1 x_2(k) + x_3(k) \\ &\vdots \\ x_{m-1}(k+1) &= p_1 x_{m-1}(k) + x_m(k) \\ x_m(k+1) &= p_1 x_m(k) + u(k) \\ x_{m+1}(k+1) &= p_{m+1} x_{m+1}(k) + u(k) \\ &\vdots \\ x_n(k+1) &= p_n x_n(k) + u(k)\end{aligned}\tag{1.4.24}$$

The output equation given by Eq. (10.9) can be rewritten as follows:

$$\begin{aligned}Y(z) &= c_1 X_1(z) + c_2 X_2(z) + \cdots + c_m X_m(z) + c_{m+1} X_{m+1}(z) \\ &\quad + c_{m+2} X_{m+2}(z) + \cdots + c_n X_n(z) + b_0 U(z)\end{aligned}$$

By taking the inverse  $\mathcal{Z}$  transform of this last equation, we get

$$\begin{aligned}y(k) &= c_1 x_1(k) + c_2 x_2(k) + \cdots + c_m x_m(k) + c_{m+1} x_{m+1}(k) \\ &\quad + c_{m+2} x_{m+2}(k) + \cdots + c_n x_n(k) + b_0 u(k)\end{aligned}\tag{1.4.25}$$

Rewriting the state equation ad the output equation in the vector-matrix form, we obtain

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ x_m(k+1) \\ \hline x_{m+1}(k+1) \\ \vdots \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} p_1 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & p_1 & 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & p_1 & 0 & \cdots & 0 \\ \hline 0 & 0 & 0 & \cdots & 0 & p_{m+1} & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & p_n \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_m(k) \\ \hline x_{m+1}(k) \\ \vdots \\ x_n(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \hline 1 \\ \vdots \\ 1 \end{bmatrix} u(k) \quad (1.4.26)$$

and

$$y(k) = \begin{bmatrix} c_1 & c_2 & \cdots & c_n \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{bmatrix} + b_0 u(k) \quad (1.4.27)$$

The  $n \times n$  state matrix in this case is in a Jordan canonical form.

## 1.5 Discretization of Continuous-time State Space System

The digital control of continuous time systems required to convert continuous time state space equations into discrete-time state space equations. Such conversion can be done by introducing samplers and holding into the continuous-time systems. The error introduced by discretization may be made negligible by using a sufficiently small sampling period compared with significant time constant of the system. **The discrete time solutions must be valid at equally spaced sampling instants.**

In what follows we shall present a procedure for discretizing continuous-time state space equations. **We assume that the input  $u(t)$  changes only at equally spaced sampling instants.**

Note that the sampling operation here is fictitious. We shall derive the discrete-time state equation and output equation which yield the exact value at  $t = kt_s$ , where  $k = 0, 1, 2, \dots$

Consider the system of Eq. (1.5.1)

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \quad (1.5.1)$$

and we derive the solution, as follows

$$e^{-\tilde{\mathbf{A}}t} \left[ \dot{\vec{x}}(t) - \tilde{\mathbf{A}}\vec{x}(t) \right] = e^{-\tilde{\mathbf{A}}t} \tilde{\mathbf{B}}u(t) \quad (1.5.2)$$

$$\frac{d}{dt} \left[ e^{-\tilde{\mathbf{A}}t} \vec{x}(t) \right] = -\tilde{\mathbf{A}}e^{-\tilde{\mathbf{A}}t} \vec{x}(t) + e^{-\tilde{\mathbf{A}}t} \dot{\vec{x}}(t) \quad (1.5.3)$$

$$\frac{d}{dt} \left[ e^{-\tilde{\mathbf{A}}t} \vec{x}(t) \right] = e^{-\tilde{\mathbf{A}}t} \tilde{\mathbf{B}}u(t) \quad (1.5.4)$$

$$\int_0^t d \left[ e^{-\tilde{\mathbf{A}}\tau} \vec{x}(\tau) \right] = \int_0^t e^{-\tilde{\mathbf{A}}\tau} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.5)$$

$$e^{-\tilde{\mathbf{A}}t} \vec{x}(t) = \vec{x}(0) + \int_0^t e^{-\tilde{\mathbf{A}}\tau} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.6)$$

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) + e^{\tilde{\mathbf{A}}t} \int_0^t e^{-\tilde{\mathbf{A}}\tau} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.7)$$

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) + \int_0^t e^{\tilde{\mathbf{A}}(t-\tau)} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.8)$$

Now we computing Eq. 1.5.8 for  $t = kt_s$  and for  $t = (k+1)t_s$

$$\vec{x}[(k+1)t_s] = e^{\tilde{\mathbf{A}}(k+1)t_s} \vec{x}(0) + \int_0^{(k+1)t_s} e^{\tilde{\mathbf{A}}[(k+1)t_s-\tau]} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.9)$$

$$\vec{x}[kt_s] = e^{\tilde{\mathbf{A}}kt_s} \vec{x}(0) + \int_0^{kt_s} e^{\tilde{\mathbf{A}}[kt_s-\tau]} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.10)$$

Left-multiplying the Eq. (1.5.10) by  $e^{\tilde{\mathbf{A}}t_s}$  and subtracting from Eq. (1.5.9) we obtain

$$\vec{x}[(k+1)t_s] = e^{\tilde{\mathbf{A}}t_s} \vec{x}[kt_s] + \int_{kt_s}^{(k+1)t_s} e^{\tilde{\mathbf{A}}[(k+1)t_s-\tau]} \tilde{\mathbf{B}}u(\tau) d\tau \quad (1.5.11)$$

Introducing the new variable  $(k+1)t_s - \tau = \lambda$  and considering that  $u(\tau) = u(kt_s)$  is constant during the time step (in fact, input changes its value only at discrete time instant  $kt_s$ ), Eq. 1.5.11 can be written as follows

$$\vec{x}[(k+1)t_s] = e^{\tilde{\mathbf{A}}t_s} \vec{x}[kt_s] + \left[ \int_0^{t_s} e^{\tilde{\mathbf{A}}\lambda} \tilde{\mathbf{B}} d\lambda \right] u[kt_s] \quad (1.5.12)$$

where

$$[(k+1)t_s - \tau] = \lambda \Rightarrow \begin{cases} \tau = kt_s \Rightarrow \lambda = t_s \\ \tau = (k+1)t_s \Rightarrow \lambda = 0 \\ d\tau = -d\lambda \end{cases} \quad (1.5.13)$$

Comparing Eq. (1.3.4) here reported

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \quad (1.5.14)$$

with Eq. (1.5.12), we observe that

$$\mathbf{A} = e^{\tilde{\mathbf{A}}t_s} \quad (1.5.15)$$

$$\mathbf{B} = \int_0^{t_s} e^{\tilde{\mathbf{A}}t} \tilde{\mathbf{B}} dt \quad (1.5.16)$$

Matrix  $\mathbf{A}$  can be computed as follows

$$\mathbf{A} = e^{\tilde{\mathbf{A}}t_s} = \mathcal{L}^{-1} \left[ \left( s\mathbf{I} - \tilde{\mathbf{A}} \right)^{-1} \right] \Big|_{t=t_s} \quad (1.5.17)$$

Matrix  $\mathbf{B}$  can be computed as follows

$$\mathbf{B} = \int_0^{t_s} \left( \mathbf{I} + \tilde{\mathbf{A}}t + \tilde{\mathbf{A}}^2 \frac{t^2}{2!} + \dots \right) \tilde{\mathbf{B}} dt \quad (1.5.18)$$

or if  $\tilde{\mathbf{A}}$  is not singular,

$$\mathbf{B} = \tilde{\mathbf{A}}^{-1} \left( \mathbf{I} + \tilde{\mathbf{A}}t_s + \tilde{\mathbf{A}}^2 \frac{t_s^2}{2!} + \dots - \mathbf{I} \right) \tilde{\mathbf{B}} = \tilde{\mathbf{A}}^{-1} \left( e^{\tilde{\mathbf{A}}t_s} - \mathbf{I} \right) \tilde{\mathbf{B}} \quad (1.5.19)$$

**Some considerations:** from Eq. (1.5.15) and Eq. (1.5.19) we can conclude that to discretize a dynamical system represented in state space form it is, first, necessary to compute the matrix exponential  $e^{\tilde{\mathbf{A}}t_s}$  and moreover

$$\begin{cases} \mathbf{A} = e^{\tilde{\mathbf{A}}t_s} \\ \mathbf{B} = \tilde{\mathbf{A}}^{-1} \left( e^{\tilde{\mathbf{A}}t_s} - \mathbf{I} \right) \tilde{\mathbf{B}} \end{cases} \quad (1.5.20)$$

or is possible to apply Taylor expansion as follows

$$\begin{cases} \mathbf{A} = \mathbf{I} + \tilde{\mathbf{A}}t_s + \frac{1}{2!} \tilde{\mathbf{A}}^2 t_s^2 + \frac{1}{3!} \tilde{\mathbf{A}}^3 t_s^3 + \dots \\ \mathbf{B} = \tilde{\mathbf{B}}t_s + \frac{1}{2!} \tilde{\mathbf{A}}\tilde{\mathbf{B}}t_s^2 + \frac{1}{3!} \tilde{\mathbf{A}}^2\tilde{\mathbf{B}}t_s^3 + \dots \end{cases} \quad (1.5.21)$$

or approximate as well

$$\begin{cases} \mathbf{A} \approx \mathbf{I} + \tilde{\mathbf{A}}t_s \\ \mathbf{B} \approx \tilde{\mathbf{B}}t_s \end{cases} \quad (1.5.22)$$

Approximation of Eq. (1.5.22) can introduce some uncertainties in the discretized model that could be not acceptable. In this case additional terms of the Taylor approximation can be add according to Eq. (1.5.21).

### Example 1.9. (Discretization of a Proportional-Resonant (PR) controller)

The **Proportional-Resonant** (PR) controller permits to track a given sinusoidal reference with zero-error, at steady state, for a given frequency:

$$\text{PR}(s) = k_p + k_i \quad \text{RES}(s) = k_p + k_i \frac{s}{s^2 + \omega_0^2} \quad (1.5.23)$$

where

$$\text{RES}(s) = \frac{s}{s^2 + \omega_0^2} \quad (1.5.24)$$

The digital implementation of the **PR** control requires some consideration regarding sampling time  $t_s$  and discretization method. In the following we show three different discretization methods, in particular, two from  $s$ -plane to  $z$ -plane and the last one directly from state space representation.

Controllers are then evaluated supposing the following plant:  $\frac{1}{sL+R}$  ( $R = 10 \text{ m}\Omega$ ,  $L = 1 \text{ mH}$ ) and the following sampling time  $t_s = 100 \mu\text{s}$ . Proportional and integral gains ( $k_p$  and  $k_i$ ) are kept small in order to amplify the effects of the resonant internal model.

#### Discretization via Backward method:

$$s = \frac{1 - z^{-1}}{t_s} = \frac{1}{t_s} \frac{z - 1}{z}$$

The equivalent  $z$ -domain control transfer function becomes:

$$\text{RES}(z) = \text{RES}(s)|_{s=\frac{1-z^{-1}}{t_s}} = t_s \frac{z(z-1)}{(1+\omega_0^2 t_s^2)z^2 - 2z + 1} \quad (1.5.25)$$

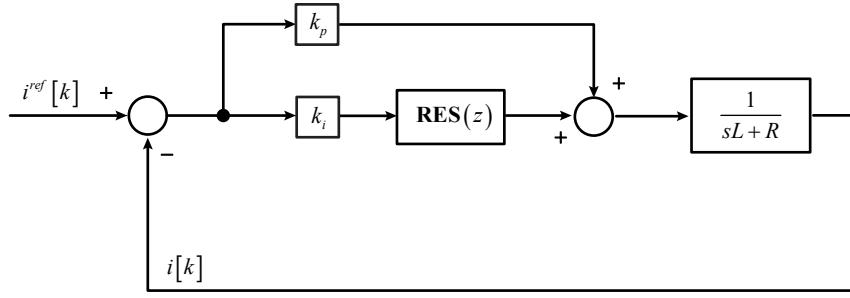


Figure 1.14: Proportional Resonant Controller structure.

The resulting performance are here shown:

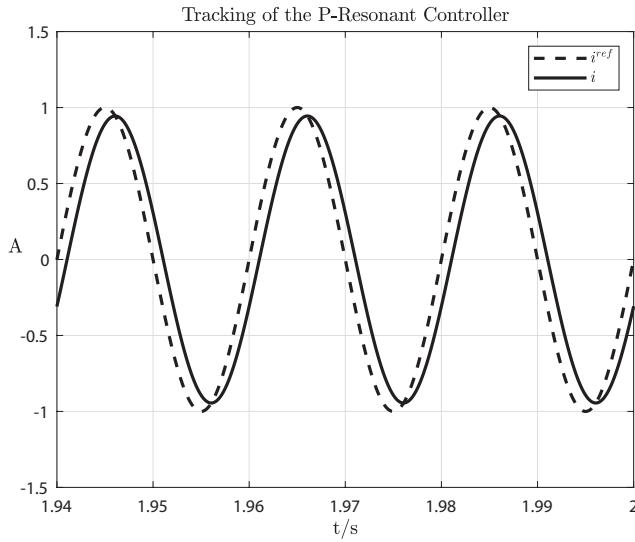


Figure 1.15: Proportional Resonant Control performance in the case of backward discretization.  
 $i^{ref}(t)$ -dashed,  $i(t)$ -solid.

As shown in the picture 1.15 backward discretization degrade considerably the performance of the resonant model.

#### Discretization via pre-warped trapezoidal method:

$$s = \frac{\omega_0}{\tan(\frac{\omega_0 t_s}{2})} \frac{z - 1}{z + 1}$$

The equivalent  $z$ -domain control transfer function becomes:

$$\begin{aligned} \mathbf{RES}(z) &= \frac{k_T(z^2 - 1)}{(k_T^2 + \omega_0^2)z^2 - 2(k_T^2 - \omega_0^2)z + k_T^2 + \omega_0^2} \\ k_T &= \frac{\omega_0}{\tan(\frac{\omega_0 t_s}{2})} \end{aligned} \quad (1.5.26)$$

The resulting performance are here shown:

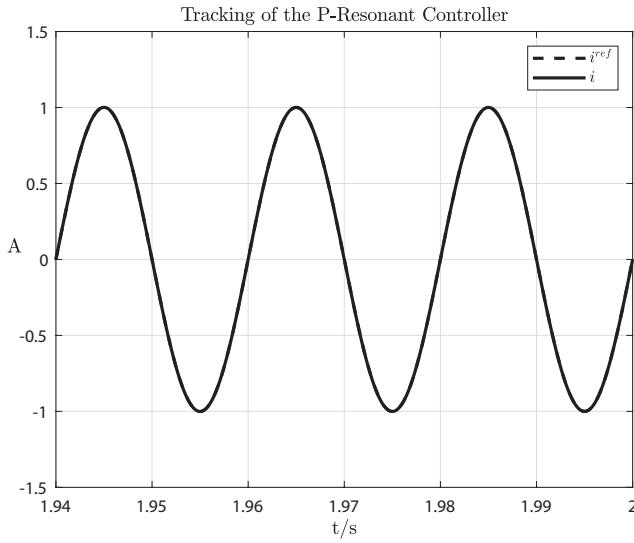


Figure 1.16: Proportional Resonant Control performance in the case of pre-warped trapezoidal discretization.  $i^{ref}(t)$ -dashed,  $i(t)$ -solid.

#### Discretization from state space representation:

The digital controller like PI and others can be implemented, for different reasons, in state space form. In the following, the state space representation of the **PR** controller is shown as well as its discretization via inverse Laplace transform.

The state space representation of the transfer function

$$\mathbf{RES}(s) = \frac{s}{s^2 + \omega_0^2} \quad (1.5.27)$$

can be written as follows, using the controllable canonical representation<sup>¶</sup>

$$\begin{aligned}\dot{\vec{x}}(t) &= \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \\ \vec{y}(t) &= \mathbf{C}\vec{x}(t)\end{aligned}\quad (1.5.33)$$

or

$$\begin{aligned}\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \\ y &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\end{aligned}\quad (1.5.34)$$

where

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & 0 \end{bmatrix} \quad (1.5.35)$$

and

$$\tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (1.5.36)$$

In order to implement the controller in digital form we have to discretize it in order to achieve the form

$$\vec{x}[(k+1)t_s] = \mathbf{A}\vec{x}(kt_s) + \mathbf{B}u(kt_s) \quad (1.5.37)$$

---

<sup>¶</sup>Given the following transfer function

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_n s^n + b_{n-1} s^{n-1} + \dots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} \quad (1.5.28)$$

as follows the corresponding controllable canonical state space representation

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{bmatrix} \quad (1.5.29)$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (1.5.30)$$

$$\mathbf{C} = [b_0 - a_0 b_n, \quad b_1 - a_1 b_n, \quad \dots, \quad b_{n-1} - a_{n-1} b_n] \quad (1.5.31)$$

$$\mathbf{D} = b_n \quad (1.5.32)$$

Where  $\mathbf{A}$  and  $\mathbf{B}$  can be calculated as follows

$$\mathbf{A} = e^{\tilde{\mathbf{A}}t_s}$$

and

$$\mathbf{B} = \int_0^{t_s} e^{\tilde{\mathbf{A}}\lambda} \tilde{\mathbf{B}} d\lambda \quad (1.5.38)$$

Hence

$$e^{\tilde{\mathbf{A}}t} = \mathcal{L}^{-1} \left[ (s\mathbf{I} - \tilde{\mathbf{A}})^{-1} \right] = \quad (1.5.39)$$

$$(s\mathbf{I} - \tilde{\mathbf{A}}) = \begin{bmatrix} s & -1 \\ \omega_0^2 & s \end{bmatrix} \quad (1.5.40)$$

$$(s\mathbf{I} - \tilde{\mathbf{A}})^{-1} = \frac{1}{s^2 + \omega_0^2} \begin{bmatrix} s & 1 \\ -\omega_0^2 & s \end{bmatrix} \xrightarrow{\mathcal{L}^{-1}} \begin{bmatrix} \cos(\omega_0 t) & \frac{\sin(\omega_0 t)}{\omega_0} \\ -\omega_0 \sin(\omega_0 t) & \cos(\omega_0 t) \end{bmatrix}_{t=t_s} = \quad (1.5.41)$$

$$= \begin{bmatrix} \cos(\omega_0 t_s) & \frac{\sin(\omega_0 t_s)}{\omega_0} \\ -\omega_0 \sin(\omega_0 t_s) & \cos(\omega_0 t_s) \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} \cos(\omega_0 t_s) & \frac{\sin(\omega_0 t_s)}{\omega_0} \\ -\omega_0 \sin(\omega_0 t_s) & \cos(\omega_0 t_s) \end{bmatrix} \quad (1.5.42)$$

$$\mathbf{B} = \int_0^{t_s} \begin{bmatrix} \cos(\omega_0 \tau) & \frac{\sin(\omega_0 \tau)}{\omega_0} \\ -\omega_0 \sin(\omega_0 \tau) & \cos(\omega_0 \tau) \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} d\tau =$$

$$= \begin{bmatrix} -\frac{\cos(\omega_0 t)}{\omega_0^2} \\ \frac{\sin \omega_0 t}{\omega_0} \end{bmatrix}_0^{t_s} = \begin{bmatrix} \frac{1-\cos(\omega_0 t_s)}{\omega_0^2} \\ \frac{\sin \omega_0 t_s}{\omega_0} \end{bmatrix} \quad (1.5.43)$$

Here a possible implementation of the controller in state space form.

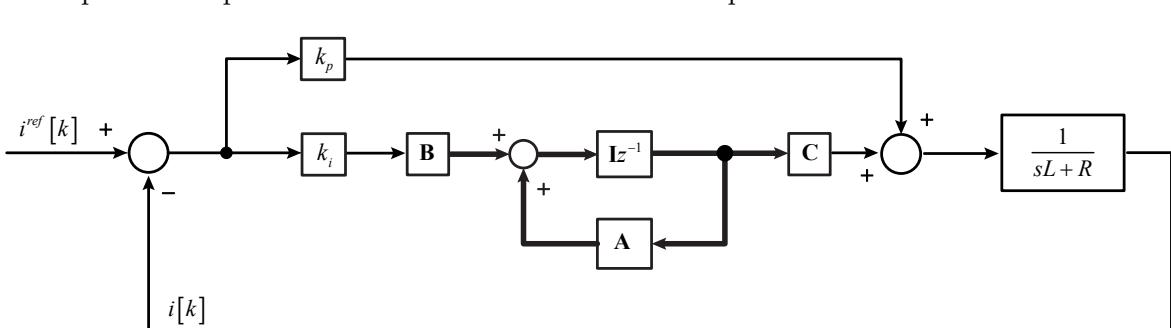


Figure 1.17: Implementation of the proportional resonant control in the case of discrete time state space representation.

The resulting performance are shown in Figure 1.18.

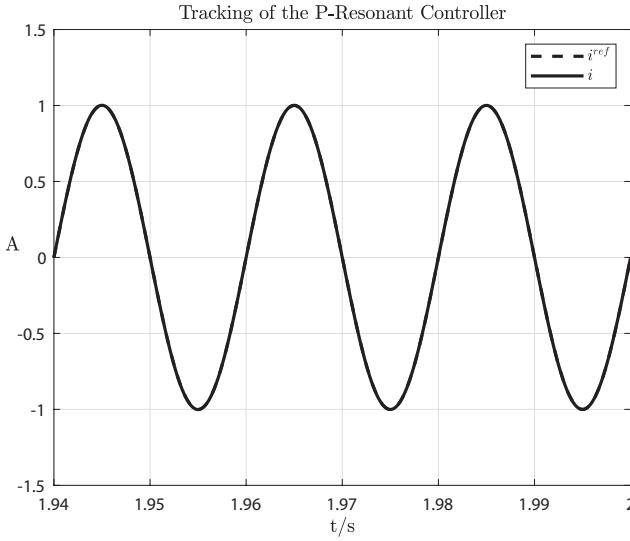


Figure 1.18: Proportional resonant control performance in the case of discrete time state space implementation.  $i^{ref}(t)$ -dashed,  $i(t)$ -solid.

◁

## 1.6 Controllability

Consider the continuous time system

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \quad (1.6.1)$$

where

- $\vec{x}$  = state vector ( $n$ -vector)
- $u$  = control input (scalar)
- $\tilde{\mathbf{A}} = n \times n$  matrix
- $\tilde{\mathbf{B}} = n \times 1$  matrix

the system described by Eq. (1.6.1) is said to be controllable at  $t = t_0$  if it is possible to construct an unconstrained control input  $u(t)$  that will transfer an initial state to any final state in a finite time interval  $t_0 \leq t \leq t_1$ . If every state is controllable, then the system is said to be completely state controllable.

We shall now derive the condition for complete state controllability. Without loss of generality, we can assume that the final state is the origin of the state space and that the initial time is zero, or  $t_0 = 0$ .

The solution of Eq. (1.6.1) is

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) + \int_0^t e^{\tilde{\mathbf{A}}(t-\tau)} \tilde{\mathbf{B}} u(\tau) d\tau \quad (1.6.2)$$

Applying the definition of complete state controllability just given, we have

$$\vec{x}(t_1) = \vec{0} = e^{\tilde{\mathbf{A}}t_1} \vec{x}(0) + \int_0^{t_1} e^{\tilde{\mathbf{A}}(t_1-\tau)} \tilde{\mathbf{B}} u(\tau) d\tau \quad (1.6.3)$$

or

$$\vec{x}(0) = - \int_0^{t_1} e^{-\tilde{\mathbf{A}}\tau} \tilde{\mathbf{B}} u(\tau) d\tau \quad (1.6.4)$$

where  $e^{-\tilde{\mathbf{A}}\tau}$  can be written as

$$e^{-\tilde{\mathbf{A}}\tau} = \sum_{k=0}^{n-1} \alpha_k(\tau) \tilde{\mathbf{A}}^k \quad (1.6.5)$$

Substituting Eq. (1.6.5) into Eq. (1.6.4) gives

$$\vec{x}(0) = - \sum_{k=0}^{n-1} \tilde{\mathbf{A}}^k \tilde{\mathbf{B}} \int_0^{t_1} \alpha_k(\tau) u(\tau) d\tau \quad (1.6.6)$$

Let us put

$$\beta_k = \int_0^{t_1} \alpha_k(\tau) u(\tau) d\tau \quad (1.6.7)$$

The Eq. (1.6.6) becomes

$$\begin{aligned} \vec{x}(0) &= - \sum_{k=0}^{n-1} \tilde{\mathbf{A}}^k \tilde{\mathbf{B}} \beta_k \\ &= - \left[ \tilde{\mathbf{B}} \quad \tilde{\mathbf{A}}\tilde{\mathbf{B}} \quad \dots \quad \tilde{\mathbf{A}}^{n-1}\tilde{\mathbf{B}} \right] \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{n-1} \end{bmatrix} \end{aligned} \quad (1.6.8)$$

If the system is completely state controllable, then, given any initial state  $\vec{x}(0)$ , Eq. (1.6.8) must be satisfied. This requires that the rank of the  $n \times n$  matrix be  $n$ .

$$\mathcal{C} = \left[ \tilde{\mathbf{B}} \quad \tilde{\mathbf{A}}\tilde{\mathbf{B}} \quad \dots \quad \tilde{\mathbf{A}}^{n-1}\tilde{\mathbf{B}} \right] \quad (1.6.9)$$

The matrix  $\mathcal{C}$  is called the *controllability matrix*.

From this analysis, we can state the condition for complete state controllability as follows: the system given by Eq. (1.6.1) is completely state controllable if and only if the vectors  $\tilde{\mathbf{B}}$ ,

$\tilde{\mathbf{A}}\tilde{\mathbf{B}}, \dots, \tilde{\mathbf{A}}^{n-1}\tilde{\mathbf{B}}$  are linearly independent, or the  $n \times n$  matrix

$$\mathcal{C} = \begin{bmatrix} \tilde{\mathbf{B}} & \tilde{\mathbf{A}}\tilde{\mathbf{B}} & \dots & \tilde{\mathbf{A}}^{n-1}\tilde{\mathbf{B}} \end{bmatrix}$$

be of rank  $n$ .

## 1.7 State Feedback Control

In this section we shall present a design method commonly called *state feedback* or *pole-placement*. We assume that all state variables are measurable, and are available for feedback. It will be shown that if the system considered is completely state controllable, then poles of closed loop system may be placed at any desired locations by means of state feedback through an appropriate state feedback gain matrix.

Let us assume that we decide that the desired closed-loop poles are to be at

$$s_1 = \mu_1, s_2 = \mu_2, \dots s_n = \mu_n. \quad (1.7.1)$$

By choosing an appropriate gain matrix for state feedback, it is possible to force the system to have closed loop poles at the desired locations, provided that the original system is completely state controllable.

Consider the continuous time system

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \quad (1.7.2)$$

$$y(t) = \mathbf{C}\vec{x}(t) \quad (1.7.3)$$

where

- $\vec{x}$  = state vector ( $n$ -vector)
- $u$  = control input (scalar)
- $y$  = system output (scalar)
- $\tilde{\mathbf{A}} = n \times n$  matrix
- $\tilde{\mathbf{B}} = n \times 1$  matrix
- $\mathbf{C} = 1 \times n$  matrix

we shall choose the control input as follows

$$u(t) = -\mathbf{K}\vec{x}(t) \quad (1.7.4)$$

This means that the control input  $u$  is determined by an instantaneous state. Such a scheme is called state feedback. The  $1 \times n$  matrix  $\mathbf{K}$  is called the state feedback gain matrix. We assume

that all state variables are available for feedback. In the following analysis we assume that  $u$  is unconstrained. A block diagram for this system is shown in Figure 1.19.

This closed loop system has no input. Its objective is to maintain the zero output. Because of the disturbance that may be present, the output will deviate from zero. The non zero output will be returned to the zero reference input because of the state feedback scheme of the system. Moreover in case the system starts from a non equilibrium point, the system will move toward the equilibrium point with a dynamic which is given by the selected closed loop poles.

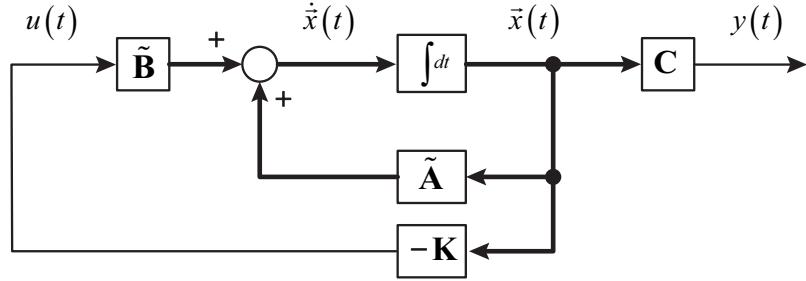


Figure 1.19: Closed loop control system with  $u = -\mathbf{K}\vec{x}$ .

Substituting Eq. (1.7.4) into Eq. (1.7.2) gives

$$\dot{\vec{x}}(t) = (\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K})\vec{x}(t) \quad (1.7.5)$$

The solution of this equation is given by

$$\vec{x}(t) = e^{(\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K})t} \vec{x}(0) \quad (1.7.6)$$

where  $\vec{x}(0)$  is the initial state caused by external disturbances. The stability and transient response characteristics are determined by the eigenvalues of the matrix  $\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K}$ . If matrix  $\mathbf{K}$  is chosen properly, the matrix  $\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K}$  can be an asymptotically stable matrix, and for all  $\vec{x}(0) \neq \vec{0}$ ,  $\vec{x}(t)$  approaches to  $\vec{0}$  as  $t$  approaches infinity. The eigenvalues of matrix  $\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K}$  are called the regulator poles. If these regulator poles are placed in the left half  $s$  plane, then  $\vec{x}(t)$  approaches  $\vec{0}$  as  $t$  approaches infinity. The problem of placing the regulator poles (closed loop poles) at the desired location is called a pole placement problem.

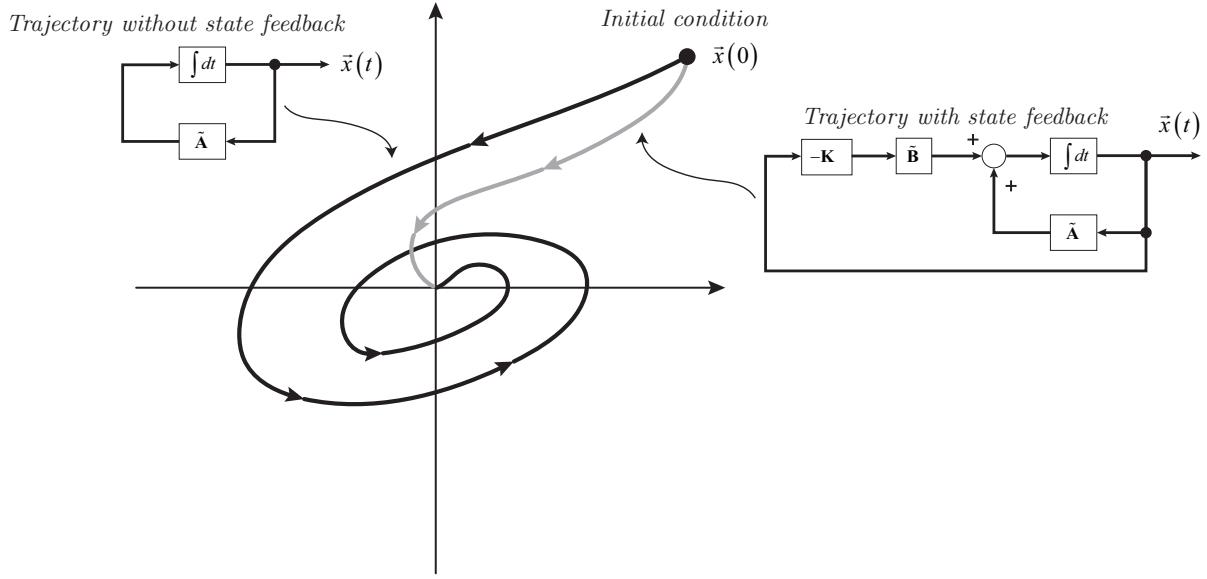


Figure 1.20: State feedback evolution for a generic asymptotically stable system. The effect of the state feedback control is to change the *nature* of the system. The new system will behave, in closed loop, as designed. The design of the state feedback matrix  $\mathbf{K}$  is here called: *regulator problem*.

**Example 1.10.** Let's consider the following plant

$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = u(t) \quad (1.7.7)$$

as shown in Figure 1.21, where  $m$  [kg] is the mass,  $b$  [ $\text{Ns m}^{-1}$ ] is the viscosity and  $k$  [ $\text{N m}^{-1}$ ] is the spring coefficient.

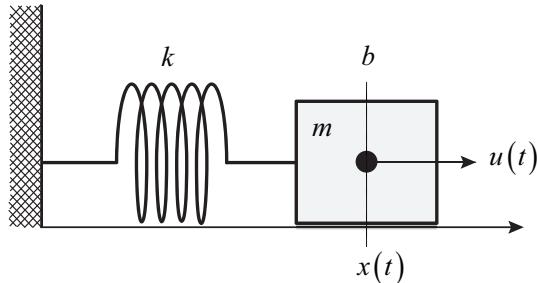


Figure 1.21: Spring mass system with force input  $u(t)$ .

Eq. (1.7.7) can be written in state space form as follows

$$\begin{aligned} \dot{x}(t) &= v(t) \\ \dot{v}(t) &= -\frac{k}{m}x(t) - \frac{b}{m}v(t) + \frac{1}{m}u(t) \end{aligned} \quad (1.7.8)$$

or

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x} + \tilde{\mathbf{B}}x(t) \quad (1.7.9)$$

where

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} \quad (1.7.10)$$

supposing  $k = 1 \text{ N m}^{-1}$ ,  $b = 0 \text{ N s m}^{-1}$  and  $m = 1 \text{ kg}$  we obtain

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (1.7.11)$$

where the natural evolution of the system can be represented by the phase portrait of Figure 1.22a<sup>||</sup>.

Applying the state feedback to the system of Eq, (1.7.11) where matrix  $\mathbf{K}$  is computed by pole placement suppose the equivalent system has two poles as  $p_1 = -1$  and  $p_2 = -1$  as follows

$$\text{find } \mathbf{K} \text{ such that } |s\mathbf{I} - (\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K})| = (s + 1)^2 \quad (1.7.15)$$

we obtain a system which its natural evolution can be represented by the phase portrait of Figure 1.22b\*\*.

<sup>||</sup>The phase portrait is built plotting the evolution  $\vec{x}(t)$  in the plane  $(x_1 \ x_2)$ . The evolution  $\vec{x}(t)$  is computed by solving the system

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t} \vec{x}(0) \quad (1.7.12)$$

where  $\vec{x}(0) = [1 \ 0]^T$  is the initial condition. The solution can be computed as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1} \left\{ \left[ s\mathbf{I} - \tilde{\mathbf{A}} \right]^{-1} \right\} \vec{x}(0) = \mathcal{L}^{-1} \left\{ \frac{1}{s^2 + 1} \begin{bmatrix} s & 1 \\ -1 & s \end{bmatrix} \right\} \vec{x}(0) = \\ &= \begin{bmatrix} \cos(t) & \sin(t) \\ -\sin(t) & \cos(t) \end{bmatrix} \vec{x}(0) = \begin{bmatrix} \cos(t) \\ -\sin(t) \end{bmatrix} \end{aligned} \quad (1.7.13)$$

The phase portrait is build as parametric curve:

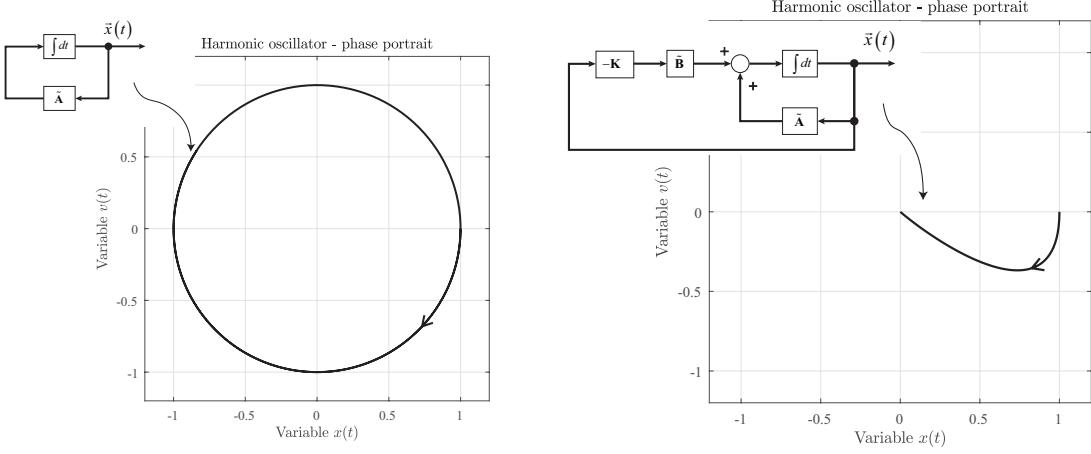
$$\mathcal{C} = \begin{cases} x_1(t) = \cos(t) \\ x_2(t) = -\sin(t) \end{cases} \quad (1.7.14)$$

<sup>\*\*</sup>Let's consider the case  $\tilde{\mathbf{A}} - \mathbf{K}\tilde{\mathbf{B}}$  where

$$\vec{x}(t) = e^{(\tilde{\mathbf{A}} - \mathbf{K}\tilde{\mathbf{B}})t} \vec{x}(0) \quad (1.7.16)$$

where  $\vec{x}(0) = [1 \ 0]^T$  is the initial condition and where  $\tilde{\mathbf{A}} - \mathbf{K}\tilde{\mathbf{B}} = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}$ . The solution can be computed as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1} \left\{ \left[ s\mathbf{I} - (\tilde{\mathbf{A}} - \mathbf{K}\tilde{\mathbf{B}}) \right]^{-1} \right\} \vec{x}(0) = \mathcal{L}^{-1} \left\{ \frac{1}{(s+1)^2} \begin{bmatrix} s+2 & 1 \\ -1 & s \end{bmatrix} \right\} \vec{x}(0) = \\ &= \begin{bmatrix} \exp(-t)(t+1) & \exp(-t)t \\ -\exp(-t)t & -\exp(-t)(t-1) \end{bmatrix} \vec{x}(0) = \begin{bmatrix} \exp(-t)(t+1) \\ -\exp(-t)t \end{bmatrix} \end{aligned} \quad (1.7.17)$$



(a) Phase portrait of the harmonic oscillator, where  $\vec{x}(0) = [1 \ 0]$  and  $u(t) = 0$ .

(b) Phase portrait of the harmonic oscillator, where  $\vec{x}(0) = [1 \ 0]$  and  $u(t) = -\mathbf{K}\vec{x}(t)$ .

Figure 1.22: Phase portraits of the harmonic oscillator with and without state feedback compensation.

◇

## 1.8 Observability

In this section we discuss the observability of linear systems. Consider the unforced system described by the following equations:

$$\dot{\vec{x}} = \tilde{\mathbf{A}}\vec{x} \quad (1.8.1)$$

$$\vec{y} = \mathbf{C}\vec{x} \quad (1.8.2)$$

where

- $\vec{x}$  = state vector ( $n$ -vector)
- $\vec{y}$  = output vector ( $m$ -vector). For SISO is  $m = 1$
- $\tilde{\mathbf{A}} = n \times n$  matrix
- $\mathbf{C} = m \times n$  matrix

The system is said to be completely observable if every initial state  $\vec{x}(0)$  can be determined from the observation of  $\vec{y}(t)$  over a finite time interval,  $0 \leq t \leq t_1$ . The system is, therefore, completely observable if every transition of the state eventually affects every element of the

The phase portrait is build as parametric curve:

$$\mathcal{C} = \begin{cases} x_1(t) = \exp(-t)(t+1) \\ x_2(t) = -\exp(-t)t \end{cases} \quad (1.7.18)$$

output vector. *The concept of observability is useful in solving the problem of reconstructioning of unmeasurable state variables from measurable variables in the minimum possible length of time.*

The concept of observability is very important because, in practice, the difficulty encountered with state feedback control is that some of the state variables are not accessible for direct measurement, with the results that it become necessary to estimate the unmeasured state variables in order to construct the control signals. Such estimates of state variables are possible if only if the system is completely observable.

Consider the system

$$\dot{\vec{x}} = \tilde{\mathbf{A}}\vec{x} + \tilde{\mathbf{B}}u \quad (1.8.3)$$

$$y = \mathbf{C}\vec{x} \quad (1.8.4)$$

where the state vector evolution is given as follows

$$\vec{x}(t) = e^{\tilde{\mathbf{A}}t}\vec{x}(0) + \int_0^t e^{\tilde{\mathbf{A}}(t-\tau)}\tilde{\mathbf{B}}u(\tau)d\tau \quad (1.8.5)$$

and where the output evolution as well

$$y(t) = \mathbf{C}e^{\tilde{\mathbf{A}}t}\vec{x}(0) + \mathbf{C} \int_0^t e^{\tilde{\mathbf{A}}(t-\tau)}\tilde{\mathbf{B}}u(\tau)d\tau \quad (1.8.6)$$

Since the matrices  $\tilde{\mathbf{A}}$ ,  $\tilde{\mathbf{B}}$ ,  $\mathbf{C}$  are known and  $u(t)$  is also known, the last two terms on the right hand side of this last equation are known quantities. Therefore, they may be subtracted from the observed value of  $y(t)$ . Hence, for investigating a necessary and sufficient condition for complete observability, it suffices to consider the system described by Eq. (1.8.1) and (1.8.2).

**Complete Observability of Continuos-Time Systems.** Consider the system described by Eq. (1.8.1) and (1.8.2). The output quantity  $y(t)$  is

$$y(t) = \mathbf{C}e^{\tilde{\mathbf{A}}t}\vec{x}(0) \quad (1.8.7)$$

where

$$e^{\tilde{\mathbf{A}}t} = \sum_{k=0}^{n-1} \alpha_k(t)\tilde{\mathbf{A}}^k \quad (1.8.8)$$

where  $n$  is the degree of the characteristic polynomial. Hence, we obtain

$$y(t) = \sum_{k=0}^{n-1} \alpha_k(t)\mathbf{C}\tilde{\mathbf{A}}^k\vec{x}(0) \quad (1.8.9)$$

or

$$y(t) = \alpha_0(t)\mathbf{C}\vec{x}(0) + \alpha_1(t)\mathbf{C}\tilde{\mathbf{A}}\vec{x}(0) + \dots + \alpha_{n-1}(t)\mathbf{C}\tilde{\mathbf{A}}^{n-1}\vec{x}(0) \quad (1.8.10)$$

If the system is completely observable, then, given the output  $y(t)$  over a time interval  $0 \leq t \leq t_1$ ,  $\vec{x}(0)$  is uniquely determined from Eq. (1.8.10). It can be shown that this requires the rank of the  $nm \times n$  matrix ( $n \times n$  for SISO) to be  $n$ .

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\tilde{\mathbf{A}} \\ \vdots \\ \mathbf{C}\tilde{\mathbf{A}}^{n-1} \end{bmatrix} \quad (1.8.11)$$

From this analysis, we can state the condition for complete observability as follow: The system described by Eq. (1.8.1) and (1.8.2) is completely observable if and only if the  $n \times nm$  matrix

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{C}\tilde{\mathbf{A}} \\ \vdots \\ \mathbf{C}\tilde{\mathbf{A}}^{n-1} \end{bmatrix} \quad (1.8.12)$$

is of rank  $n$  or has  $n$  linearly independent column vectors. This matrix is called the *observability matrix*.

**Condition for Complete Observability in the s Plane.** The conditions for complete observability can also be stated in terms of transfer functions. The necessary and sufficient conditions for complete observability is that no cancellation occur in the transfer function. *If cancellation occurs, the cancelled mode cannot be observed in the output.*

## 1.9 Principle of Duality

We shall now discuss the relationship between controllability and observability. We shall introduce the principle of duality, due to Kalman, to clarify apparent analogies between controllability and observability. Consider the system  $S_1$  described by

$$\begin{aligned} \dot{\vec{x}} &= \mathbf{A}\vec{x} + \mathbf{B}\vec{u} \\ \vec{y} &= \mathbf{C}\vec{x} \end{aligned} \quad (1.9.1)$$

where

- $\vec{x}$  = state vector ( $n$ -vector)
- $\vec{u}$  = input vector ( $r$ -vector). For SISO is  $r = 1$
- $\vec{y}$  = output vector ( $m$ -vector). For SISO is  $m = 1$
- $\mathbf{A} = n \times n$  matrix
- $\mathbf{B} = n \times r$  matrix

- $\mathbf{C} = m \times n$  matrix

and the dual system  $S_2$  described by

$$\begin{aligned}\dot{\vec{z}} &= \mathbf{A}^* \vec{z} + \mathbf{C}^* \vec{v} \\ \vec{n} &= \mathbf{B}^* \vec{z}\end{aligned}\tag{1.9.2}$$

where

- $\vec{z}$  = state vector ( $n$ -vector)
- $\vec{v}$  = input vector ( $r$ -vector). For SISO is  $r = 1$
- $\vec{n}$  = output vector ( $m$ -vector). For SISO is  $m = 1$
- $\mathbf{A}^*$  = conjugate transpose of  $\mathbf{A} \in \mathbb{C}^{n \times n}$  (transpose of  $\mathbf{A}$  if  $\mathbf{A} \in \mathbb{R}^{n \times n}$ )
- $\mathbf{B}^*$  = conjugate transpose of  $\mathbf{B} \in \mathbb{C}^{n \times r}$  (transpose of  $\mathbf{B}$  if  $\mathbf{B} \in \mathbb{R}^{n \times r}$ )
- $\mathbf{C}^*$  = conjugate transpose of  $\mathbf{C} \in \mathbb{C}^{m \times n}$  (transpose of  $\mathbf{C}$  if  $\mathbf{C} \in \mathbb{R}^{m \times n}$ )

The principle of duality states that the system  $S_1$  is completely state controllable (observable) if and only if system  $S_2$  is completely observable (state controllable). To verify this principle, let us write down the necessary and sufficient condition for complete state controllability and complete observability for system  $S_1$  and  $S_2$ .

For system  $S_1$

- 1** A necessary and sufficient condition for complete state controllability is that the rank of the  $n \times nr$  matrix

$$[\mathbf{B} \quad \mathbf{AB} \quad \dots \quad \mathbf{A}^{n-1}\mathbf{B}]$$

be  $n$ .

- 2** A necessary and sufficient condition for complete state observability is that the rank of the  $n \times nm$  matrix

$$[\mathbf{C}^* \quad \mathbf{A}^*\mathbf{C}^* \quad \dots \quad (\mathbf{A}^*)^{n-1}\mathbf{C}^*] = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}^*$$

be  $n$ .

For system  $S_2$

- 1** A necessary and sufficient condition for complete state controllability is that the rank of the  $n \times nm$  matrix

$$\begin{bmatrix} \mathbf{C}^* & \mathbf{A}^* \mathbf{C}^* & \dots & (\mathbf{A}^*)^{n-1} \mathbf{C}^* \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}^*$$

be  $n$ .

- 2** A necessary and sufficient condition for complete state observability is that the rank of the  $n \times nr$  matrix

$$\begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{n-1} \mathbf{B} \end{bmatrix}$$

be  $n$ .

By use of this principle, the observability of a given system can be checked by testing the state controllability of its dual.

## 1.10 State Observer

In the pole placement approach to the design of control systems, we assume that all state variable are available for feedback. In practice, however, not all state variables are available for feedback. Then we need to estimate unavailable state variables.

**State Observer.** A state observer estimates the state variables based on the measurements of the output and control variables. *State observers can be designed if and only if the observability condition is satisfied.* Considering the plant defined by

$$\begin{aligned} \dot{\vec{x}} &= \tilde{\mathbf{A}}\vec{x} + \tilde{\mathbf{B}}u \\ y &= \mathbf{C}\vec{x} \end{aligned} \tag{1.10.1}$$

The observer is a subsystem to reconstruct the state vector of the plant. The mathematical model of the observer is basically the same as that of the plant, except that we include an additional term that includes the estimation error to compensate for inaccuracies in matrices  $\tilde{\mathbf{A}}$  and  $\tilde{\mathbf{B}}$  and the lack of the initial error. The estimation error or the observation error is the difference between the measured output and the estimated output. The initial error is the difference between the initial state and the initial estimated state. Thus, we define the mathematical model of the observer to be

$$\begin{aligned} \dot{\hat{x}} &= \tilde{\mathbf{A}}\hat{x} + \tilde{\mathbf{B}}u + \mathbf{L}(y - \mathbf{C}\hat{x}) \\ &= (\tilde{\mathbf{A}} - \mathbf{LC})\hat{x} + \tilde{\mathbf{B}}u + \mathbf{Ly} \end{aligned} \tag{1.10.2}$$

where  $\hat{\vec{x}}$  is the estimated state and  $\mathbf{C}\hat{\vec{x}}$  is the estimated output. *The inputs of the observer are the output  $\vec{y}$  and the control input  $\vec{u}$ .* Matrix  $\mathbf{L}$  which is called the observer gain matrix, is a weighting matrix to the correction term involving the difference between the measured output  $\vec{y}$  and the estimated output  $\mathbf{C}\hat{\vec{x}}$ . This term continuously corrects the model output and improves the performance of the observer. Figure 1.23 shows the block diagram of the system and the full-order state observer. The order of the state observer that will be discussed here is the same as that of the plant.

To obtain the observer error equation, let us subtract Eq. (1.10.2) from Eq. (1.10.1):

$$\begin{aligned}\dot{\vec{x}} - \dot{\hat{\vec{x}}} &= \tilde{\mathbf{A}}\vec{x} - \tilde{\mathbf{A}}\hat{\vec{x}} - \mathbf{L}(\mathbf{C}\vec{x} - \mathbf{C}\hat{\vec{x}}) \\ &= (\tilde{\mathbf{A}} - \mathbf{LC})(\vec{x} - \hat{\vec{x}})\end{aligned}\quad (1.10.3)$$

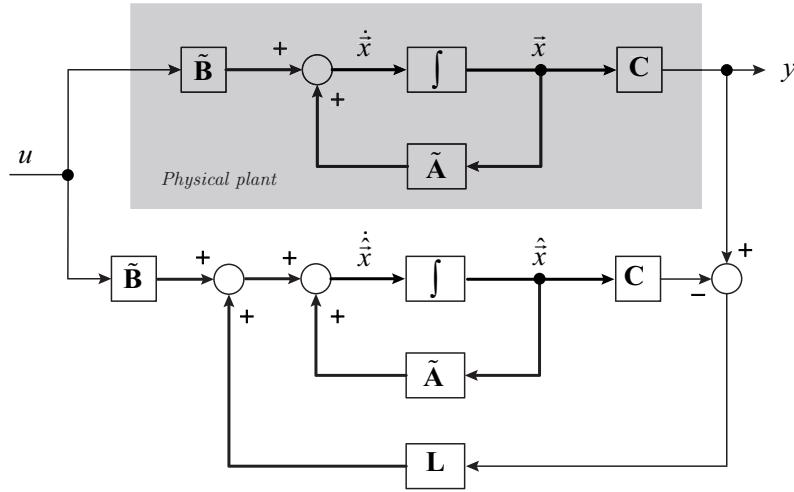
Define the difference between  $\vec{x}$  and  $\hat{\vec{x}}$  as the error vector  $\vec{e}$ , or

$$\vec{e} = \vec{x} - \hat{\vec{x}} \quad (1.10.4)$$

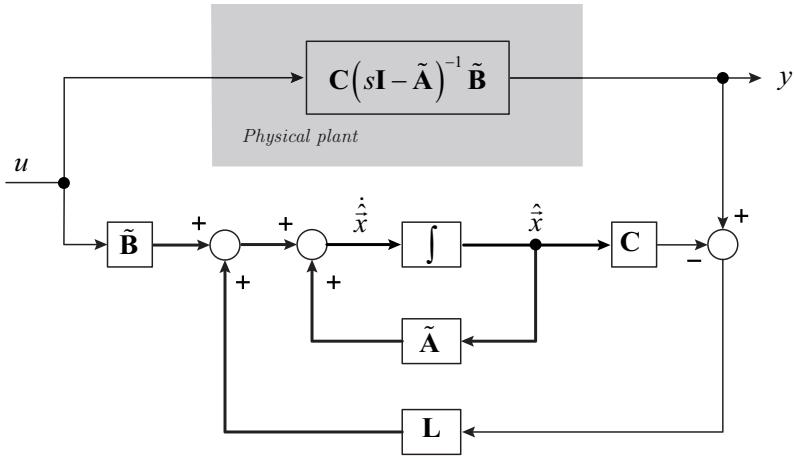
the equation (1.10.3) becomes

$$\dot{\vec{e}} = (\tilde{\mathbf{A}} - \mathbf{LC})\vec{e} \quad (1.10.5)$$

From Eq. (1.10.5), we see that the dynamic behaviour of the error vector is determined by the eigenvalues of matrix  $(\tilde{\mathbf{A}} - \mathbf{LC})$ . If matrix  $(\tilde{\mathbf{A}} - \mathbf{LC})$  is a stable matrix, the error vector will converge to zero for any initial error vector  $\vec{e}(0)$ . That is,  $\hat{\vec{x}}(t)$  will converge to  $\vec{x}(t)$  regardless of the value of  $\vec{x}(0)$  and  $\hat{\vec{x}}(0)$ . If the eigenvalues of the matrix  $(\tilde{\mathbf{A}} - \mathbf{LC})$  are chosen in such a way that the dynamic behaviour of the error vector is asymptotically stable and is adequately fast, then any error vector will tend to zero (the origin) with an adequate speed. If the plant is completely observable, then it can be proved that it is possible to choose matrix  $\mathbf{L}$  such that  $(\tilde{\mathbf{A}} - \mathbf{LC})$  has arbitrarily desired eigenvalues. That is, the observer gain matrix  $\mathbf{L}$  can be determined to yield the desired matrix  $(\tilde{\mathbf{A}} - \mathbf{LC})$ .



(a) Physical Plant in the form of state space representation.



(b) Physical Plant in the form of transfer function.

Figure 1.23: Full-order state observer.

**Dual Problem.** The design of the full-order observer becomes that of determining an appropriate  $\mathbf{L}$  such that  $(\mathbf{A} - \mathbf{LC})$  has desired eigenvalues. Thus the problem here become the same as the pole placement problem and this property is called duality.

Consider the system defined by

$$\begin{aligned}\dot{\vec{x}} &= \mathbf{A}\vec{x} + \mathbf{B}u \\ y &= \mathbf{C}\vec{x}\end{aligned}\tag{1.10.6}$$

In designing the full-order state observer, we may solve the dual problem, that is, solve the

pole-placement problem for the dual system

$$\begin{aligned}\dot{\vec{z}} &= \mathbf{A}^* \vec{z} + \mathbf{B}^* v \\ n &= \mathbf{C}^* \vec{z}\end{aligned}\tag{1.10.7}$$

assuming the control signal  $v$  to be

$$v = -\mathbf{K} \vec{z}\tag{1.10.8}$$

If the dual system is completely state controllable, then the state feedback gain matrix  $\mathbf{K}$  can be determined such that matrix  $(\mathbf{A}^* - \mathbf{C}^* \mathbf{K})$  will yield a set of the desired eigenvalues. If  $\mu_1, \mu_2, \dots, \mu_n$  are the desired eigenvalues of the state observer matrix, then by taking the same  $\mu_i$ 's as the desired eigenvalues of the state-feedback gain matrix of the dual system, we obtain

$$|s\mathbf{I} - (\mathbf{A}^* - \mathbf{C}^* \mathbf{K})| = (s - \mu_1)(s - \mu_2)\dots(s - \mu_n)\tag{1.10.9}$$

Noting that the eigenvalues of  $\mathbf{A}^* - \mathbf{C}^* \mathbf{K}$  and those of  $\mathbf{A} - \mathbf{K}^* \mathbf{C}$  are the same, we have

$$|s\mathbf{I} - (\mathbf{A}^* - \mathbf{C}^* \mathbf{K})| = |s\mathbf{I} - (\mathbf{A} - \mathbf{K}^* \mathbf{C})|\tag{1.10.10}$$

Comparing the characteristic polynomial  $|s\mathbf{I} - (\mathbf{A} - \mathbf{K}^* \mathbf{C})|$  and the characteristic polynomial  $|s\mathbf{I} - (\mathbf{A} - \mathbf{L} \mathbf{K})|$  for the observer system we find that  $\mathbf{L}$  and  $\mathbf{K}^*$  are related by

$$\mathbf{L} = \mathbf{K}^*\tag{1.10.11}$$

Thus, using the matrix  $\mathbf{K}$  determined by pole placement approach in the dual system, the observer gain matrix  $\mathbf{L}$  for the original system can be determined by using the relationship  $\mathbf{L} = \mathbf{K}^*$ .

**Necessary and Sufficient Condition for State Observation.** As discussed, a necessary and sufficient condition for the determination of the observer gain matrix  $\mathbf{L}$  for the desired eigenvalues of  $\mathbf{A} - \mathbf{L} \mathbf{C}$  is that the dual of the original system

$$\dot{\vec{z}} = \mathbf{A}^* \vec{z} + \mathbf{C}^* v\tag{1.10.12}$$

be completely state controllable. The complete state controllability condition for this dual system is that the rank of

$$\left[ \mathbf{C}^* \quad \mathbf{A}^* \mathbf{C}^* \quad \dots \quad (\mathbf{A}^*)^{n-1} \mathbf{C}^* \right] = \left[ \begin{array}{c} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{array} \right]^*\tag{1.10.13}$$

be  $n$ . This is the condition for complete observability of the original system defined by Eq. (1.10.6).

**Direct-Substitution Approach to Obtain State Observer Gain Matrix  $\mathbf{L}$ .** Similar to the case of pole placement, if the system is of low order, then direct substitution of matrix  $\mathbf{L}$  into the desired characteristic polynomial may be simpler. For example, if  $\vec{x}$  is a 3-vector, then write the observer gain matrix  $\mathbf{L}$  as

$$\mathbf{L} = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} \quad (1.10.14)$$

Substitute this  $\mathbf{L}$  matrix into the desired characteristic polynomial:

$$|s\mathbf{I} - (\mathbf{A} - \mathbf{LC})| = (s - \mu_1)(s - \mu_2)(s - \mu_3) \quad (1.10.15)$$

By equating the coefficient of the like powers of  $s$  on both sides of this last equation, we can determine the values of  $l_1, l_2$  and  $l_3$ .

**Ackermann's Formula.** Let's go back to state feedback design and we consider the system defined by

$$\begin{aligned} \dot{\vec{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ y &= \mathbf{Cx} \end{aligned} \quad (1.10.16)$$

Use of the state feedback control

$$u = -\mathbf{K}\vec{x}$$

modifies the system equation to

$$\dot{\vec{x}} = [\mathbf{A} - \mathbf{BK}] \vec{x} \quad (1.10.17)$$

Let us define

$$\tilde{\mathbf{A}} = [\mathbf{A} - \mathbf{BK}]$$

The desired characteristic equation is

$$\begin{aligned} |s\mathbf{I} - (\mathbf{A} - \mathbf{LC})| &= |s\mathbf{I} - \tilde{\mathbf{A}}| = (s - \mu_1)(s - \mu_2)\dots(s - \mu_n) \\ &= s^n + \alpha_1 s^{n-1} + \dots + \alpha_{n-1} s + \alpha_n = 0 \end{aligned} \quad (1.10.18)$$

Since the **Cayley-Hamilton** theorem states that  $\tilde{\mathbf{A}}$  satisfies its own characteristic equation, we have

$$\phi(\tilde{\mathbf{A}}) = \tilde{\mathbf{A}}^n + \alpha_1 \tilde{\mathbf{A}}^{n-1} + \dots + \alpha_{n-1} \tilde{\mathbf{A}} + \alpha_n \mathbf{I} = 0 \quad (1.10.19)$$

We shall utilize Eq. (1.10.19) to derive Ackermann's formula. We consider the case  $n = 3$ .

$$\mathbf{I} = \mathbf{I}$$

$$\tilde{\mathbf{A}} = \mathbf{A} - \mathbf{BK}$$

$$\tilde{\mathbf{A}}^2 = (\mathbf{A} - \mathbf{BK})^2 = \mathbf{A}^2 - \mathbf{ABK} - \mathbf{BKA}$$

$$\tilde{\mathbf{A}}^3 = (\mathbf{A} - \mathbf{BK})^3 = \mathbf{A}^3 \mathbf{BK} - \mathbf{A}^2 \mathbf{BK} - \mathbf{ABKA} - \mathbf{BKA}^2$$

$$\begin{aligned}
 \alpha_3\mathbf{I} + \alpha_2\tilde{\mathbf{A}} + \alpha_1\tilde{\mathbf{A}}^2 + \tilde{\mathbf{A}}^3 &= \\
 &= \alpha_3\mathbf{I} + \alpha_2(\mathbf{A} - \mathbf{B}\mathbf{K}) + \alpha_1(\mathbf{A}^2 - \mathbf{ABK} - \mathbf{BKA}) + \\
 &\quad + \mathbf{A}^3 - \mathbf{A}^2\mathbf{BK} - \mathbf{ABK}\tilde{\mathbf{A}} - \mathbf{BKA}\tilde{\mathbf{A}}^2 = \\
 &= \alpha_3\mathbf{I} + \alpha_2\mathbf{A} + \alpha_1\mathbf{A}^2 + \mathbf{A}^3 - \alpha_2\mathbf{BK} - \alpha_1\mathbf{ABK} + \\
 &\quad - \alpha_1\mathbf{BKA}(\mathbf{A}) - \mathbf{A}^2\mathbf{BK} - \mathbf{ABK}(\mathbf{A}) - \mathbf{BKA}\tilde{\mathbf{A}}^2
 \end{aligned} \tag{1.10.20}$$

Referring to Eq. (10.9), we have

$$\alpha_3\mathbf{I} + \alpha_2\tilde{\mathbf{A}} + \alpha_1\tilde{\mathbf{A}}^2 + \tilde{\mathbf{A}}^3 = \phi(\tilde{\mathbf{A}}) = 0$$

Also we have

$$\alpha_3\mathbf{I} + \alpha_2\mathbf{A} + \alpha_1\mathbf{A}^2 + \mathbf{A}^3 = \phi(\mathbf{A}) \neq 0$$

Substituting the last two equations into equation (4.1.10) we have

$$\phi(\tilde{\mathbf{A}}) = \phi(\mathbf{A}) - \alpha_2\mathbf{BK} - \alpha_1\mathbf{BKA} - \mathbf{BKA}^2 - \alpha_1\mathbf{ABK} - \mathbf{ABK}\tilde{\mathbf{A}} - \mathbf{A}^2\mathbf{BK}$$

Since  $\phi(\tilde{\mathbf{A}}) = 0$  we obtain

$$\begin{aligned}
 \phi(\tilde{\mathbf{A}}) &= \mathbf{B}(\alpha_2\mathbf{K} + \alpha_1\mathbf{K}\tilde{\mathbf{A}} + \mathbf{K}\tilde{\mathbf{A}}^2) + \mathbf{AB}(\alpha_1\mathbf{K} + \mathbf{K}\tilde{\mathbf{A}}) + \mathbf{A}^2\mathbf{BK} + \\
 &= [\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B}] \begin{bmatrix} \alpha_2\mathbf{K} + \alpha_1\mathbf{K}\tilde{\mathbf{A}} + \mathbf{K}\tilde{\mathbf{A}}^2 \\ \alpha_1\mathbf{K} + \mathbf{K}\tilde{\mathbf{A}} \\ \mathbf{K} \end{bmatrix}
 \end{aligned} \tag{1.10.21}$$

Since the system is completely state controllable, the inverse of the controllability matrix

$$[\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B}] \tag{1.10.22}$$

exists. Premultiplying both side of Eq. (10.9) by the inverse of the controllability matrix, we obtain

$$[\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B}]^{-1} \phi(\mathbf{A}) = \begin{bmatrix} \alpha_2\mathbf{K} + \alpha_1\mathbf{K}\tilde{\mathbf{A}} + \mathbf{K}\tilde{\mathbf{A}}^2 \\ \alpha_1\mathbf{K} + \mathbf{K}\tilde{\mathbf{A}} \\ \mathbf{K} \end{bmatrix} \tag{1.10.23}$$

Premultiplying both sides of this last equation by  $[0 \quad 0 \quad 1]$ , we obtain

$$[0 \quad 0 \quad 1] [\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B}]^{-1} \phi(\mathbf{A}) = [0 \quad 0 \quad 1] \begin{bmatrix} \alpha_2\mathbf{K} + \alpha_1\mathbf{K}\tilde{\mathbf{A}} + \mathbf{K}\tilde{\mathbf{A}}^2 \\ \alpha_1\mathbf{K} + \mathbf{K}\tilde{\mathbf{A}} \\ \mathbf{K} \end{bmatrix} = \mathbf{K} \tag{1.10.24}$$

Which can be rewritten as

$$\mathbf{K} = [0 \quad 0 \quad 1] [\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B}]^{-1} \phi(\mathbf{A})$$

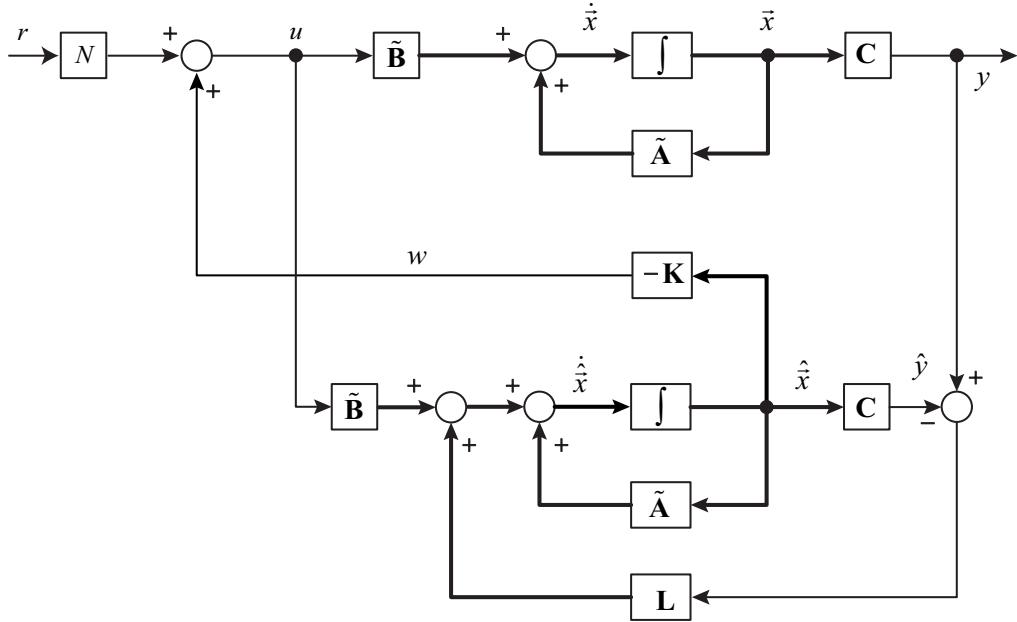


Figure 1.24: Observed state feedback control system.

This last equation gives the required state feedback gain matrix  $\mathbf{K}$ .

For an arbitrary positive integer  $n$  we have

$$K = \begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix}^{-1} \phi(\mathbf{A}) \quad (1.10.25)$$

Which is known as Ackermann's formula for the determination of the state feedback gain matrix  $\mathbf{K}$ .

For the duality properties we can modify the above Ackermann's formula to

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}^* & \mathbf{A}^*\mathbf{C}^* & \dots & (\mathbf{A}^*)^{n-1}\mathbf{C}^* \end{bmatrix}^{-1} \phi(\mathbf{A}^*) \quad (1.10.26)$$

Thus

$$\mathbf{L} = \mathbf{K}^* = \phi(\mathbf{A}) \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-2} \\ \mathbf{CA}^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (1.10.27)$$

**Effects of the Addition of the State Observer to the State Feedback.** In the pole-placement design process, we assumed that the actual state  $\vec{x}(t)$  was available for feedback. In practice, however, the actual state  $\vec{x}(t)$  may not be measurable, so we will need to design an observer and use the observed state  $\hat{\vec{x}}(t)$  for feedback as shown in Figure 1.24. The design process, therefore, becomes a two stage process, the first stage being the determination of the feedback gain matrix  $\mathbf{K}$  to yield the desired characteristic equation and the second stage being

the determination of the observer gain matrix  $\mathbf{L}$  to yield the desired observer characteristic equation.

Let us now investigate the effects of the use of the observed state  $\hat{\vec{x}}(t)$ , rather than the actual state  $\vec{x}(t)$ , on the characteristic equation of a closed-loop control system.

Consider the completely state controllable and completely observable system defined by the equations

$$\begin{aligned}\dot{\vec{x}}(t) &= \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \\ y(t) &= \mathbf{C}\vec{x}(t)\end{aligned}$$

For the state feedback control based on the observed state  $\hat{\vec{x}}(t)$

$$u(t) = -\mathbf{K}\hat{\vec{x}}(t)$$

The state equation becomes

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) - \tilde{\mathbf{B}}\mathbf{K}\hat{\vec{x}}(t) = [\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K}]\vec{x}(t) + \tilde{\mathbf{B}}\mathbf{K}[\vec{x}(t) - \hat{\vec{x}}(t)] \quad (1.10.28)$$

The difference between the actual state  $\vec{x}(t)$  and the observed state  $\hat{\vec{x}}(t)$  has been defined as the error  $\vec{e}(t)$ :

$$\vec{e}(t) = \vec{x}(t) - \hat{\vec{x}}(t)$$

Substitution of the error vector  $\vec{e}(t)$  into the Eq. (1.10.28) gives

$$\dot{\vec{x}}(t) = [\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K}]\vec{x}(t) + \tilde{\mathbf{B}}\mathbf{K}\vec{e}(t) \quad (1.10.29)$$

While the observer error equation is given by

$$\dot{\vec{e}}(t) = [\tilde{\mathbf{A}} - \mathbf{L}\mathbf{C}]\vec{e}(t) \quad (1.10.30)$$

Combining Eq. (1.10.29) and (1.10.30) we obtain

$$\begin{bmatrix} \dot{\vec{x}}(t) \\ \dot{\vec{e}}(t) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K} & \tilde{\mathbf{B}}\mathbf{K} \\ 0 & \tilde{\mathbf{A}} - \mathbf{L}\mathbf{C} \end{bmatrix} \begin{bmatrix} \vec{x}(t) \\ \vec{e}(t) \end{bmatrix} \quad (1.10.31)$$

Eq. (1.10.31) describes the dynamics of the observed state feedback control system. The characteristic equation for the system is

$$\begin{vmatrix} s\mathbf{I} - \tilde{\mathbf{A}} + \tilde{\mathbf{B}}\mathbf{K} & -\tilde{\mathbf{B}}\mathbf{K} \\ 0 & s\mathbf{I} - \tilde{\mathbf{A}} + \mathbf{L}\mathbf{C} \end{vmatrix} = 0 \quad (1.10.32)$$

or

$$\left| s\mathbf{I} - \tilde{\mathbf{A}} + \tilde{\mathbf{B}}\mathbf{K} \right| \left| s\mathbf{I} - \tilde{\mathbf{A}} + \mathbf{L}\mathbf{C} \right| = 0 \quad (1.10.33)$$

Notice that the closed-loop poles of the observed-state feedback control system consist of the poles due to the pole placement design alone and the poles due to the observer design alone. This means that the pole-placement design and the observer design are independent of each other. They can be designed separately and combined to form the observed state feedback control system. Note that, if the order of the plant is  $n$ , then the observer is also of  $n$ th order and the resulting characteristic equation for the entire closed loop system become or order  $2n$ .

Suppose now to use the following input quantity:

$$u(t) = -\mathbf{K}\hat{\vec{x}}(t) + N\mathbf{r}(t) \quad (1.10.34)$$

The equation of the state observer become the following:

$$\dot{\hat{\vec{x}}}(t) = [\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K} - \mathbf{L}\mathbf{C}] \hat{\vec{x}}(t) + \tilde{\mathbf{B}}\mathbf{N} + \mathbf{L}\mathbf{y}(t) \quad (1.10.35)$$

as shown in Figure 1.24.

**Transfer Function of the Observer-Based Controller.** Considering the plant defined by

$$\begin{aligned} \dot{\vec{x}}(t) &= \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \\ y(t) &= \mathbf{C}\vec{x}(t) \end{aligned} \quad (1.10.36)$$

Assume that the plant is completely observable. Assume that we use observed-state feedback control  $u = -\mathbf{K}\hat{\vec{x}}$ . Then the equation for the observer are given by

$$\dot{\hat{\vec{x}}}(t) = [\mathbf{A} - \mathbf{L}\mathbf{C} - \tilde{\mathbf{B}}\mathbf{K}] \hat{\vec{x}}(t) + \mathbf{L}\mathbf{y}(t) \quad (1.10.37)$$

where

$$u(t) = -\mathbf{K}\hat{\vec{x}}(t) \quad (1.10.38)$$

By taking the Laplace transform of Eq. (1.10.38) we obtain

$$\hat{\vec{X}}(s) = [s\mathbf{I} - \tilde{\mathbf{A}} + \mathbf{L}\mathbf{C} + \tilde{\mathbf{B}}\mathbf{K}]^{-1} \mathbf{L}\mathbf{Y}(s) \quad (1.10.39)$$

By substituting  $\hat{\vec{X}}(s)$  into the Laplace transform of Eq. (1.10.38), we obtain

$$U(s) = -\mathbf{K} [s\mathbf{I} - \tilde{\mathbf{A}} + \mathbf{L}\mathbf{C} + \tilde{\mathbf{B}}\mathbf{K}]^{-1} \mathbf{L}\mathbf{Y}(s) \quad (1.10.40)$$

Then the transfer function  $U(s)/Y(s)$  can be obtained as

$$\frac{U(s)}{Y(s)} = -\mathbf{K} \left[ s\mathbf{I} - \tilde{\mathbf{A}} + \mathbf{LC} + \tilde{\mathbf{B}}\mathbf{K} \right]^{-1} \mathbf{L} \quad (1.10.41)$$

Hence we call the transfer function

$$\frac{U(s)}{-Y(s)} = \mathbf{K} \left[ s\mathbf{I} - \tilde{\mathbf{A}} + \mathbf{LC} + \tilde{\mathbf{B}}\mathbf{K} \right]^{-1} \mathbf{L} \quad (1.10.42)$$

the observer-based controller transfer function or, simply, the observer-controller transfer function.

Note that the observer-controller matrix

$$\tilde{\mathbf{A}} - \mathbf{LC} - \tilde{\mathbf{B}}\mathbf{K} \quad (1.10.43)$$

may or may not be stable, although  $\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K}$  and  $\tilde{\mathbf{A}} - \mathbf{LC}$  are chosen to be stable. In fact, in some cases the matrix  $\tilde{\mathbf{A}} - \mathbf{LC} - \tilde{\mathbf{B}}\mathbf{K}$  may be poorly stable or even unstable.

**Example 1.11 (Double Integrator Observer).** Consider the plant

$$\begin{cases} \ddot{x} = 0 \\ y = x_1 \end{cases} \quad (1.10.44)$$

and by its state space representation as follows

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) \\ y(t) = \mathbf{C}\vec{x}(t) \end{cases} \quad (1.10.45)$$

where  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$  and  $\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ .

The equations which describe the **double integrator observer** can be written as follows

$$\dot{\hat{x}}(t) = \tilde{\mathbf{A}}\hat{x}(t) + \tilde{\mathbf{L}}(y(t) - \mathbf{C}\hat{x}(t)) \quad (1.10.46)$$

Eq. (1.10.46) can also be written as follows

$$\begin{cases} \dot{\hat{x}}_1(t) = \hat{x}_2(t) + \tilde{l}_1[x_1(t) - \hat{x}_1(t)] \\ \dot{\hat{x}}_2(t) = \tilde{l}_2[x_1(t) - \hat{x}_1(t)] \end{cases} \quad (1.10.47)$$

where  $y(t) = x_1(t)$  and  $\tilde{\mathbf{L}} = [\tilde{l}_1 \quad \tilde{l}_2]^T$ ;

The calculation of the observer gain matrix  $\tilde{\mathbf{L}}$  can be done using the pole placement proce-

dure e.g. applying the `acker` command in matlab

$$L\tilde{t} = \text{acker}(A\tilde{t}, C, [-p_1, -p_2]); \quad (1.10.48)$$

The graphical representation of Eq. (1.10.47) results as shown in Figure 1.25

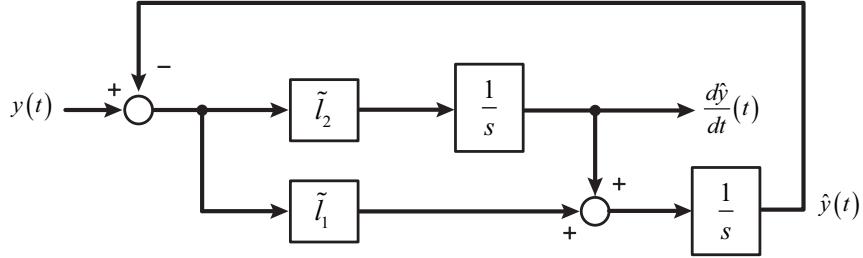


Figure 1.25: Double Integrator Observer.

The discrete time representation of the dynamical system  $\ddot{x} = 0$ , can be represented as follows

$$\begin{cases} \vec{x}(k+1) = \mathbf{A}\vec{x}(k) \\ y(k) = \mathbf{C}\vec{x}(k) \end{cases} \quad (1.10.49)$$

where  $\mathbf{A} = \mathbf{I} + \tilde{\mathbf{A}}t_s = \begin{bmatrix} 1 & t_s \\ 0 & 1 \end{bmatrix}$  and  $\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ .

The equation of the state observer in the discrete time domain results as follows

$$\hat{\vec{x}}(k+1) = \mathbf{A}\hat{\vec{x}}(k) + \mathbf{L}(y(k) - \mathbf{C}\hat{\vec{x}}(k)) \quad (1.10.50)$$

which can be represented as well

$$\begin{cases} \hat{x}_1(k+1) = \hat{x}_1(k) + \hat{x}_2(k)t_s + l_1[x_1(k) - \hat{x}_1(k)] \\ \hat{x}_2(k+1) = \hat{x}_2(k) + l_2[x_1(k) - \hat{x}_1(k)] \end{cases} \quad (1.10.51)$$

where  $y(k) = x_1(k)$  and  $\mathbf{L} = [l_1 \ l_2]^T$ . The calculation of the  $\mathbf{L}$  gain can be done using the `acker` command in matlab:

$$\mathbf{L} = \text{acker}(\mathbf{A}', \mathbf{C}', [\exp(-p_1*ts), \ \exp(-p_2*ts)])';$$

with the following graphical representation

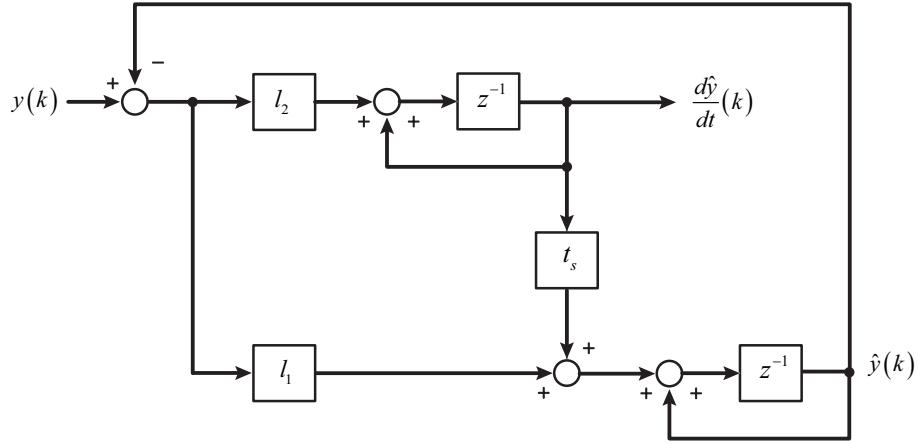


Figure 1.26: Double Integrator Observer in Discrete Time Domain.

The above example shows how to implement a derivative block by an double integrator observer. The use of the observer as derivative is often used in practical implementation.

Let's now consider the case of derivative of a rotating phase signal generated by the function  $\vartheta(t) = \text{atan}2(y(t), x(t))$  where

$$\begin{cases} x(t) = \cos(\omega t) \\ y(t) = \sin(\omega t) \end{cases} \quad (1.10.52)$$

and

$$\vartheta(t) = \text{atan}2(y(t), x(t)) \quad (1.10.53)$$

the variable  $\vartheta(t)$  is a saw-tooth curve bonded between  $-\pi$  to  $\pi$ ;

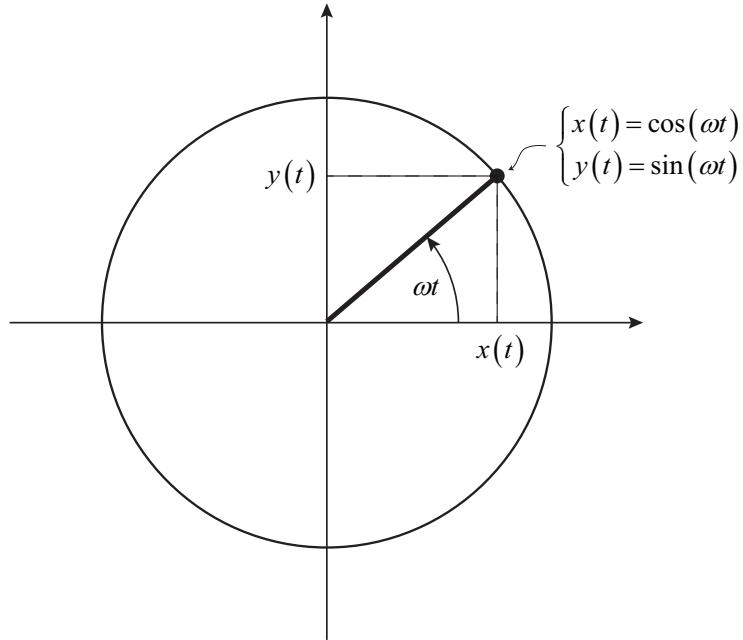


Figure 1.27: Phase generation.

A way to emulate the image (or codomain) of atan2() is the following function

$$\vartheta(t) = -\pi + \text{mod}(\omega t + 3\pi, 2\pi) \quad (1.10.54)$$

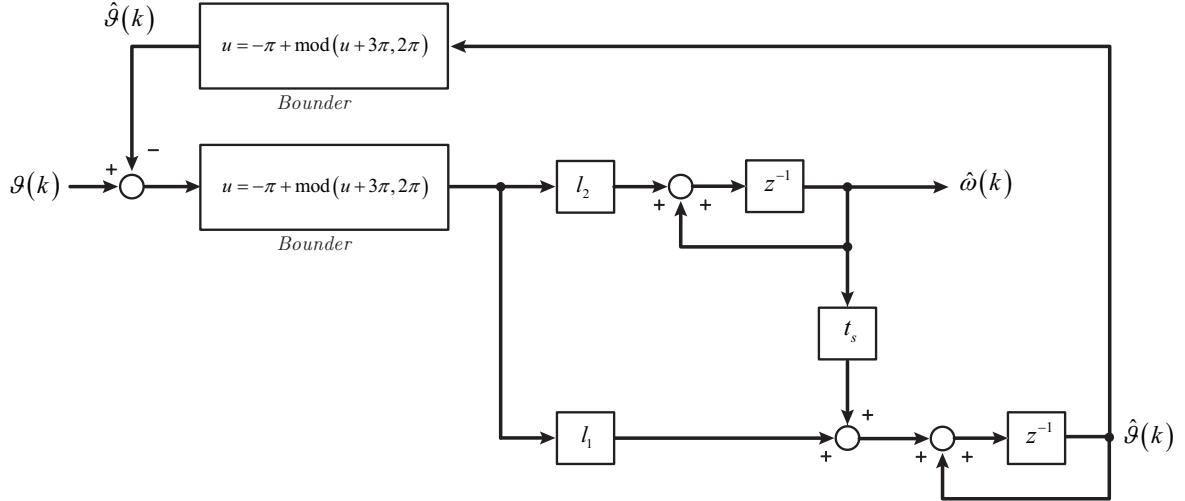
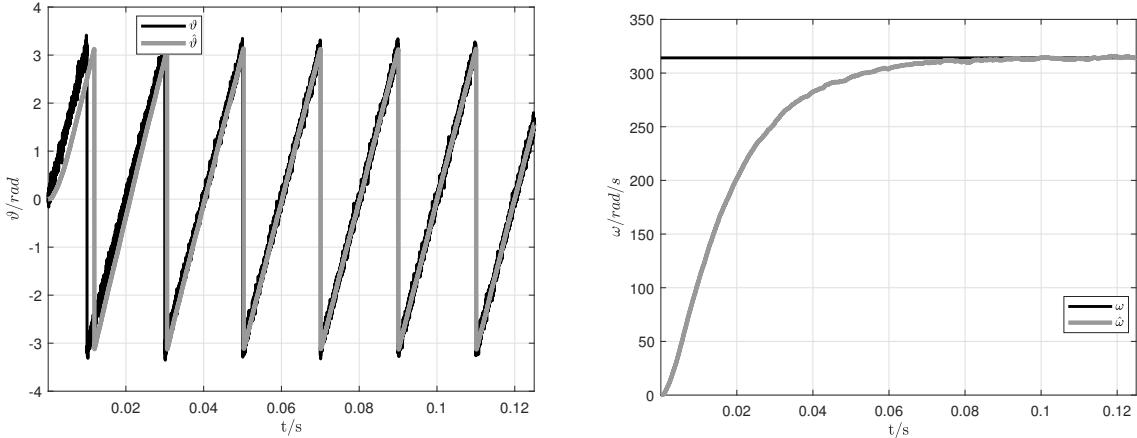


Figure 1.28: Double Integrator Observer in Discrete Time Domain with bounder block.

here some simulation results



(a) Phase  $\vartheta(t) + w(t)$ , where  $w(t)$  is white noise, and estimated phase  $\hat{\vartheta}(t)$

(b) Speed  $\omega(t)$  and estimated speed  $\hat{\omega}(t)$

Figure 1.29: Double integrator observer used as derivative applied to a rotating phase signal.

△

## 1.11 State Feedback with Integrator

### 1.11.1 Continuous-time Domain Case

The following control architecture is useful when it is required to track a given constant reference with zero steady state error.

The basic principle of the pole placement (or state feedback) with integrator is to insert an integrator in the feed-forward path between the error comparator and the plant, as shown in Figure 1.30.

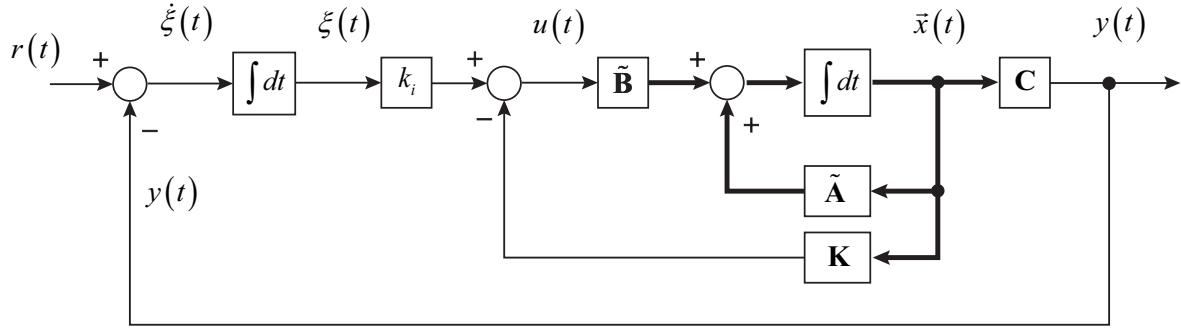


Figure 1.30: Servo system with state feedback and integral control - continuous time case

**Considering SISO case.**

From figure 1.30 we can write the following equations:

$$\begin{aligned} \dot{\vec{x}}(t) &= \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) \\ y(t) &= \mathbf{C}\vec{x}(t) \\ u(t) &= -\mathbf{K}\vec{x}(t) + k_i\xi \\ \dot{\xi}(t) &= r(t) - y(t) = r(t) - \mathbf{C}\vec{x}(t) \end{aligned} \tag{1.11.1}$$

where  $\vec{x}$  is the state vector of the plant.

We assume that  $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}})$  is completely state controllable. The set of equations reported in Eq. (1.11.1) can be rewritten as follows

$$\begin{bmatrix} \dot{\vec{x}}(t) \\ \dot{\xi}(t) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \vec{x}(t) \\ \xi(t) \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{B}} \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} r(t) \tag{1.11.2}$$

The control problem we are considering is basically a tracking problem, that means we want to analyse the behaviour of the tracking error  $r(t) - y(t)$ , in particular we shall design an asymptotically stable system such that  $\vec{x}_\infty$ ,  $\xi_\infty$  and  $y_\infty$  approach constant values respectively (we assume  $r(t) = r_\infty$  is constant). Then, at steady state, we obtain that  $\dot{\xi}(t) = 0$ , and results

in  $y_\infty = r_\infty$ . Notice that at steady state we have

$$\begin{bmatrix} \dot{\vec{x}}_\infty \\ \dot{\xi}_\infty \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \vec{x}_\infty \\ \xi_\infty \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{B}} \\ 0 \end{bmatrix} u_\infty + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} r_\infty \quad (1.11.3)$$

Noting, again, that  $r(t)$  is a step input, we have  $r(t) = r_\infty = r$  (constant) for  $t > 0$ . By subtracting Eq. (1.11.3) from Eq. 1.11.2, we obtain

$$\begin{bmatrix} \dot{\vec{x}}(t) - \dot{\vec{x}}_\infty \\ \dot{\xi}(t) - \dot{\xi}_\infty \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \vec{x}(t) - \vec{x}_\infty \\ \xi(t) - \xi_\infty \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{B}} \\ 0 \end{bmatrix} [u(t) - u_\infty] \quad (1.11.4)$$

Define (we apply a change of variable in order to remove the steady state off-set)

$$\begin{aligned} \vec{x}(t) - \vec{x}_\infty &= \Delta \vec{x}(t) \\ \xi(t) - \xi_\infty &= \Delta \xi(t) \\ u(t) - u_\infty &= \Delta u(t) \end{aligned} \quad (1.11.5)$$

The Eq. (1.11.4) can be written as

$$\begin{bmatrix} \Delta \dot{\vec{x}}(t) \\ \Delta \dot{\xi}(t) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \Delta \vec{x}(t) \\ \Delta \xi(t) \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{B}} \\ 0 \end{bmatrix} [\Delta u(t)] \quad (1.11.6)$$

where

$$\Delta u(t) = -\mathbf{K} \Delta \vec{x}(t) + k_i \Delta \xi(t) \quad (1.11.7)$$

The change of variable of Eq.(1.11.5) permit to convert a tracking problem into a regulation problem, that means the control problem reduces to find  $\mathbf{K}$  and  $k_i$  such that

$$\Delta \vec{x}(t) \xrightarrow[t \rightarrow \infty]{} 0 \quad (1.11.8)$$

$$\Delta \xi(t) \xrightarrow[t \rightarrow \infty]{} 0 \quad (1.11.9)$$

$$\Delta u(t) \xrightarrow[t \rightarrow \infty]{} 0 \quad (1.11.10)$$

Define a new  $(n + 1)$ th-order error vector  $\vec{e}(t)$  by

$$\vec{e}(t) = \begin{bmatrix} \Delta \vec{x}(t) \\ \Delta \xi(t) \end{bmatrix} = (n + 1)\text{-vector}$$

Then Eq. 1.11.6 becomes

$$\dot{\vec{e}}(t) = \tilde{\mathbf{A}}' \vec{e}(t) + \tilde{\mathbf{B}}' \Delta u(t) \quad (1.11.11)$$

where

$$\tilde{\mathbf{A}}' = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix}, \quad \tilde{\mathbf{B}}' = \begin{bmatrix} \tilde{\mathbf{B}} \\ 0 \end{bmatrix} \quad (1.11.12)$$

and Eq. (1.11.7) becomes

$$\begin{aligned} \Delta u(t) &= -\mathbf{K}' \vec{e}(t) \\ &= [\mathbf{K} \quad -k_i] \begin{bmatrix} \Delta \vec{x}(t) \\ \Delta \xi(t) \end{bmatrix} \\ &= -\mathbf{K} \Delta \vec{x}(t) + k_i \Delta \xi(t) \end{aligned} \quad (1.11.13)$$

where

$$\mathbf{K}' = [\mathbf{K} \quad -k_i] \quad (1.11.14)$$

The dynamic of the state vector  $\vec{e}(t)$  can be computed by the Eq. (1.11.15) which can be obtained by substituting Eq. (1.11.13) into Eq. (1.11.11), as follows

$$\dot{\vec{e}}(t) = (\tilde{\mathbf{A}}' - \tilde{\mathbf{B}}' \mathbf{K}') \vec{e}(t) \quad (1.11.15)$$

where  $\mathbf{K}'$  can be computed by poles placement.

**The design of the state feedback with integrator, in the continuous-time domain, reduces to the pole-placement problem:**

$$\left| s\mathbf{I} - (\tilde{\mathbf{A}}' - \tilde{\mathbf{B}}' \mathbf{K}') \right| = \sum_{k=1}^n (s - \mu_k) \quad (1.11.16)$$

where

$$\mathbf{K}' = [\mathbf{K} \quad -k_i] \quad (1.11.17)$$

and

$$\tilde{\mathbf{A}}' = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix}, \quad \tilde{\mathbf{B}}' = \begin{bmatrix} \tilde{\mathbf{B}} \\ 0 \end{bmatrix} \quad (1.11.18)$$

## 1.11.2 Discrete-time Representation of an Integrator

This subsection wants to describe how to represent an integrator operator in discrete time domain. At this aim the discretization of a PI control is proposed. The method which will be used is not the classical which can be find in literature, but it will give a kind of continuity with the tools up to now described.

Figure 1.31 shows the typical control architecture of a PI controller applied to a generic *Plant*.

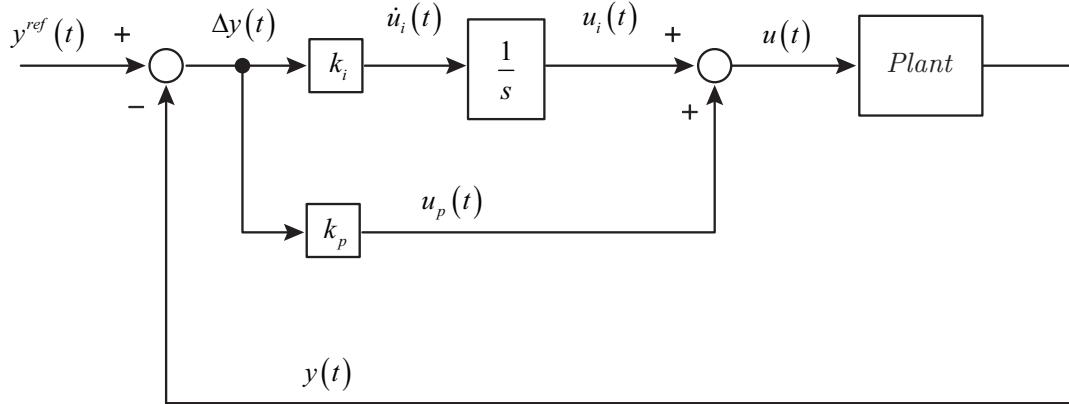


Figure 1.31: PI-controller architecture in continuous time domain.

Mathematically speaking the control structure of Figure 1.31 can be represented as follows

$$\Delta y(t) = y^{ref}(t) - y(t) \quad (1.11.19)$$

$$u(t) = u_p(t) + u_i(t) \quad (1.11.20)$$

$$u_p(t) = k_p \Delta y(t) \quad (1.11.21)$$

$$\dot{u}_i(t) = k_i \Delta y(t) \quad (1.11.22)$$

Eq. (1.11.22) can be correlated to the canonical representation of a linear system as follows

$$\dot{u}_i(t) = \tilde{\mathbf{A}} u_i(t) + \tilde{\mathbf{B}} \Delta y(t) \quad (1.11.23)$$

where

$$\tilde{\mathbf{A}} = 0 \quad (1.11.24)$$

$$\tilde{\mathbf{B}} = k_i \quad (1.11.25)$$

Now the discrete time representation of Eq. (1.11.23) can be computed applying the equations

$$\mathbf{A} = e^{\tilde{\mathbf{A}} t_s} \quad (1.11.26)$$

$$\mathbf{B} = \int_0^{t_s} e^{\tilde{\mathbf{A}} \tau} \tilde{\mathbf{B}} d\tau \quad (1.11.27)$$

which results in

$$u_i(k+1) = \mathbf{A} u_i(k) + \mathbf{B} \Delta y(k) \quad (1.11.28)$$

where

$$\mathbf{A} = e^{\tilde{\mathbf{A}}t_s} = 1 \quad (1.11.29)$$

$$\mathbf{B} = \int_0^{t_s} e^{\tilde{\mathbf{A}}\tau} \tilde{\mathbf{B}} d\tau = k_i t_s \quad (1.11.30)$$

Hence Eqs. (1.11.19)- (1.11.22) becomes

$$\Delta y(k) = y^{ref}(k) - y(k) \quad (1.11.31)$$

$$u(k) = u_p(k) + u_i(k) \quad (1.11.32)$$

$$u_p(k) = k_p \Delta y(k) \quad (1.11.33)$$

$$u_i(k+1) = u_i(k) + k_i t_s \Delta y(k) \quad (1.11.34)$$

as shown in Figure 1.32

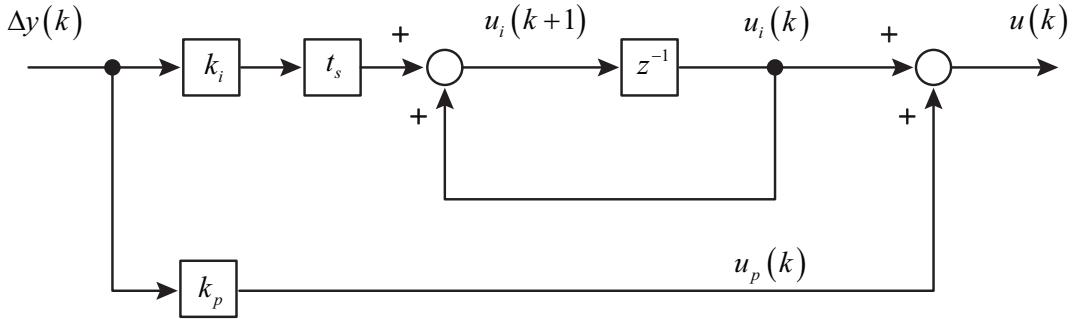


Figure 1.32: PI-controller architecture in discrete time domain.

In some application Eq. (1.11.34) can also be represented as follows

$$u_i(k) = u_i(k-1) + k_i t_s \Delta y(k) \quad (1.11.35)$$

as shown in Figure 1.33

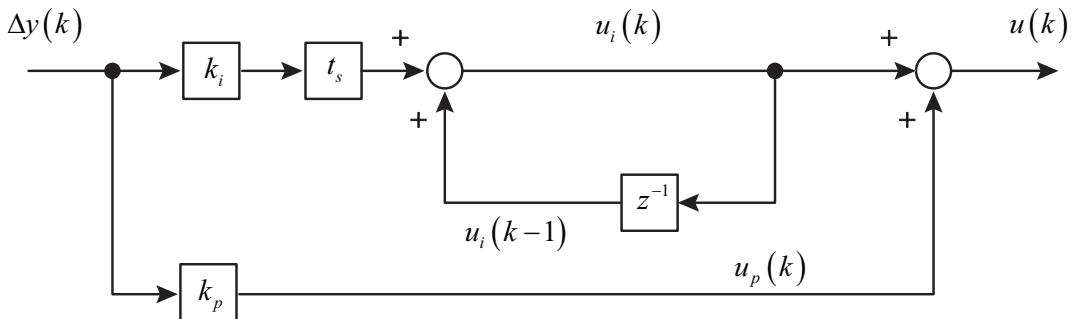


Figure 1.33: PI-controller architecture in discrete time domain - alternative representation.

### 1.11.3 Discrete-time Domain Case

A control system in which the output must follow a reference signals is called a *servo system*, that means a servo system problem is a *tracking problem*. In a servo, it is generally required that the system have one or more integrators within the closed loop to eliminate the steady state error. One way to introduce an integrator in the mathematical model of the closed loop system is to introduce a new state vector that integrates the difference between the reference vector  $\vec{r}$  and the output vector  $\vec{y}$ . Figure 1.34 shows a possible block diagram configuration for a servo system with state feedback and integral control. The integrator can be included as part of the pole placement formulation.

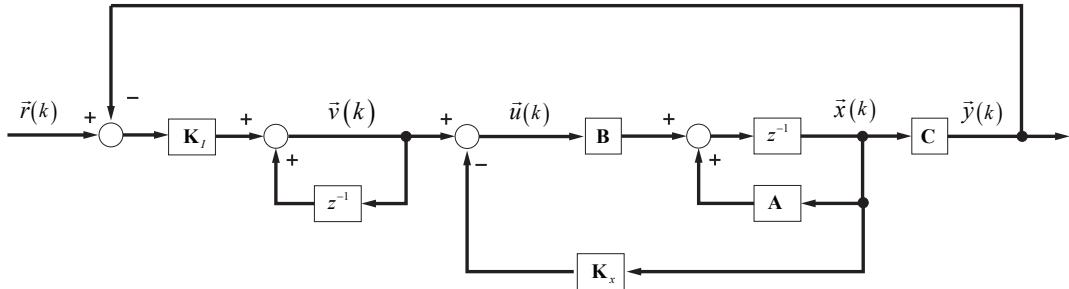


Figure 1.34: Servo system with state feedback and integral control

Considering the following state and output equation

$$\begin{aligned} \vec{x}(k+1) &= \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \\ \vec{y}(k) &= \mathbf{C}\vec{x}(k) \end{aligned} \tag{1.11.36}$$

where

$\vec{x}(k)$  = plant state vector ( $n$ -vector)

$\vec{u}(k)$  = control vector ( $m$ -vector)

$\vec{y}(k)$  = output vector ( $m$ -vector)

$\mathbf{A}$  =  $n \times n$  matrix

$\mathbf{B}$  =  $n \times m$  matrix

$\mathbf{C}$  =  $m \times n$  matrix

The integrator state equation is

$$\vec{v}(k) = \mathbf{K}_I [\vec{r}(k) - \vec{y}(k)] + \vec{v}(k-1) \tag{1.11.37}$$

where

$$\begin{aligned}\vec{v}(k) &= \text{actuating error vector } (m\text{-vector}) \\ \vec{r}(k) &= \text{command input vector } (m\text{-vector)}\end{aligned}\tag{1.11.38}$$

Eq. (1.11.37) can be rewritten as follows:

$$\vec{v}(k+1) = \mathbf{K}_I [\vec{r}(k+1) - \vec{y}(k+1)] + \vec{v}(k)\tag{1.11.39}$$

In steady state conditions which means  $\vec{v}(k+1) = \vec{v}(k) = \vec{v}_\infty$  Eq. (1.11.39) reduces to

$$\vec{r}(t) - \vec{y}_\infty = \vec{r}_\infty - \vec{y}_\infty = 0\tag{1.11.40}$$

The control vector  $\vec{u}(k)$  is given by

$$\begin{cases} \vec{v}(k) = \mathbf{K}_I [\vec{r}(k) - \vec{y}(k)] + \vec{v}(k-1) \\ \vec{u}(k) = -\mathbf{K}_x \vec{x}(k) + \vec{v}(k) \end{cases}\tag{1.11.41}$$

or

$$\begin{cases} \vec{v}(k+1) = \mathbf{K}_I [\vec{r}(k+1) - \vec{y}(k+1)] + \vec{v}(k) \\ \vec{u}(k+1) = -\mathbf{K}_x \vec{x}(k+1) + \vec{v}(k+1) \end{cases}\tag{1.11.42}$$

Our design parameters are matrices  $\mathbf{K}_x$  and  $\mathbf{K}_I$ . In what follows we shall discuss the procedure for determining matrices  $\mathbf{K}_x$  and  $\mathbf{K}_I$ .

The system dynamics may hence be described as follows

$$\vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) = \mathbf{A} \vec{x}(k) + \mathbf{B} [\vec{v}(k) - \mathbf{K}_x \vec{x}(k)]\tag{1.11.43}$$

and the integral term as follows

$$\begin{aligned}\vec{v}(k+1) &= \mathbf{K}_I [\vec{r}(k+1) - \vec{y}(k+1)] + \vec{v}(k) = \\ &= \mathbf{K}_I [\vec{r}(k+1) - \mathbf{C} \vec{x}(k+1)] + \vec{u}(k) + \mathbf{K}_x \vec{x}(k) = \\ &= \mathbf{K}_I [\vec{r}(k+1) - \mathbf{C} (\mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k))] + \vec{u}(k) + \mathbf{K}_x \vec{x}(k) = \\ &= [\mathbf{I}_m - \mathbf{K}_I \mathbf{C} \mathbf{B}] \vec{u}(k) + [\mathbf{K}_x - \mathbf{K}_I \mathbf{C} \mathbf{A}] \vec{x}(k) + \mathbf{K}_I \vec{r}(k+1)\end{aligned}\tag{1.11.44}$$

but  $\vec{v}(k+1) = \vec{u}(k+1) + \mathbf{K}_x \vec{x}(k+1)$  or

$$\begin{aligned}
 \vec{u}(k+1) &= \vec{v}(k+1) - \mathbf{K}_x \vec{x}(k+1) \\
 &= \vec{v}(k+1) - \mathbf{K}_x [\mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k)] \\
 &= [\mathbf{I}_m - \mathbf{K}_I \mathbf{C} \mathbf{B}] \vec{u}(k) + [\mathbf{K}_x - \mathbf{K}_I \mathbf{C} \mathbf{A}] \vec{x}(k) - \mathbf{K}_x [\mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k)] \\
 &\quad + \mathbf{K}_I \vec{r}(k+1)
 \end{aligned} \tag{1.11.45}$$

hence

$$\begin{aligned}
 \vec{u}(k+1) &= [\mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B}] \vec{u}(k) + \\
 &\quad + [\mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A}] \vec{x}(k) + \\
 &\quad + \mathbf{K}_I \vec{r}(k+1)
 \end{aligned} \tag{1.11.46}$$

Eq. (1.11.46) describes the evolution of the control input  $\vec{u}(k+1)$  as function previous control input  $\vec{u}(k)$  and by the state vector  $\vec{x}(k)$ , removing any relation with the integral corrective term  $\vec{v}(k)$ .

Combining  $\vec{x}(k+1)$  and  $\vec{u}(k+1)$  in matrix form we obtain

$$\begin{aligned}
 \begin{bmatrix} \vec{u}(k+1) \\ \vec{x}(k+1) \end{bmatrix} &= \begin{bmatrix} \mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B} & \mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A} \\ \mathbf{B} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \vec{u}(k) \\ \vec{x}(k) \end{bmatrix} + \\
 &\quad + \begin{bmatrix} \mathbf{K}_I \\ \mathbf{0} \end{bmatrix} \vec{r}(k+1) \\
 \vec{y}(k) &= \begin{bmatrix} 0 & \mathbf{C} \end{bmatrix} \begin{bmatrix} \vec{u}(k) \\ \vec{x}(k) \end{bmatrix}
 \end{aligned} \tag{1.11.47}$$

Note that the closed loop poles of the system are determined by the system itself and do not depend on the command input  $\vec{r}(k)$ . The eigenvalues of the state matrix in Eq. (1.11.47) determine the closed loop poles of the system. In order to apply the pole placement technique to the design of the servo system, consider the case where the command vector  $\vec{r}(k)$  is constant  $\vec{r}(k) \rightarrow \vec{r}$ , hence, Eq. (1.11.47) become

$$\begin{aligned}
 \begin{bmatrix} \vec{u}(k+1) \\ \vec{x}(k+1) \end{bmatrix} &= \begin{bmatrix} \mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B} & \mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A} \\ \mathbf{B} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \vec{u}(k) \\ \vec{x}(k) \end{bmatrix} + \\
 &\quad + \begin{bmatrix} \mathbf{K}_I \vec{r} \\ \mathbf{0} \end{bmatrix}
 \end{aligned} \tag{1.11.48}$$

In steady state conditions the quantities  $\vec{x}(k)$ ,  $\vec{u}(k)$  and  $\vec{v}(k)$  tend to constant value as

$$\begin{aligned}\vec{x}(k) &\rightarrow \vec{x}_\infty \\ \vec{u}(k) &\rightarrow \vec{u}_\infty \\ \vec{v}(k) &\rightarrow \vec{v}_\infty\end{aligned}\tag{1.11.49}$$

respectively. Thus from Eq. (1.11.37) we obtain the following equation at steady state

$$\vec{v}_\infty = \vec{v}_\infty + \vec{r} - \vec{y}_\infty\tag{1.11.50}$$

There is no steady state error in the output when the command input is a step vector. Hence Eq. (1.11.48) becomes

$$\begin{bmatrix} \vec{u}_\infty \\ \vec{x}_\infty \end{bmatrix} = \begin{bmatrix} \mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B} & \mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A} \\ \mathbf{B} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \vec{u}_\infty \\ \vec{x}_\infty \end{bmatrix} + \begin{bmatrix} \mathbf{K}_I \vec{r} \\ \mathbf{0} \end{bmatrix}\tag{1.11.51}$$

Let us define the error vectors by

$$\begin{aligned}\Delta \vec{x}(k) &= \vec{x}(k) - \vec{x}_\infty \\ \Delta \vec{u}(k) &= \vec{u}(k) - \vec{u}_\infty\end{aligned}\tag{1.11.52}$$

Again by applying Eq. (1.11.52) a tracking problem is reduced to a regulation problem, where

$$\Delta \vec{x}(k) \xrightarrow[k \rightarrow \infty]{} \vec{0}\tag{1.11.53}$$

$$\Delta \vec{u}(k) \xrightarrow[k \rightarrow \infty]{} \vec{0}\tag{1.11.54}$$

Then subtracting Eq. (1.11.51) from Eq. (1.11.48) we obtain

$$\begin{bmatrix} \Delta \vec{u}(k+1) \\ \Delta \vec{x}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B} & \mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A} \\ \mathbf{B} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \Delta \vec{u}(k) \\ \Delta \vec{x}(k) \end{bmatrix}\tag{1.11.55}$$

The dynamics of the system are determined by the eigenvalues of the state matrix appearing

in Eq. (1.11.55). Eq (1.11.55) can be modified as following

$$\begin{bmatrix} \Delta \vec{u}(k+1) \\ \Delta \vec{x}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{A} \end{bmatrix}}_{\mathbf{A}'} \begin{bmatrix} \Delta \vec{u}(k) \\ \Delta \vec{x}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix}}_{\mathbf{B}'} \vec{w}(k) \quad (1.11.56)$$

where

$$\vec{w}(k) = \begin{bmatrix} \mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B} & \mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A} \end{bmatrix} \begin{bmatrix} \Delta \vec{u}(k) \\ \Delta \vec{x}(k) \end{bmatrix} \quad (1.11.57)$$

If we define

$$\begin{aligned} \vec{\xi}(k) &= \begin{bmatrix} \Delta \vec{u}(k) \\ \Delta \vec{x}(k) \end{bmatrix} = (n+m)\text{-vector} \\ \mathbf{A}' &= \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{A} \end{bmatrix} = (n+m) \times (n+m) \text{ matrix} \\ \mathbf{B}' &= \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} = (n+m) \times m \text{ matrix} \\ \mathbf{K}' &= - \begin{bmatrix} \mathbf{I}_m - \mathbf{K}_x \mathbf{B} - \mathbf{K}_I \mathbf{C} \mathbf{B} & \mathbf{K}_x - \mathbf{K}_x \mathbf{A} - \mathbf{K}_I \mathbf{C} \mathbf{A} \end{bmatrix} = m \times (n+m) \text{ matrix} \end{aligned}$$

The Eq. (1.11.56) and (1.11.57) become, respectively

$$\begin{aligned} \vec{\xi}(k+1) &= \mathbf{A}' \vec{\xi} + \mathbf{B}' \vec{w}(k) \\ \vec{w}(k) &= -\mathbf{K}' \vec{\xi}(k) \end{aligned} \quad (1.11.58)$$

Once the desired closed loop poles are specified, matrix  $\mathbf{K}'$  can be determined by the pole placement technique, e.g. by use of Ackermann's formula

$$\begin{aligned} \mathbf{K}' &= \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \end{bmatrix} \mathbf{M}'^{-1} q(\mathbf{A}') \\ \mathbf{M}' &= \begin{bmatrix} \mathbf{B}' & \mathbf{A}' \mathbf{B}' & \dots & (\mathbf{A}')^{n+m-1} \mathbf{B}' \end{bmatrix} \end{aligned}$$

Once matrix  $\mathbf{K}'$  is derived via pole placement or other methods, the matrices  $\mathbf{K}_x$  and  $\mathbf{K}_I$  can be obtained as follows.

Note that

$$\begin{bmatrix} \mathbf{K}_I & \mathbf{K}_x \end{bmatrix} \begin{bmatrix} \mathbf{C}\mathbf{B} & \mathbf{C}\mathbf{A} \\ \mathbf{B} & \mathbf{A} - \mathbf{I}_n \end{bmatrix} = \begin{bmatrix} \mathbf{K}_x\mathbf{B} + \mathbf{K}_I\mathbf{C}\mathbf{B} & -\mathbf{K}_x + \mathbf{K}_x\mathbf{A} + \mathbf{K}_I\mathbf{C}\mathbf{A} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \mathbf{K}' + \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \end{bmatrix} \quad (1.11.59)$$

the desired gain matrices  $\begin{bmatrix} \mathbf{K}_I & \mathbf{K}_x \end{bmatrix}$  may be calculated according to

$$\begin{bmatrix} \mathbf{K}_I & \mathbf{K}_x \end{bmatrix} = \left\{ \mathbf{K}' + \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \end{bmatrix} \right\} \begin{bmatrix} \mathbf{C}\mathbf{B} & \mathbf{C}\mathbf{A} \\ \mathbf{B} & \mathbf{A} - \mathbf{I}_n \end{bmatrix}^{-1} \quad (1.11.60)$$

**The design of state feedback with integrator, in the discrete-time domain, reduces to a pole placement problem applied to the state feedback system**

$$\vec{\xi}(k+1) = \mathbf{A}'\vec{\xi} + \mathbf{B}'\vec{w}(k) \quad (1.11.61)$$

$$\vec{w}(k) = -\mathbf{K}'\vec{\xi}(k) \quad (1.11.62)$$

where

$$\mathbf{A}' = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{A} \end{bmatrix}, \quad \mathbf{B}' = \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix}, \quad \vec{\xi}(k) = \begin{bmatrix} \Delta\vec{u}(k) \\ \Delta\vec{x}(k) \end{bmatrix} \quad (1.11.63)$$

In particular, the control problem, reduces to the derivation of the preliminary state feedback matrix  $\mathbf{K}'$  by pole placement or other techniques like linear quadratic regulator, e.g.

$$[\mathbf{K}'] = \text{dlqr}(\mathbf{A}', \mathbf{B}', \mathbf{Q}, \mathbf{R}) \quad (1.11.64)$$

or

$$[\mathbf{K}'] = \text{acker}(\mathbf{A}', \mathbf{B}', [\text{zpoles}]) \quad (1.11.65)$$

Once the matrix gain  $\mathbf{K}'$  is calculated the effective state feedback matrix  $\begin{bmatrix} \mathbf{K}_I & \mathbf{K}_x \end{bmatrix}$  can be derived by the transformation

$$\begin{bmatrix} \mathbf{K}_I & \mathbf{K}_x \end{bmatrix} = \left\{ \mathbf{K}' + \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \end{bmatrix} \right\} \begin{bmatrix} \mathbf{C}\mathbf{B} & \mathbf{C}\mathbf{A} \\ \mathbf{B} & \mathbf{A} - \mathbf{I}_n \end{bmatrix}^{-1} \quad (1.11.66)$$

**State observer inclusion.** When state vector is not measurable, condition which frequently occur, the use of a state observer is necessary. Figure 1.35 shows a possible architecture including in addition to the state feedback with integrator control, a state observer block.

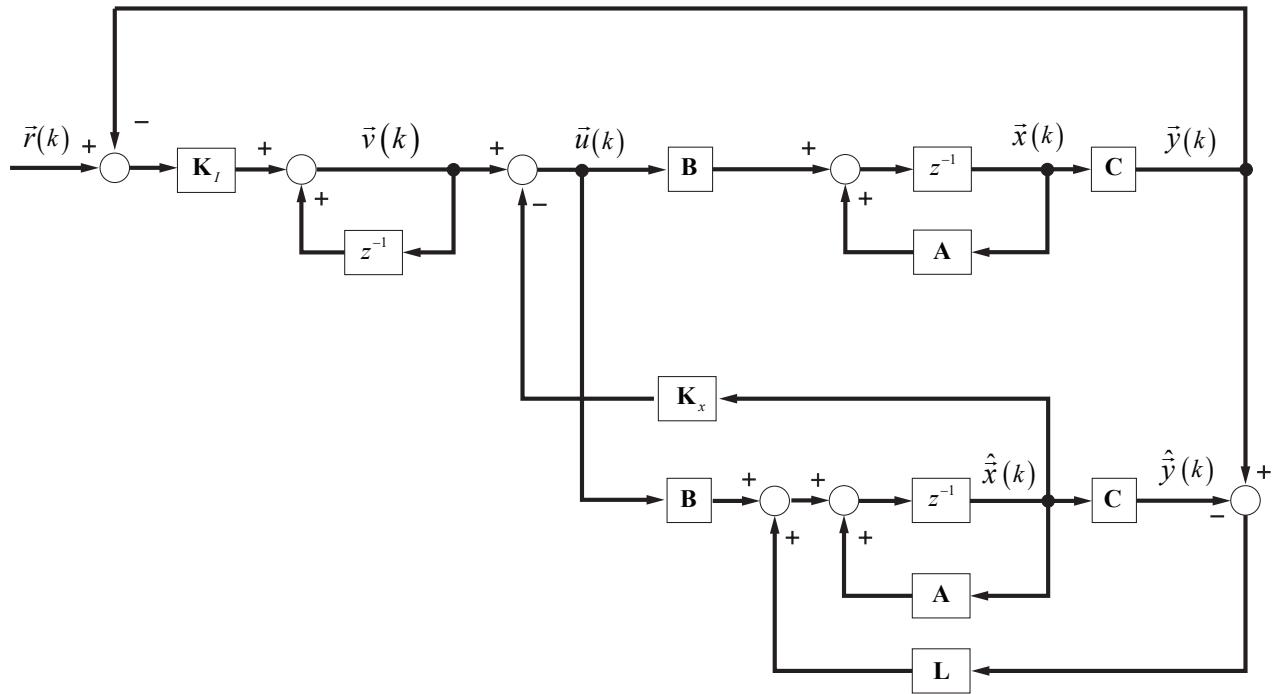


Figure 1.35: Servo system with observed state feedback and integral control

The same considerations hold for the case SISO where  $r(t)$ ,  $y(t)$  and  $v(t)$  are scalars and  $\mathbf{K}_I$  reduces to  $k_i$ , see also Figure 1.36 and Figure 1.37

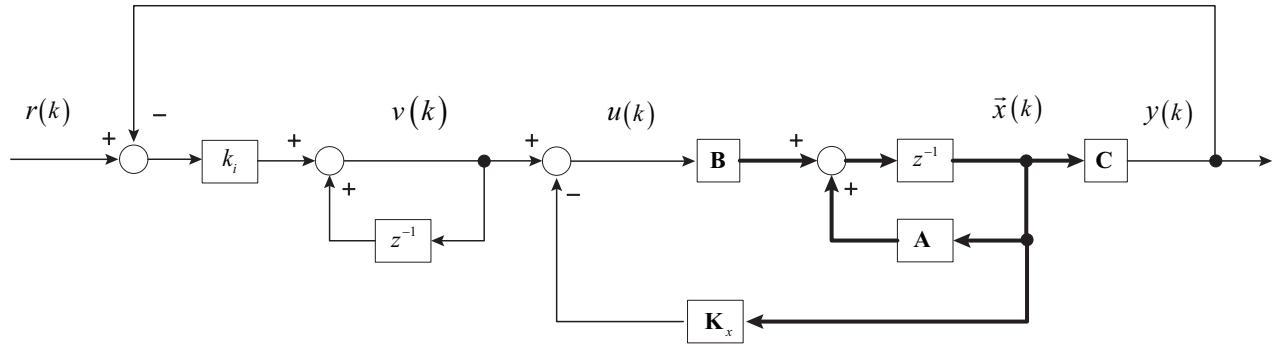


Figure 1.36: Servo system with observed state feedback and integral control. SISO case.

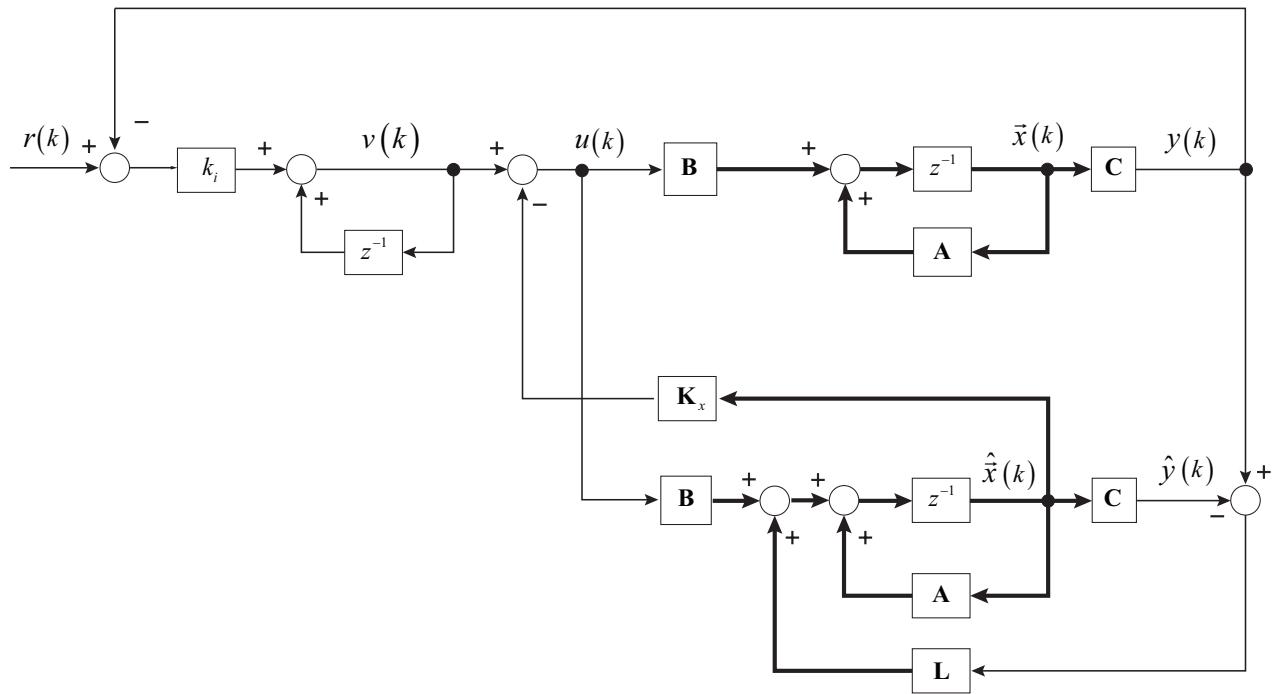


Figure 1.37: Servo system with observed state feedback and integral control. SISO case.

# Chapter 2

## Ljapunov Stability

This lecture covers some arguments regarding the Ljapunov stability and its application in design of control system. We start with an overview on phase portrait representation and after that, we will face up to the basic concept of Ljapunov's direct method applied to autonomous system. Main references are [?] and [?].

### 2.1 Some backgrounds

Consider the pendulum of Figure 2.1

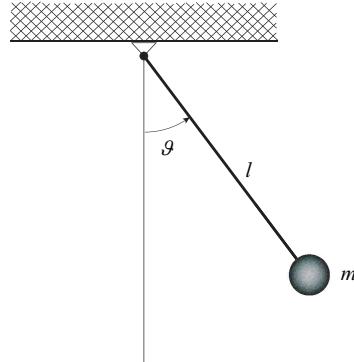


Figure 2.1: Pendulum.

Where the kinetic and potential energy can be expressed as follows

$$E_{kin} = \frac{1}{2}ml^2\dot{\vartheta}^2$$

$$E_{pot} = mgl(1 - \cos \vartheta)$$

Applying the Euler-Lagrange equation we obtain

$$\frac{d}{dt} \left( \frac{\partial E_{kin}}{\partial \dot{q}_i} \right) - \frac{\partial E_{kin}}{\partial q_i} + \frac{\partial E_{pot}}{\partial q_i} = Q_{diss}$$

where  $Q_{diss} = -b\dot{\vartheta}$  which is a force due to the friction and results in the following second order differential equation

$$ml^2\ddot{\vartheta} + mgl \sin \vartheta = -b\dot{\vartheta} \quad (2.1.1)$$

which in state space form results as follows

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{g}{l} \sin x_1 - \frac{b}{ml} x_2 \end{cases} \quad (2.1.2)$$

Now we consider the case of small oscillation introducing the approximation  $\sin \vartheta \approx \vartheta$  and we consider the case  $b = 0$  and  $g = l$  which results in the following harmonic oscillator

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ \vec{x}(0) &= \begin{bmatrix} 1 & 0 \end{bmatrix}^T \end{aligned} \quad (2.1.3)$$

where  $\vec{x}(0)$  is the initial condition,  $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$  and  $\vec{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T$ .

The solution of the dynamical system is

$$\vec{x}(t) = e^{\mathbf{A}t} \vec{x}(0) \quad (2.1.4)$$

and can be solved as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1} \left\{ \left[ s\mathbf{I} - \mathbf{A} \right]^{-1} \right\} \vec{x}(0) = \mathcal{L}^{-1} \left\{ \frac{1}{s^2 + 1} \begin{bmatrix} s & 1 \\ -1 & s \end{bmatrix} \right\} \vec{x}(0) = \\ &= \begin{bmatrix} \cos(t) & \sin(t) \\ -\sin(t) & \cos(t) \end{bmatrix} \vec{x}(0) = \begin{bmatrix} \cos(t) & -\sin(t) \end{bmatrix}^T \end{aligned} \quad (2.1.5)$$

The solution  $\vec{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T = \begin{bmatrix} \cos(t) & -\sin(t) \end{bmatrix}^T$  can be plotted in the plane  $(x_1, x_2)$  - which is called *phase-portrait*. The solution  $\vec{x}(t)$  is the parametric representation of a circle with unitary radius. Selecting any initial condition different from zero, we can observe that the trajectory state vector is stationary - there isn't any loss of energy - and the motus is a periodic cycles as in a harmonic oscillator, see also Figure 2.2.

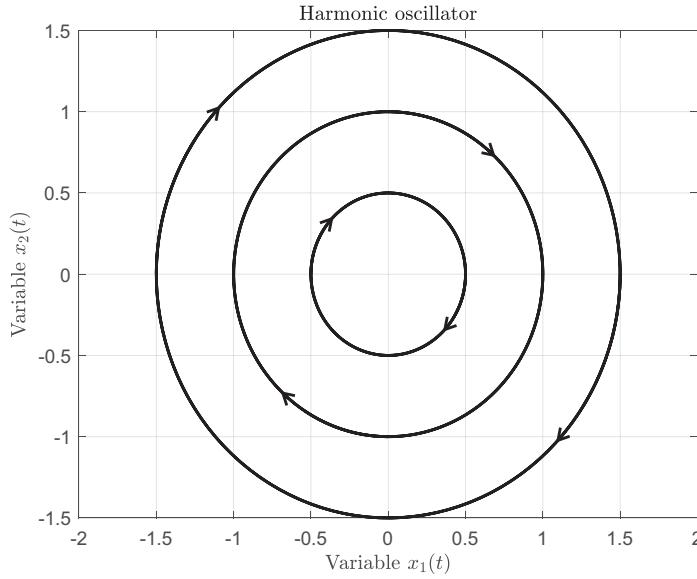


Figure 2.2: Phase portrait of the harmonic oscillator for different initial conditions.

Consider now the damped version (with friction factor like  $b = \frac{1}{2}$ ) of the harmonic oscillator

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (2.1.6)$$

$$\vec{x}(0) = [1 \ 0]^T$$

where  $\vec{x}(0)$  is the initial condition and  $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & -\frac{1}{2} \end{bmatrix}$  and  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$ .

The solution of the dynamical system is

$$\vec{x}(t) = e^{\mathbf{A}t} \vec{x}(0) \quad (2.1.7)$$

and can be solved as follows

$$\begin{aligned} \vec{x}(t) &= \mathcal{L}^{-1} \left\{ \left[ s\mathbf{I} - \mathbf{A} \right]^{-1} \right\} \vec{x}(0) = \mathcal{L}^{-1} \left\{ \frac{1}{2s^2 + s + 2} \begin{bmatrix} 2s+1 & 2 \\ -2 & 2s \end{bmatrix} \right\} \vec{x}(0) = \\ &= \begin{bmatrix} e^{-\alpha t} [\gamma \cos(\beta t) + \sin(\beta t)] & e^{-\alpha t} \sin(\beta t) \\ -e^{-\alpha t} \sin(\beta t) & e^{-\alpha t} [\gamma \cos(\beta t) - \sin(\beta t)] \end{bmatrix} \vec{x}(0) \quad (2.1.8) \\ &= \begin{bmatrix} e^{-\alpha t} (\gamma \cos(\beta t) + \sin(\beta t)) & -e^{-\alpha t} \sin(\beta t) \end{bmatrix}^T \end{aligned}$$

where  $\alpha, \beta, \gamma \in \mathbb{R}^+$ .

The plotting of the solution  $\vec{x}(t) = [x_1(t) \ x_2(t)]^T$  in the plane  $(x_1, x_2)$  is shown in Figure 2.3, where its sink nature is noticeable. Starting from any initial condition the trajectory of the state vector goes asymptotically to the origin.

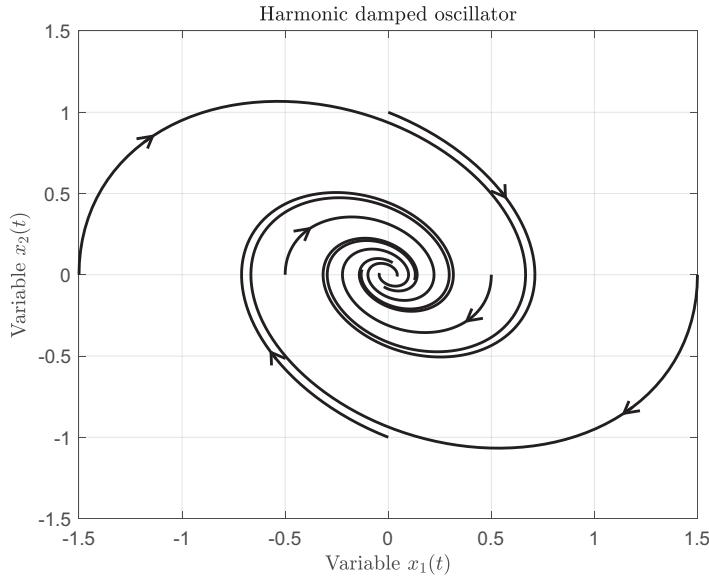


Figure 2.3: Phase portrait of the harmonic damped oscillator.

### 2.1.1 Equilibrium Points in Continuous-time Domain

Let us consider the following continuous time linear system

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}\vec{u}(t) \\ \vec{y}(t) = \tilde{\mathbf{C}}\vec{x}(t) + \tilde{\mathbf{D}}\vec{u}(t) \end{cases} \quad (2.1.9)$$

The equilibrium points  $\vec{x}_e$  of the system when the input is constant  $\vec{u} = \vec{u}_e$  can be determined imposing  $\dot{\vec{x}} = 0$

$$\tilde{\mathbf{A}}\vec{x}_e + \tilde{\mathbf{B}}\vec{u}_e = 0, \quad \rightarrow \quad \boxed{\vec{x}_e = -\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}\vec{u}_e} \quad (2.1.10)$$

If matrix  $\tilde{\mathbf{A}}$  is invertible, the system has only one equilibrium point  $\vec{x}_e$ .

If matrix  $\tilde{\mathbf{A}}$  is singular (that is if matrix  $\tilde{\mathbf{A}}$  has at least one eigenvalue in the origin) we can have two different cases:

- *There are infinite equilibrium points.* This situation happens when  $\text{rank} [\tilde{\mathbf{A}}] = \text{rank} [\tilde{\mathbf{A}} \ \tilde{\mathbf{B}}]$ .
- *There are no equilibrium points.* This situation happens when  $\text{rank} [\tilde{\mathbf{A}}] \neq \text{rank} [\tilde{\mathbf{A}} \ \tilde{\mathbf{B}}]$ .

The output  $\vec{y}_e$  corresponding to the particular equilibrium point  $(\vec{x}_e, \vec{u}_e)$  can be directly obtained using the output equation:

$$\vec{y}_e = \tilde{\mathbf{C}}\vec{x}_e + \tilde{\mathbf{D}}\vec{u}_e \quad (2.1.11)$$

If the matrix  $\tilde{\mathbf{A}}$  is invertible, the following relation holds:

$$\begin{aligned}\vec{y}_e &= -\tilde{\mathbf{C}}\tilde{\mathbf{A}}\tilde{\mathbf{B}}\vec{u}_e + \tilde{\mathbf{D}}\vec{u}_e = \left[ \tilde{\mathbf{C}} \left( s\mathbf{I} - \tilde{\mathbf{A}} \right)^{-1} \tilde{\mathbf{B}} + \tilde{\mathbf{D}} \right]_{s=0} \\ &= \mathbf{H}(\mathbf{0})\vec{u}_e\end{aligned}\quad (2.1.12)$$

where  $\mathbf{H}(s)$  denotes the transfer matrix of the system.

For linear systems the stability of an equilibrium point does not depend on the point itself, but it depends on the stability of the system which is completely determined by the position of the eigenvalue of matrix  $\tilde{\mathbf{A}}$ .

### 2.1.2 Equilibrium Points in Discrete-time Domain

Let us consider the following discrete-time linear system

$$\begin{cases} \vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \\ \vec{y}(k) = \mathbf{C}\vec{x}(k) + \mathbf{D}\vec{u}(k) \end{cases} \quad (2.1.13)$$

When the input is constant  $\vec{u}(t) = \vec{u}_e$ , the equilibrium points  $\vec{x}_e$  of the system can be determined imposing  $\vec{x}(k+1) = \vec{x}(k) = \vec{x}_e$ :

$$\vec{x}_e = \mathbf{A}\vec{x}_e + \mathbf{B}\vec{u}_e, \quad \rightarrow \quad \boxed{\vec{x}_e = (\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}\vec{u}_e} \quad (2.1.14)$$

In this case the system has only one equilibrium point  $\vec{x}_e$ , if and only if matrix  $(\mathbf{I} - \mathbf{A})$  is invertible.

If matrix  $(\mathbf{I} - \mathbf{A})$  is singular (that is if matrix  $\mathbf{A}$  has at least one eigenvalue in  $z = 1$ ), then the system:

- *has infinite equilibrium points* if  $\text{Rank}[\mathbf{I} - \mathbf{A}] = \text{Rank}[\mathbf{I} - \mathbf{A} \ \mathbf{B}]$
- *does not have equilibrium points* if  $\text{Rank}[\mathbf{I} - \mathbf{A}] \neq \text{Rank}[\mathbf{I} - \mathbf{A} \ \mathbf{B}]$

The output value  $\vec{y}_e$  corresponding to the equilibrium point  $(\vec{x}_e, \vec{u}_e)$  can be determined as follows:

$$\vec{y}_e = \mathbf{C}\vec{x}_e + \mathbf{D}\vec{u}_e \quad (2.1.15)$$

If matrix  $(\mathbf{I} - \mathbf{A})$  is invertible, then the following relation holds:

$$\begin{aligned}\vec{y}_e &= \mathbf{C}(\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}\vec{u}_e + \mathbf{D}\vec{u}_e = \left[ \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D} \right]_{z=1} \\ &= \mathbf{H}(z)_{z=1} \vec{u}_e\end{aligned}\quad (2.1.16)$$

where  $\mathbf{H}(z)$  denotes the transfer matrix of the discrete system.

The static gain  $\mathbf{H}_{z=1}$  of the transfer matrix  $\mathbf{H}(z)$  is infinite if matrix  $\mathbf{I} - \mathbf{A}$  has at least one eigenvalue in the origin, that is if matrix  $\mathbf{A}$  has at least one eigenvalue in  $z = 1$ .

## 2.2 Fundamentals of Ljapunov Theory.

Consider the autonomous system

$$\dot{\vec{x}} = f(\vec{x}) \quad (2.2.1)$$

where  $f : D \rightarrow \mathbb{R}^n$  is a local Lipschitz map from a domain  $D \subset \mathbb{R}^n$  into  $\mathbb{R}^n$ . Suppose  $\vec{x}_e \in D$  is an equilibrium point of Eq. (2.2.1); that is  $f(\vec{x}_e) = \vec{0}$ . Our goal is to characterize and study the stability of  $\vec{x}_e$ . For convenience, we consider the case when the equilibrium point is at the origin of  $\mathbb{R}^n$ ; that is  $\vec{x}_e = \vec{0}$ . There is no loss of generality in doing so because any equilibrium point can be shifted to the origin via a change of variables. Suppose  $\vec{x}_e \neq \vec{0}$  and consider the change of variables  $\vec{y} = \vec{x} - \vec{x}_e$ . The derivative of  $\vec{y}$  is given by

$$\dot{\vec{y}} = \dot{\vec{x}} = f(\vec{x}) = f(\vec{y} + \vec{x}_e) = g(\vec{y}), \quad \text{where } g(\vec{0}) = \vec{0} \quad (2.2.2)$$

In the new variable  $\vec{y}$ , the system has equilibrium at the origin. Therefore, without loss of generality, we will always assume that  $f(\vec{x})$  satisfies  $f(\vec{0}) = \vec{0}$  and study the stability of the origin  $\vec{x} = \vec{0}$ .

**Definition 2.1.** *The equilibrium point  $\vec{x} = \vec{0}$  of Eq. (2.2.1) is*

- *stable if, for each  $\varepsilon > 0$ , there is  $\delta(\varepsilon) > 0$  such that*

$$\|\vec{x}(0)\| < \delta \Rightarrow \|\vec{x}(t)\| < \varepsilon, \quad \forall t \geq 0$$

- *unstable if it is not stable*
- *asymptotically stable if it is stable and  $\delta$  can be chosen such that*

$$\|\vec{x}(0)\| < \delta \Rightarrow \lim_{t \rightarrow \infty} \vec{x}(t) = \vec{0}$$

□

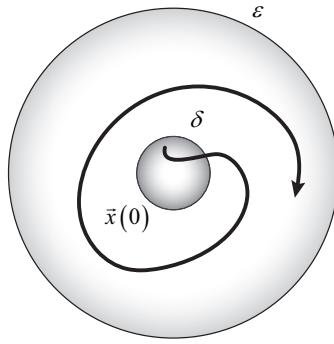


Figure 2.4: Ljapunov stability criterion.

To demonstrate that the origin is stable, then, for any value of  $\varepsilon$  we must produce a value of  $\delta$  such that a trajectory starting in a  $\delta$  neighbourhood of the origin will never leave the  $\varepsilon$  neighbourhood. The three type of stability properties can be illustrated by pendulum example. The pendulum of equation

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\alpha \sin x_1 - \beta x_2 \end{cases}$$

has two equilibrium points at  $(x_1 = 0, x_2 = 0)$  and  $(x_1 = \pi, x_2 = 0)$ . Neglecting friction, by setting  $\beta = 0$  we have that trajectories in the neighbourhood of the first equilibrium are closed orbits. Therefore, by starting sufficiently close to the equilibrium point, trajectories can be guaranteed to stay within any specified ball centred at the equilibrium point. Hence, the  $\varepsilon - \delta$  requirement for stability is satisfied. The equilibrium point, however, is not asymptotically stable since trajectories starting from a not equilibrium point do not tend to it eventually. Instead they remain in their closed orbits. When friction is taken into consideration ( $\beta > 0$ ), the equilibrium point at the origin becomes a stable focus. Inspection of the phase portrait of a stable focus shows that the  $\varepsilon - \delta$  requirement for stability is satisfied. In addition, trajectories starting close to the equilibrium point tend to it as  $t \rightarrow \infty$ . The second equilibrium point at  $x_1 = \pi$  is saddle point. Clearly the  $\varepsilon - \delta$  requirement cannot be satisfied since, for any  $\varepsilon > 0$ , there is always a trajectory that will leave the ball  $\{\vec{x} \in \mathbb{R}^n \mid \|\vec{x} - \vec{x}_e\| \leq \varepsilon\}$  even when  $\vec{x}(0)$  is arbitrarily close to the equilibrium point  $\vec{x}_e$ .

The conclusion we reached about the stable equilibrium point of the pendulum can also be reached by using energy concepts. Let us define the energy of the pendulum  $H(\vec{x})$  as sum of its potential and kinetic energies, with the reference of the potential energy chosen such that  $H(\vec{0}) = 0$ ; that is

$$H(\vec{x}) = \alpha(1 - \cos x_1) + \frac{1}{2}x_2^2$$

When friction is neglected ( $\beta = 0$ ), the system is conservative; that is, there is no dissipation of energy. Hence,  $H = \text{constant}$  during the motion of the system or, in other words,  $dH/dt = 0$  **along the trajectories of the system**. Since  $H(\vec{x}) = c$  forms a closed contour around  $\vec{x} = \vec{0}$  for small  $c$ , we can again arrive at the conclusion that  $\vec{x} = \vec{0}$  is a stable equilibrium point. When friction is accounted for ( $\beta > 0$ ), energy will be dissipated during the motion of the system, that

is,  $dH/dt \leq 0$  along the trajectories of the system. Due to friction,  $H$  cannot remain constant indefinitely while the system is in motion. Hence, it keeps decreasing until it eventually reaches zero, showing that the trajectory tends to  $\vec{x} = \vec{0}$  as  $t \rightarrow \infty$ . Thus, by examining the derivative of  $H$  along the trajectories of the system, it is possible to determine the stability of the equilibrium point. In 1892, Ljapunov showed that certain other function could be used instead of energy to determine stability of an equilibrium point. Let  $V : D \rightarrow \mathbb{R}$  be a continuously differentiable function defined in a domain  $D \subset \mathbb{R}^n$  that contains the origin. The derivative of  $V$  along the trajectories of Eq. (2.2.1) denoted by  $\dot{V}(\vec{x})$ , is given by

$$\begin{aligned}\dot{V}(\vec{x}) &= \sum_{i=1}^n \frac{\partial V}{\partial x_i} \dot{x}_i = \sum_{i=1}^n \frac{\partial V}{\partial x_i} f_i(\vec{x}) \\ &= \begin{bmatrix} \frac{\partial V}{\partial x_1} & \frac{\partial V}{\partial x_2} & \dots & \frac{\partial V}{\partial x_n} \end{bmatrix} \begin{bmatrix} f_1(\vec{x}) \\ f_2(\vec{x}) \\ \vdots \\ f_n(\vec{x}) \end{bmatrix} = \frac{\partial V}{\partial \vec{x}} f(\vec{x})\end{aligned}\quad (2.2.3)$$

**Theorem 2.1.** Let  $\vec{x} = \vec{0}$  be an equilibrium point for Eq. (2.2.1) and  $D \subset \mathbb{R}^n$  be a domain containing  $\vec{x} = \vec{0}$ . Let  $V : D \rightarrow \mathbb{R}$  be a continuous differentiable function such that

$$\begin{aligned}V(\vec{0}) &= 0 \quad \text{and} \quad V(\vec{x}) > 0 \quad \text{in} \quad D - \{\vec{0}\} \\ \dot{V}(\vec{x}) &\leq 0 \in D\end{aligned}\quad (2.2.4)$$

Then,  $\vec{x} = \vec{0}$  is stable. Moreover, if

$$\dot{V}(\vec{x}) < 0 \quad \text{in} \quad D - \{\vec{0}\} \quad (2.2.5)$$

then  $\vec{x} = \vec{0}$  is asymptotically stable. □

A continuously differentiable function  $V(\vec{x})$  satisfying Eq. (2.2.4) and Eq. (2.2.5) is called *Ljapunov function*. The surface  $V(\vec{x}) = c$ , for some  $c > 0$ , is called a *Ljapunov level surface*.

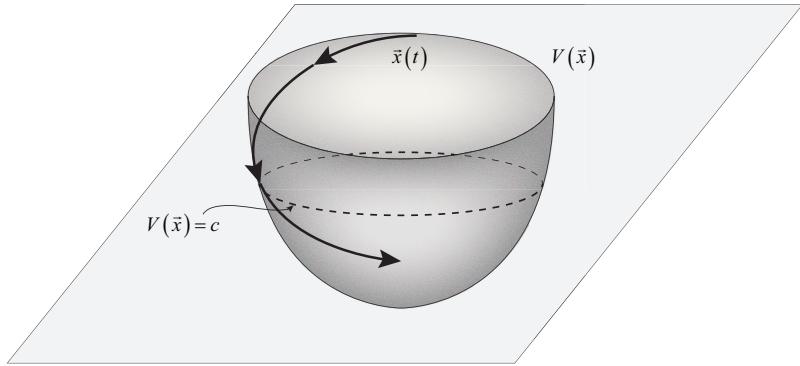


Figure 2.5: Ljapunov function  $V(\vec{x})$  and the Ljapunov level surface  $V(\vec{x}) = c$  (dashed).

Using Ljapunov surfaces, we notice that Figure 2.5 makes the theorem intuitively clear. It shows Ljapunov surface for increasing values of  $c$ . The condition  $\dot{V}(\vec{x}) \leq 0$  implies that when a trajectory crosses a Ljapunov level surface  $V(\vec{x}) = c$ , it moves inside the set  $\Omega_c = \{\vec{x} \in \mathbb{R}^n \mid V(\vec{x}) \leq c\}$  and can never come out again. When  $\dot{V} < 0$  the trajectory moves from one Ljapunov level surface to an inner Ljapunov level surface with a smaller  $c$ . As  $c$  decreases, the Ljapunov level surface  $V(\vec{x}) = c$  shrinks to the origin, showing that the trajectory approaches the origin as time progresses.

**Definition 2.2.** A function  $V(\vec{x})$  satisfying the condition Eq 2.2.4 that is,  $V(\vec{0}) = 0$  and  $V(\vec{x}) > 0$  for  $\vec{x} \neq \vec{0}$  is said to be **positive definite**. If it satisfies the weaker condition  $V(\vec{x}) \geq 0$  for  $\vec{x} \neq \vec{0}$ , it is said to be **positive semidefinite**. A function  $V(\vec{x})$  is said to be **negative definite** or **negative semidefinite** if  $V(\vec{x})$  is negative definite or negative semidefinite, respectively.  $\square$

Now we can rephrase the Ljapunov's theorem to say that *the origin is stable if there is a continuous differentiable positive definite function  $V(\vec{x})$  so that  $\dot{V}(\vec{x})$  is negative semidefinite, and it is asymptotically stable if  $\dot{V}(\vec{x})$  is negative definite*.

**Example 2.1.** Consider the pendulum equation without friction, namely,

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\alpha \sin x_1 \end{cases}$$

and let us study the stability of the equilibrium point at the origin. A natural Ljapunov function is the energy function

$$H(\vec{x}) = \alpha(1 - \cos x_1) + \frac{1}{2}x_2^2$$

Clearly,  $V(\vec{0}) = 0$  and  $V(\vec{x})$  is positive definite over the domain  $-\pi < x_1 < \pi$ . The derivative of  $V(\vec{x})$  along the trajectories of the system is given by

$$\dot{V}(\vec{x}) = \alpha \dot{x}_1 \sin x_1 + x_2 \dot{x}_2 = \alpha x_2 \sin x_1 - \alpha x_2 \sin x_1 = 0$$

The condition Eq. (2.2.4) and Eq. (2.2.5) are satisfied, and we conclude that the origin is stable. Since  $\dot{V}(\vec{x}) = 0$ , we can conclude that the origin is not asymptotically stable; for trajectories starting on a Ljapunov surface  $V(\vec{x}) = c$  remain on the same surface for all future time.  $\triangleleft \triangleleft$

**Example 2.2.** Consider again the pendulum equation, but this time with friction, namely,

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\alpha \sin x_1 - \beta x_2 \end{cases}$$

Again, let us try  $V(\vec{x}) = \alpha(1 - \cos x_1) + \frac{1}{2}x_2^2$  as a Ljapunov function candidate.

$$\dot{V}(\vec{x}) = \alpha \dot{x}_1 \sin x_1 + x_2 \dot{x}_2 = -\beta x_2^2$$

The derivative  $\dot{V}(\vec{x})$  is negative semidefinite. It is not negative definite because  $\dot{V}(\vec{x}) = 0$  for  $x_2 = 0$  **irrespective of the value of  $x_1$** ; that is,  $\dot{V}(\vec{x}) = 0$  along the  $x_1$ -axis. Therefore, we can only conclude that the origin is stable. However, using the phase portrait of the harmonic damped oscillator, we have seen that when friction is counted the origin is asymptotically stable. The energy Ljapunov function fails to show this fact. Consider, now, a more general quadratic form of the term  $1/2x_2^2$ , considering the form  $1/2\vec{x}^T \mathbf{P}\vec{x}$  resulting as follows

$$\begin{aligned} V(\vec{x}) &= \frac{1}{2}\vec{x}^T \mathbf{P}\vec{x} + \alpha(1 - \cos x_1) \\ &= \frac{1}{2} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \alpha(1 - \cos x_1) \end{aligned}$$

For the quadratic form  $\frac{1}{2}\vec{x}^T \mathbf{P}\vec{x}$  to be positive definite, the element of the matrix  $\mathbf{P}$  must satisfy

$$p_{11} > 0, \quad p_{11}p_{12} - p_{12}^2 > 0$$

The derivative  $\dot{V}(\vec{x})$  is given by

$$\begin{aligned} \dot{V}(\vec{x}) &= (p_{11}x_1 + p_{12}x_2 + \alpha \sin x_1)x_2 + (p_{12}x_1 + p_{22}x_2)(-\alpha \sin x_1 - \beta x_2) \\ &= \alpha(1 - p_{22})x_2 \sin x_1 - \alpha p_{12}x_1 \sin x_1 + (p_{11} - p_{12}\beta)x_1 x_2 + (p_{12} - p_{22}\beta)x_2^2 \end{aligned}$$

Now we want to choose  $p_{11}$ ,  $p_{12}$  and  $p_{22}$  such that  $\dot{V}(\vec{x})$  is negative definite. Since the cross product terms  $x_2 \sin x_1$  and  $x_1 x_2$  are sign indefinite, we will cancel them by taking  $p_{22} = 1$  and  $p_{11} = \beta p_{12}$ . With these choices,  $p_{12}$  must satisfy  $0 < p_{12} < \beta$  for  $V(\vec{x})$  to be positive definite. Let us take  $p_{12} = \beta/2$ . Then,  $\dot{V}(\vec{x})$  is given by

$$\dot{V}(\vec{x}) = -\frac{1}{2}\alpha\beta x_1 \sin x_1 - \frac{1}{2}\beta x_2^2$$

The term  $x_1 \sin x_1 > 0$  for all  $0 < |x_1| < \pi$ . Taking  $D = \{\vec{x} \in \mathbb{R}^2 \mid |x_1| < \pi\}$ , we see that

$V(\vec{x})$  is positive definite and  $\dot{V}(\vec{x})$  is negative definite over  $D$  and we conclude that the origin is asymptotically stable.  $\triangleleft \triangleleft$

This example emphasizes an important feature of Ljapunov's stability theorem; namely, *the theorem's conditions are only sufficient*. Failure of the Ljapunov function candidate to satisfy the condition for stability or asymptotic stability does not mean that the equilibrium is not stable or asymptotically stable. It only means that such stability property cannot be established by using this Ljapunov function candidate.

In searching for a Ljapunov function in the previous example we approach the problem in a backward manner. We investigate an expression for the derivative  $\dot{V}(\vec{x})$  and went back to choose the parameters of  $V(\vec{x})$  so as to make  $\dot{V}(\vec{x})$  definite negative. This is a useful idea in searching for a Ljapunov function. A procedure that exploits this idea is known as the *variable gradient method*. To describe the procedure, let  $V(\vec{x})$  be scalar function of  $\vec{x}$  and  $g(\vec{x}) = \nabla V = (\partial V / \partial \vec{x})^T$ . The derivation  $\dot{V}(\vec{x})$  along the trajectories of Eq. (2.2.1) is given by

$$\dot{V}(\vec{x}) = \frac{\partial V}{\partial \vec{x}} f(\vec{x}) = g^T(\vec{x}) f(\vec{x})$$

The idea now is to try to choose  $g(\vec{x})$  such that it would be the gradient of a positive definite function  $V(\vec{x})$  and, at the same time,  $\dot{V}(\vec{x})$  would be negative definite. Moreover  $g(\vec{x})$  is a gradient of a scalar function ( $g(\vec{x}) = \nabla V$ ) that means the following constrain must hold

$$\frac{\partial g_i}{\partial x_j} = \frac{\partial g_j}{\partial x_i}$$

Under this constraint, we select a  $g(\vec{x})$  such that  $g^T(\vec{x})f(\vec{x})$  is negative definite. The function  $V(\vec{x})$  is then computed from the integral

$$\begin{aligned} V(\vec{x}) &= \int_{\vec{0}}^{\vec{x}} g^T(\vec{y}) d\vec{y} = \int_{\vec{0}}^{\vec{x}} \sum_{i=1}^n g_i(\vec{y}) dy_i \\ &= \int_0^{x_1} g_1(y_1, 0, \dots, 0) dy_1 + \int_0^{x_2} g_2(x_1, y_2, 0, \dots, 0) dy_2 + \dots \\ &\quad \dots + \int_0^{x_n} g_n(x_1, x_2, \dots, x_{n-1}, y_n) dy_n \end{aligned}$$

By leaving some parameters of  $g(\vec{x})$  undetermined, one would try to choose then to ensure that  $V(\vec{x})$  is positive definite.

**Example 2.3.** Consider the following scalar function

$$V(\vec{x}) = x_1^3 x_2^4 + x_1^2 x_2^2$$

and its gradient  $\vec{g}(\vec{x})$  is

$$g_1(\vec{x}) = \frac{\partial V}{\partial x_1} = 3x_1^2x_2^4 + 2x_1x_2^2$$

$$g_2(\vec{x}) = \frac{\partial V}{\partial x_2} = 4x_1^3x_2^3 + 2x_1^2x_2$$

the symmetric properties hold as follows

$$\frac{\partial g_1(\vec{x})}{\partial x_2} = 12x_2^2x_2^3 + 4x_1x_2$$

$$\frac{\partial g_2(\vec{x})}{\partial x_1} = 12x_2^2x_2^3 + 4x_1x_2$$

Integrating the gradient  $\vec{g}(\vec{x})$  along any path

$$V(\vec{x}) = \int_{\vec{0}}^{\vec{x}} g^T(\vec{y}) d\vec{y} = \int_0^{x_1} 0 dy + \int_0^{x_2} (4x_1^3y^3 + 2x_1^2y) dy = x_1^3x_2^4 + x_1^2x_2^2$$

◇

**Example 2.4.** Consider the pendulum equation, namely,

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -a \sin x_1 - bx_2 \end{cases}$$

where  $a, b \in \mathbb{R}^+$ . To apply the variable gradient method, we want to choose a second-order vector  $g(\vec{x})$  that satisfies (*symmetric property of the Jacobian matrix*)

$$\frac{\partial g_1}{\partial x_2} = \frac{\partial g_2}{\partial x_1}$$

$$\dot{V}(\vec{x}) = g_1(\vec{x})x_2 - g_2(\vec{x})[-a \sin x_1 - bx_2] < 0, \quad \text{for } \vec{x} \neq \vec{0}$$

and

$$V(\vec{x}) = \int_0^{\vec{x}} g^T(\vec{y}) d\vec{y} > 0, \quad \text{for } \vec{x} \neq \vec{0}$$

Let us try

$$g(\vec{x}) = \begin{bmatrix} \alpha(\vec{x})x_1 + \beta(\vec{x})x_2 \\ \gamma(\vec{x})x_1 + \delta(\vec{x})x_2 \end{bmatrix}$$

where the scalar function  $\alpha(\vec{x})$ ,  $\beta(\vec{x})$ ,  $\gamma(\vec{x})$  and  $\delta(\vec{x})$  are to be determined. To satisfy the symmetry of requirement, we must have

$$\beta(\vec{x}) + \frac{\partial \alpha(\vec{x})}{\partial x_2}x_1 + \frac{\partial \beta(\vec{x})}{\partial x_2}x_2 = \gamma(\vec{x}) + \frac{\partial \gamma(\vec{x})}{\partial x_1}x_1 + \frac{\partial \delta(\vec{x})}{\partial x_1}x_2 \quad (2.2.6)$$

The derivative  $\dot{V}(\vec{x})$  is given by

$$\dot{V}(\vec{x}) = \alpha(\vec{x})x_1x_2 + \beta(\vec{x})x_2^2 - a\gamma(\vec{x})x_1 \sin x_1 - a\delta(\vec{x})x_2 \sin x_1 - b\gamma(\vec{x})x_1x_2 - b\delta(\vec{x})x_2^2$$

To cancel the cross-product terms, we choose

$$\alpha(\vec{x})x_1 - b\gamma(\vec{x})x_1 - a\delta(\vec{x}) \sin x_1 = 0 \quad (2.2.7)$$

where the derivative  $\dot{V}(\vec{x})$  results in

$$\dot{V}(\vec{x}) = -[\delta(\vec{x}) - \beta(\vec{x})]x_2^2 - a\gamma(\vec{x})x_1 \sin x_1$$

Consider  $\delta(\vec{x}) = \delta = \text{constant}$ ,  $\gamma(\vec{x}) = \gamma = \text{constant}$  and  $\beta(\vec{x}) = \beta = \text{constant}$ . To satisfy Eq. (2.2.7) we obtain

$$\alpha(\vec{x})x_1 = b\gamma x_1 - a\delta \sin x_1 = 0$$

↓

$$\alpha(x_1)x_1 = b\gamma x_1 - a\delta \sin x_1 = 0$$

then,  $\alpha(\vec{x})$  depends only by  $x_1$ . To satisfy Eq. (2.2.6) we obtain

$$\beta(\vec{x}) = \gamma(\vec{x}) \quad \Rightarrow \quad \beta = \gamma \quad \left[ \frac{\partial \alpha(x_1)}{\partial x_2} = 0 \right]$$

the symmetry requirement of the Jacobian matrix is satisfied by choosing  $\beta = \gamma$ .

The expression for  $g(\vec{x})$  reduces to

$$g(\vec{x}) = \begin{bmatrix} b\gamma x_1 + a\delta \sin x_1 + \gamma x_2 \\ \gamma x_1 + \delta x_2 \end{bmatrix}$$

and

$$\alpha(x_1)x_1 = b\gamma x_1 + a\delta \sin x_1$$

$V(\vec{x})$  is obtained by integration as follows

$$\begin{aligned} V(\vec{x}) &= \int_0^{x_1} (b\gamma y_1 + a\delta \sin y_1) dy_1 + \int_0^{x_2} (\gamma x_1 + \delta y_2) dy_2 = \\ &= \frac{1}{2}a\gamma x_1^2 + a\delta(1 - \cos x_1) + \gamma x_1 x_2 + \frac{1}{2}\delta x_2^2 = \\ &= \frac{1}{2}\vec{x}^T \mathbf{P} \vec{x} + a\delta(1 - \cos x_1) \end{aligned}$$

where

$$\mathbf{P} = \begin{bmatrix} a\gamma & \gamma \\ \gamma & \delta \end{bmatrix}$$

Choosing  $\delta = 2$  and  $\gamma = 1$  ( $\beta = \gamma$ ) we obtain

$$V(\vec{x}) = \frac{1}{2}ax_1^2 + x_2^2 + x_1x_2 + a(1 - \cos x_1)$$

$$\dot{V}(\vec{x}) = -x_2^2 - ax_1 \sin x_1$$

where  $\dot{V}(\vec{x})$  is negative definite for  $\vec{x} \in [(-\pi, \pi) \times (-\pi, \pi)] - \{\vec{0}\}$ . Therefore, the origin is asymptotically stable.  $\triangleleft$

In our study of the pendulum equation with friction, we saw that the energy Ljapunov function fails to satisfy the asymptotic stability condition because  $\dot{V}(\vec{x}) = -\beta x_2^2$  is only negative semidefinite (the component  $x_1$  can assume any value). Notice, however, that  $\dot{V}(\vec{x})$  is negative every where except on the line  $x_2 = 0$ , where  $\dot{V}(\vec{x}) = 0$ . For the system to maintain the  $\dot{V}(\vec{x}) = 0$  condition, the trajectory of the system must be confined to the line  $x_2 = 0$ . Unless  $x_1 = 0$ , this is impossible because from the pendulum equation

$$x_2(t) = 0 \Rightarrow \dot{x}_2(t) = 0 \Rightarrow \sin x_1(t) = 0$$

Hence, on the segment  $-\pi < x_1 < \pi$  of the  $x_2 = 0$  line, the system can maintain the  $\dot{V}(\vec{x}) = 0$  condition only at the origin  $\vec{x} = \vec{0}$ . Therefore,  $\dot{V}(\vec{x}(t))$  must decrease toward 0 and consequently,  $\vec{x}(t) \rightarrow \vec{0}$  as  $t \rightarrow \infty$ , which is consistent with the fact that, due to friction, energy cannot remain constant while the system is in motion.

The foregoing argument shows, formally, that if in a domain about the origin we can find a Ljapunov function whose derivative along the trajectories of the system is negative semidefinite, and if we can establish that no trajectory can stay identically at point where  $\dot{V}(\vec{x}) = 0$ , except at the origin, then the origin is asymptotically stable. This idea follows from LaSalle's *Invariance Principle*.

**Theorem 2.2.** Let  $\vec{x} = \vec{0}$  be an equilibrium point for Eq. (2.2.1). Let  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  be a continuous differentiable, positive definite function such that  $\dot{V}(\vec{x}) \leq 0$  for all  $\vec{x} \in \mathbb{R}^n$ . Let  $S = \{\vec{x} \in \mathbb{R}^n \mid \dot{V}(\vec{x}) = 0\}$  and suppose that no solution can stay identically in  $S$ , other than the trivial solution  $\vec{x}(t) = \vec{0}$ . Then the origin is asymptotically stable.  $\square$

## 2.3 Ljapunov Stability.

- **First Ljapunov criterion:** the stability analysis of an equilibrium point  $\vec{x}_e$  is done studying the stability of the corresponding linearised system in the vicinity of the equilibrium point.
- **Second Ljapunov criterion:** the stability analysis of an equilibrium point  $\vec{x}_e$  is evaluated using proper **scalar** functions, called **Ljapunov functions**, defined in the state space.

The principle of Ljapunov's direct method is a mathematical extension of a fundamental physical observation: if the total energy of a dynamical system is continuously dissipated, then the system, whether linear or nonlinear, must eventually settle down to an equilibrium point.

## 2.4 First Ljapunov Criterion

### 2.4.1 Continuous-time domain

Let us consider the following *continuous-time* nonlinear system:

$$\dot{\vec{x}}(t) = \mathbf{f}(\vec{x}(t), \vec{u}(t)) \quad (2.4.1)$$

In the vicinity of the equilibrium point  $(\vec{x}_e, \vec{u}_e)$ , let us consider the corresponding linearized system:

$$\Delta \dot{\vec{x}}(t) = \tilde{\mathbf{A}} \Delta \vec{x}(t) + \tilde{\mathbf{B}} \Delta \vec{u}(t) \quad (2.4.2)$$

The following statements hold:

1. If all the eigenvalues of matrix  $\tilde{\mathbf{A}}$  have *negative real part*, then the equilibrium point  $(\vec{x}_e, \vec{u}_e)$  is asymptotically stable also for the nonlinear system.
2. If at least one of the eigenvalues of the matrix  $\tilde{\mathbf{A}}$  has *positive real part*, then the equilibrium point  $(\vec{x}_e, \vec{u}_e)$  is unstable also for the nonlinear system.
3. If at least one eigenvalue of matrix  $\tilde{\mathbf{A}}$  is located *on the imaginary axis* while all the other eigenvalues have *negative real part*, then it is not possible to conclude anything about the stability of the equilibrium point  $\vec{x}_e, \vec{u}_e$  for the nonlinear system.

### 2.4.2 Discrete-time domain

Let us consider the following *discrete-time* nonlinear system:

$$\vec{x}(k+1) = \mathbf{f}(\vec{x}(k), \vec{u}(k)) \quad (2.4.3)$$

In the vicinity of the equilibrium point  $(\vec{x}_e, \vec{u}_e)$ , let us consider the corresponding linearized system:

$$\Delta \vec{x}(k+1) = \mathbf{A} \Delta \vec{x}(k) + \mathbf{B} \Delta \vec{u}(k) \quad (2.4.4)$$

The following statements hold:

1. If all the eigenvalues of matrix  $\mathbf{A}$  have *modulus less than 1*, then the equilibrium point  $(\vec{x}_e, \vec{u}_e)$  is asymptotically stable also for the nonlinear system.
2. If at least one of the eigenvalues of the matrix  $\mathbf{A}$  has *modulus greater than 1*, then the equilibrium point  $(\vec{x}_e, \vec{u}_e)$  is unstable also for the nonlinear system.

3. If at least one eigenvalue of matrix  $\mathbf{A}$  is located *on the unitary circle* while all the other eigenvalues have *modulus less than 1*, then it is not possible to conclude anything about the stability of the equilibrium point  $\vec{x}_e, \vec{u}_e$  for the nonlinear system.

## 2.5 Second Ljapunov Criterion - Direct Method

Let us consider the following *continuous-time* nonlinear system:

$$\dot{\vec{x}}(t) = \mathbf{f}(\vec{x}(t), \vec{u}(t)) \quad (2.5.1)$$

and let  $\vec{x}_e$  be an equilibrium point corresponding to the constant input  $\vec{u}_0$ .

1. if in a neighbourhood  $W$  of  $\vec{x}_e$  exists a function  $V(\vec{x}) : W \rightarrow \mathbb{R}$ , **positive definite**, with continuous first derivatives and if  $\dot{V}(\vec{x})$  is **negative semidefinite**, then the point  $\vec{x}_e$  is stable for the nonlinear system.
2. if  $\dot{V}(\vec{x})$  is **negative definite**, then the point  $\vec{x}_e$  is asymptotically stable.

The stability of the point  $\vec{x}_e$  is guaranteed only if both the condition  $V(\vec{x}) > 0$  and  $\dot{V}(\vec{x}) < 0$  are satisfied.

## 2.6 Ljapunov Analysis of Linear Time-Invariant Systems

Consider the following linear time invariant system:

$$\dot{\vec{x}} = \tilde{\mathbf{A}}\vec{x} \quad (2.6.1)$$

We assume that  $\tilde{\mathbf{A}}$  is non-singular. The candidate Ljapunov function must be a definite positive scalar function of the vector state. A simple way to satisfy this constraint is to write it as quadratic form.

A continuous function  $V(\vec{x})$  is a **quadratic form** if it can be expressed in the following way:

$$V(\vec{x}) = \vec{x}^T \mathbf{P} \vec{x} \quad (2.6.2)$$

where  $\mathbf{P} \in \mathbb{R}$  is a **symmetric** matrix.

If the matrix  $\mathbf{P}$  is positive definite, then also the corresponding quadratic form is positive definite.

A symmetric matrix  $\mathbf{P}$  is positive definite if and only if all its principal minors are positive, that is if the following determinants are positive:

$$p_{1,1} > 0, \quad \begin{vmatrix} p_{1,1} & p_{1,2} \\ p_{2,1} & p_{2,2} \end{vmatrix} > 0, \quad \dots \quad \begin{vmatrix} p_{1,1} & \cdots & p_{1,n} \\ \vdots & & \vdots \\ p_{n,1} & \cdots & p_{n,n} \end{vmatrix} > 0 \quad (2.6.3)$$

Let  $\lambda_i$  be the eigenvalues of matrix  $\mathbf{P}_s$ , where  $\mathbf{P}_s$  is a symmetric matrix. The quadratic form:

$$V(\vec{x}) = \vec{x}^T \mathbf{P}_s \vec{x} \quad (2.6.4)$$

is positive definite if and only if all the eigenvalues  $\lambda_i$  are positive.

The quadratic forms  $V(\vec{x})$  always refer to symmetric matrices  $\mathbf{P}_s$  because:

1. a generic matrix  $\mathbf{P}$  can always be expressed as the sum of a symmetric matrix  $\mathbf{P}_s$  and a skew-symmetric matrix  $\mathbf{P}_w$ .

$$\mathbf{P} = \frac{\mathbf{P} + \mathbf{P}^T}{2} + \frac{\mathbf{P} - \mathbf{P}^T}{2} = \mathbf{P}_s + \mathbf{P}_w$$

2. only the symmetric part  $\mathbf{P}_s$  has influence on the quadratic form  $V(\vec{x})$ :

$$V(\vec{x}) = \vec{x}^T \mathbf{P} \vec{x} = \vec{x}^T \mathbf{P}_s \vec{x} + \underbrace{\vec{x}^T \mathbf{P}_w \vec{x}}_0 = \vec{x}^T \mathbf{P}_s \vec{x}$$

The skew-symmetric matrices  $\mathbf{P}_w$  satisfies the following property: the vector  $\mathbf{P}_w \vec{x}$  is always perpendicular to vector  $\vec{x}$ .

The time derivative of  $V(\vec{x})$  along any trajectory is

$$\begin{aligned} \dot{V}(\vec{x}) &= \dot{\vec{x}}^T \mathbf{P}_s \vec{x} + \vec{x}^T \mathbf{P}_s \dot{\vec{x}} = \\ &= \vec{x}^T \tilde{\mathbf{A}}^T \mathbf{P}_s \vec{x} + \vec{x}^T \mathbf{P}_s \tilde{\mathbf{A}} \vec{x} \\ &= \vec{x}^T (\tilde{\mathbf{A}}^T \mathbf{P}_s + \mathbf{P}_s \tilde{\mathbf{A}}) \vec{x} \end{aligned} \quad (2.6.5)$$

Since  $V(\vec{x})$  was chosen to be positive definite, we require, for asymptotic stability, that  $\dot{V}(\vec{x})$  be negative definite. Therefore, we require that

$$\dot{V}(\vec{x}) = -\vec{x}^T \mathbf{Q} \vec{x} \quad (2.6.6)$$

where

$$\mathbf{Q} = -(\tilde{\mathbf{A}}^T \mathbf{P}_s + \mathbf{P}_s \tilde{\mathbf{A}}) \quad (2.6.7)$$

be positive definite.

**Theorem 2.3.** Consider the system described by

$$\dot{\vec{x}} = \tilde{\mathbf{A}} \vec{x}$$

A necessary and sufficient condition for the equilibrium state  $\vec{x}_e$  to be asymptotically stable is that, given any symmetric and positive definite matrix  $\mathbf{Q}$ , there exist a symmetric and positive definite matrix  $\mathbf{P}_s$  such that

$$\mathbf{Q} = -(\tilde{\mathbf{A}}^T \mathbf{P}_s + \mathbf{P}_s \tilde{\mathbf{A}})$$

The scalar function  $V(\vec{x}) = \vec{x}^T \mathbf{P}_s \vec{x}$  is a Ljapunov function for the system.

**Example 2.5.** Determine the stability of the equilibrium state of the following system

$$\dot{x}_1 = -x_1 - 2x_2$$

$$\dot{x}_2 = x_1 - 4x_2$$

The system has only one equilibrium state at the origin. By choosing  $\mathbf{Q} = \mathbf{I}$ , we have

$$-\mathbf{I} = \tilde{\mathbf{A}}^T \mathbf{P}_s + \mathbf{P}_s \tilde{\mathbf{A}}$$

This last equation may then be written as follows

$$\begin{bmatrix} -1 & 1 \\ -2 & -4 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} + \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} -1 & -2 \\ 1 & -4 \end{bmatrix} = - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Solving for  $p$  terms we obtain a matrix  $\mathbf{P}_s$  which is positive definite. Hence we conclude that the origin of the system is asymptotically stable.  $\triangleleft$

**Example 2.6.** Let us consider the following **autonomous** nonlinear systems:

$$\begin{cases} \dot{x}_1 = \beta x_1^3 + x_2 \\ \dot{x}_2 = -x_1^3 + \beta x_2^3 \end{cases}$$

and consider the three cases ( $\beta = -1$ ,  $\beta = 1$ ,  $\beta = 0$ ), which results in the following systems:

$$\begin{cases} \dot{x}_1 = -x_1^3 + x_2 \\ \dot{x}_2 = -x_1^3 - x_2^3 \end{cases} \quad \begin{cases} \dot{x}_1 = x_1^3 + x_2 \\ \dot{x}_2 = -x_1^3 + x_2^3 \end{cases} \quad \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_1^3 \end{cases}$$

it can be easily verified that the origin  $\vec{x}_e$  is an equilibrium point for all three systems. The Jacobian matrices of the three systems have the following structure:

$$\mathbf{A}_1 = \begin{bmatrix} -3x_1^2 & 1 \\ -3x_1^2 & -3x_2^2 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 3x_1^2 & 1 \\ -3x_1^2 & 3x_2^2 \end{bmatrix}, \quad \mathbf{A}_3 = \begin{bmatrix} 0 & 1 \\ -3x_1^2 & 0 \end{bmatrix}$$

Computing the Jacobian matrices in the origin, we obtain the same matrix

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},$$

for all three systems.

This matrix has two eigenvalues in the origin and therefore the first Ljapunov criterion cannot be used.

Let us now consider the Ljapunov function  $V(\vec{x}) = x_1^4 + 2x_2^2$ . Computing the derivatives along the trajectories of the nonlinear system, one obtains:

$$\begin{aligned}\dot{u}_1(\vec{x}) &= \frac{\partial V(\vec{x})}{\partial x_1} \dot{x}_1 + \frac{\partial V(\vec{x})}{\partial x_2} \dot{x}_2 = -4x_1^6 - 4x_2^4 < 0 \\ \dot{u}_2(\vec{x}) &= \frac{\partial V(\vec{x})}{\partial x_1} \dot{x}_1 + \frac{\partial V(\vec{x})}{\partial x_2} \dot{x}_2 = 4x_1^6 + 4x_2^4 < 0 \\ \dot{u}_3(\vec{x}) &= \frac{\partial V(\vec{x})}{\partial x_1} \dot{x}_1 + \frac{\partial V(\vec{x})}{\partial x_2} \dot{x}_2 = 0\end{aligned}$$

and applying the second Ljapunov criterion it is possible to prove that in  $\vec{x} = \vec{0}$  the first system is asymptotically stable, the second is unstable, and the third is simply stable.

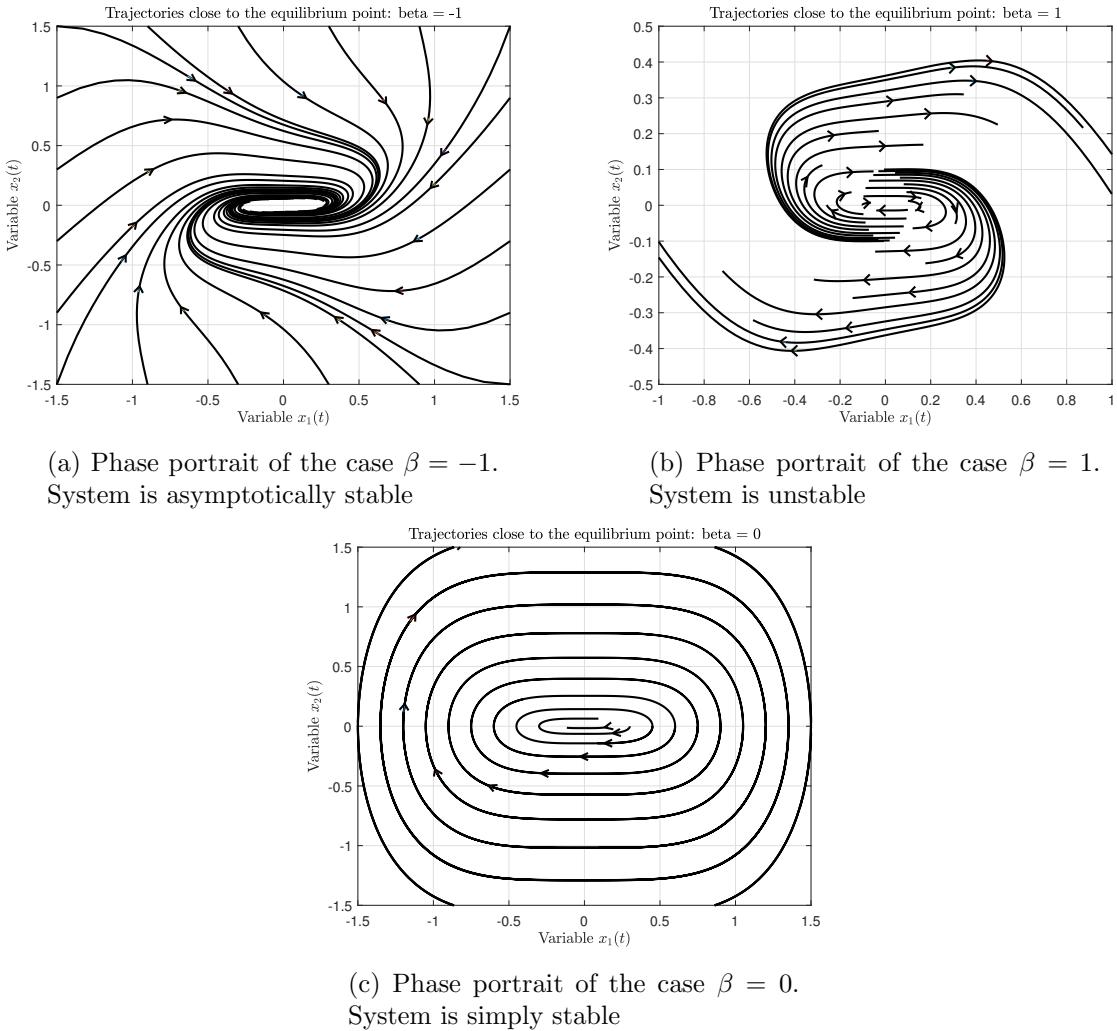


Figure 2.6: Plotting results.

△

# Chapter 3

## Optimal Control

### 3.1 Some Background

In this section we want to refresh the Lagrange multiplier method. This method is used for searching local maxima and minima of a function subject to constraints. This method is part of the **calculus of variations** and it is widely used in the optimal control problem. When it is necessary to calculate maximum and minimum subjected to constraints exist different methods as we will see also for the Kalman Filter Theory; the Lagrange multiplier is one of these methods. Let's see an example applied to curve into the euclidean space.

**Example 3.1 (Maxima and minima subject to constraint).** Find the critical points of the function

$$z = x + y \quad \text{this is a plane in the space} \quad (3.1.1)$$

subject to the constraint

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0 \quad \text{this is an ellipsoid cylinder in the space} \quad (3.1.2)$$

which means, in words, to find the minimum and maximum points of the plane which belong also to an ellipsoid cylinder. The intersection between the plane and the ellipsoid cylinder create an ellipse which lays on the plane.

For the sake of clarity we can interpret the above problem also as follows: find the critical points of the function

$$z = x + y \quad \text{where } (x, y) \in \mathcal{C} \quad (3.1.3)$$

where  $\mathcal{C}$  is the set of points which satisfy the following equation

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0 \quad \text{this is an ellipse in the plane} \quad (3.1.4)$$

To find a solution of the above problem we apply the Lagrange Multiplier Method. We start

defining the *Hamilton function*  $H(x, y, \lambda)$  as

$$\begin{aligned} H(x, y, \lambda) &= f(x, y) + \lambda g(x, y) = \\ &= x + y + \lambda \left( \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 \right) \end{aligned} \quad (3.1.5)$$

where  $\lambda$  is the Lagrange multiplier and where

$$f(x, y) = x + y \quad (3.1.6)$$

$$g(x, y) = \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0 \quad (3.1.7)$$

The Lagrange Multiplier Method says that the constrained extrema (minima or maxima) points of the function  $f(x+y)$  are the unconstrained extrema points of  $H(x, y, \lambda)$ . The unconstrained extrema points of  $H(x, y, \lambda)$  can be found imposing the equation

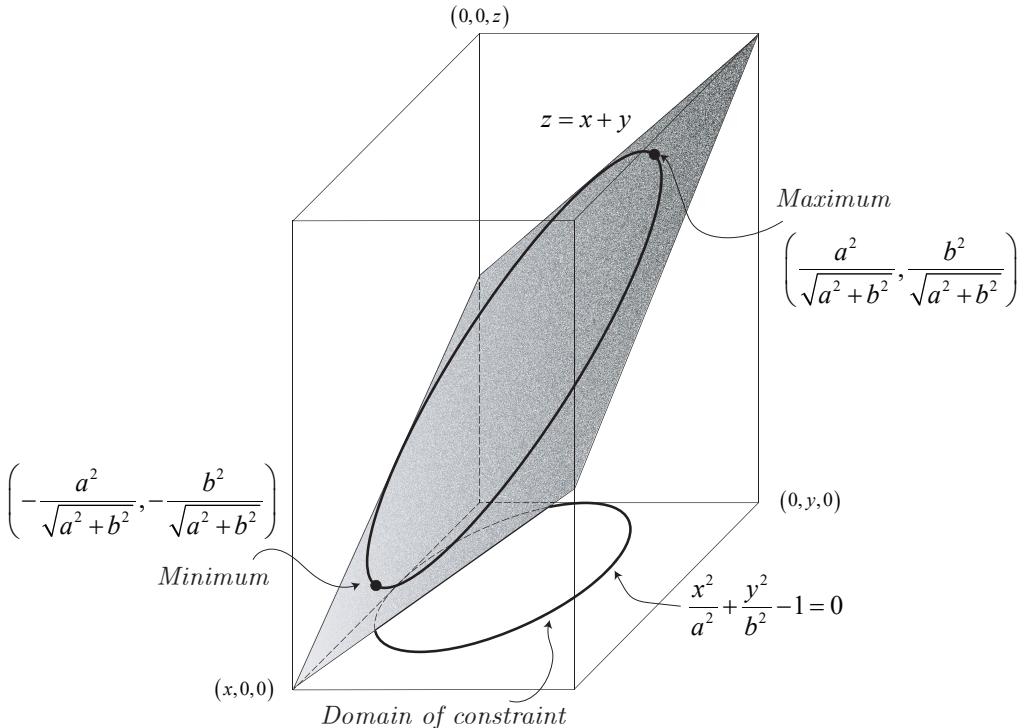


Figure 3.1: Maximum and minimum of the function  $f(x, y) = x + y$  subject to the constraint  $\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0$ .

$$\frac{d}{dt} H(x, y, \lambda) = H_x + H_y + H_\lambda + H_t = 0 \quad (3.1.8)$$

Eq. (3.1.8) is null when each term is null\*, that results in the following system of algebraic

---

\*  $H(x, y, \lambda)$  is not directly dependent of the time, i.e.  $H_t(x, y, \lambda) = 0$

equations

$$\begin{cases} \frac{\partial H}{\partial x} = 1 + 2\lambda \frac{x}{a^2} = 0 \\ \frac{\partial H}{\partial y} = 1 + 2\lambda \frac{y}{b^2} = 0 \\ \frac{\partial H}{\partial \lambda} = \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0 \end{cases} \quad (3.1.9)$$

which solved respect to  $(x, y, \lambda)$  admits the following solution

$$\left( x_1 = \frac{a^2}{\sqrt{a^2 + b^2}}, y_1 = \frac{b^2}{\sqrt{a^2 + b^2}} \right), \quad \left( x_2 = -\frac{a^2}{\sqrt{a^2 + b^2}}, y_2 = -\frac{b^2}{\sqrt{a^2 + b^2}} \right) \quad (3.1.10)$$

with

$$\lambda_{12} = \mp \frac{1}{2} \sqrt{a^2 + b^2} \quad (3.1.11)$$

evaluating the Hessian matrix and the value of the function  $N(x, y, \lambda)$  in the critical points we can identify the nature of maximum or minimum extreme<sup>†</sup> points (mean maxima or minima), see also Figure. 3.1. ◇

## 3.2 Solution of the General Continuous-time Optimization Problem

In this section we derive the solution of the continuous optimal control problem in a general case.

### 3.2.1 Problem formulation and solution

Suppose a plant as follows<sup>‡</sup>

$$\dot{x}(t) = f(x, u) \quad (3.2.1)$$

where

$$x(t) \in \mathbb{R}^n \quad (3.2.2)$$

$$u(t) \in \mathbb{R}^m. \quad (3.2.3)$$

With this system let us associate the performance index (or cost functional)

$$J(x, u) = \phi(x(t_{\text{end}})) + \int_0^{t_{\text{end}}} L(x(t), u(t)) dt \quad (3.2.4)$$

---

<sup>†</sup>Recall that a minimum problem can be always reduce to a maximum problem, in fact

$$\min(f) = -\max(-f) \quad (3.1.12)$$

<sup>‡</sup>we write  $x(t)$  instead of  $\vec{x}(t)$  by defining  $x(t) \in \mathbb{R}^n$  as well as,  $u(t)$  instead of  $\vec{u}(t)$  by defining  $u(t) \in \mathbb{R}^m$

where  $[0, t_{\text{end}}]$  is the time interval of interest. The final weighting function  $\phi(x(t_{\text{end}}))$  depends on the final state and final time, and the weighting function  $L(x(t), u(t))$  depends on the state and input at intermediate times in  $[0, t_{\text{end}}]$ .

The *optimal control problem* is to find the input  $u^*(t)$  on the time interval  $[0, t_{\text{end}}]$  that brings the plant of Eq. (3.2.2) along the trajectory  $x^*(t)$  such that the cost functional of Eq. (3.2.4) is minimized *and* such that

$$\psi(x(t_{\text{end}})) = 0 \quad (3.2.5)$$

for a given function  $\psi \in \mathbb{R}^p$ . This corresponds to the function of fixed final state.

The role of the final weighting function  $\phi(x(t_{\text{end}}))$  and of the fixed final state function  $\psi(x(t_{\text{end}})) = 0$  are mutually exclusive. The use of the final state function  $\psi(x(t_{\text{end}})) = 0$  imposes a fixed final state condition while the use of the final weighting function  $\phi(x(t_{\text{end}}))$  leaves the final state  $x(t_{\text{end}})$  free to move around the final state according to the weighting gains.

As first step we consider the following optimal control problem:

$$\left\{ \begin{array}{l} \text{find } u^*(t) \text{ such that the cost functional } J(x, u^*) \text{ is minimum, where} \\ J(x, u) = \int_0^{t_{\text{end}}} L(x(t), u(t)) dt \quad \text{under the constraints} \\ \psi(x(t_{\text{end}})) = 0 \\ \dot{x}(t) = f(x, u) \end{array} \right. \quad (3.2.6)$$

To solve the continuous optimal control problem of Eq. (3.2.6), we shall use Lagrange multiplier by interposing the multiplier  $\lambda(t) \in \mathbb{R}^n$  which is function of the time. The augmented performance index becomes

$$J'(u, x, \dot{x}, \lambda) = \int_0^{t_{\text{end}}} \left[ L(x, u) + \lambda^T (f(x, u) - \dot{x}) \right] dt \quad (3.2.7)$$

We define the Hamilton function as follows

$$H(u, x, \lambda) = L(x, u) + \lambda^T f(x, u) \quad (3.2.8)$$

Then we can rewrite Eq. (3.2.7) as follows

$$J'(u, x, \lambda) = \int_0^{t_{\text{end}}} \left[ H(x, u, \lambda) - \lambda^T \dot{x} \right] dt \quad (3.2.9)$$

The solution of the constrained optimal control problem of Eq. (3.2.6) can be computed solving

the unconstrained optimal control problem where<sup>§</sup>

$$\frac{d}{dt} J'(u, x, \lambda) = 0 \quad (3.2.10)$$

which results as follows (recall that  $H_t = 0$ )

$$\begin{aligned} \int_0^{t_{\text{end}}} \frac{d}{dt} \left[ H(u, x, \lambda) - \lambda^T \dot{x} \right] dt &= \int_0^{t_{\text{end}}} \left[ H_x^T \dot{x} + H_u^T \dot{u} + H_\lambda^T \dot{\lambda} - \dot{x}^T \dot{\lambda} - \lambda^T \ddot{x} \right] dt \\ &= \int_0^{t_{\text{end}}} \left[ H_x^T \dot{x} + H_u^T \dot{u} + (H_\lambda - \dot{x})^T \dot{\lambda} - \lambda^T \ddot{x} \right] dt \end{aligned} \quad (3.2.11)$$

To eliminate  $\ddot{x}$  we integrate by parts¶:

$$\int_0^{t_{\text{end}}} -\lambda^T \ddot{x} dt = -\lambda^T \dot{x} \Big|_{t_{\text{end}}} + \lambda^T \dot{x} \Big|_0 + \int_0^{t_{\text{end}}} \dot{\lambda}^T \dot{x} dt \quad (3.2.15)$$

---

<sup>§</sup>Leibniz integral rules for  $d/dt$  as total derivative:

$q \in \mathbb{R}^n$

$\frac{d}{dt}$  is the total derivative

$$\begin{aligned} \frac{d}{dt} f(q, t) &= \sum_i \left[ \frac{\partial f(q, t)}{\partial q_i} \frac{dq_i}{dt} \right] + \frac{\partial f(q, t)}{\partial t} = \sum_i \left[ \frac{\partial f(q, t)}{\partial t} \dot{q}_i \right] + f'(q, t) \\ \frac{d}{dt} \int_0^{t_{\text{end}}} f(q, t) dt &= \int_0^{t_{\text{end}}} \left[ \sum_i \frac{\partial f(\vec{q}, t)}{\partial q_i} \dot{q}_i \right] dt + \int_0^{t_{\text{end}}} f'(q, t) dt \\ &= \int_0^{t_{\text{end}}} \left[ \sum_i \frac{\partial f(q, t)}{\partial t} \dot{q}_i \right] dt + f(q, t) \Big|_{t=t_{\text{end}}} - f(q, t) \Big|_{t=0} \end{aligned}$$

here the case where  $f$  is time independent:

$$\begin{cases} \frac{d}{dt} \int_0^{t_{\text{end}}} f(q) dt = \int_0^{t_{\text{end}}} \left[ \sum_i \frac{\partial f(q)}{\partial q_i} \dot{q}_i \right] dt \\ \frac{\partial f(q)}{\partial t} = 0 \end{cases}$$

¶

$$d ab = \dot{a}\dot{b} + a\ddot{b} \quad (3.2.12)$$

$$ab \Big|_{t_{\text{end}}} - ab \Big|_0 = \int_0^{t_{\text{end}}} \dot{a}\dot{b} + \int_0^{t_{\text{end}}} a\ddot{b} \quad (3.2.13)$$

$$\int_0^{t_{\text{end}}} -a\ddot{b} = -ab \Big|_{t_{\text{end}}} + ab \Big|_0 + \int_0^{t_{\text{end}}} \dot{a}\dot{b} \quad (3.2.14)$$

Eq. (3.2.15) results as follows

$$\begin{aligned}
 \int_0^{t_{\text{end}}} \frac{d}{dt} \left[ H(u, x, \lambda) - \lambda^T \dot{x} \right] dt &= -\lambda^T \dot{x} \Big|_{t_{\text{end}}} + \lambda^T \dot{x} \Big|_0 + \\
 &\quad + \int_0^{t_{\text{end}}} \left[ H_x^T \dot{x} + H_u^T \dot{u} + (H_\lambda - \dot{x})^T \dot{\lambda} + \lambda^T \dot{x} \right] dt \quad (3.2.16) \\
 &= \int_0^{t_{\text{end}}} \left[ H_x^T \dot{x} + H_u^T \dot{u} + (H_\lambda - \dot{x})^T \dot{\lambda} + \lambda^T \dot{x} \right] dt
 \end{aligned}$$

where we assume the terms  $-\lambda^T \dot{x} \Big|_{t_{\text{end}}} = 0$ , due to the fact the final state is fixed, and  $\lambda^T \dot{x} \Big|_0 = 0$ .

Eq. (3.2.16) is null when

$$\begin{aligned}
 (H_x + \dot{\lambda}) &= 0 \\
 H_u &= 0 \quad (3.2.17) \\
 (H_\lambda - \dot{x}) &= f(x, u) - \dot{x} = 0
 \end{aligned}$$

which result in the following optimal control problem

$$0 = \frac{\partial H}{\partial u} = \frac{\partial L}{\partial u} + \frac{\partial f^T}{\partial u} \lambda \quad (3.2.18)$$

$$-\dot{\lambda} = \frac{\partial H}{\partial x} = \frac{\partial f^T}{\partial x} \lambda + \frac{\partial L}{\partial x}, \quad t \leq t_{\text{end}} \quad (3.2.19)$$

$$\psi(x(t_{\text{end}})) = 0 \quad (3.2.20)$$

$$\dot{x}(t) = f(x, u) \quad t \geq 0 \quad (3.2.21)$$

Let's now analyse the case where the final state is not fixed and consider the following optimal control problem

$$\begin{cases} \text{find } u^*(t) \text{ such that the cost functional } J(x, u) \text{ is minimum} \\ J(x, u) = \phi(x(t_{\text{end}})) + \int_0^{t_{\text{end}}} L(x(t), u(t)) dt \quad \text{under the constraints} \\ \dot{x}(t) = f(x, u) \end{cases} \quad (3.2.22)$$

The solution of the problem of Eq. (3.2.22) is equivalent to the solution of the problem

$$\begin{cases} \text{find } u^*(t) \text{ such that the cost functional } J'(x, u^*) \text{ is minimum} \\ J'(u, x, \lambda) = \phi(x(t_{\text{end}})) + \int_0^{t_{\text{end}}} \left[ H(x, u, \lambda) - \lambda^T(t) \dot{x}(t) \right] dt \end{cases} \quad (3.2.23)$$

where

$$H(u, x, \lambda) = L(x, u) + \lambda^T(t) f(x, u) \quad (3.2.24)$$

To find a general solution of the problem of Eq. (3.2.23) we find the conditions where the equation

$$\frac{d}{dt} J'(u, x, \lambda) = 0 \quad (3.2.25)$$

is satisfied:

$$\begin{aligned} \frac{d}{dt} J'(u, x, \lambda) &= \frac{\partial \phi(x)}{\partial x} \dot{x}(t) \Big|_{t_{\text{end}}} + \frac{d}{dt} \int_0^{t_{\text{end}}} \left[ H - \lambda^T \dot{x} \right] dt = \\ &= \frac{\partial \phi(x)}{\partial x} \dot{x}(t) \Big|_{t_{\text{end}}} + \int_0^{t_{\text{end}}} \left[ H_x^T \dot{x} + H_u^T \dot{u} + H_\lambda^T \dot{\lambda} - \dot{x}^T \dot{\lambda} - \lambda^T \ddot{x} \right] dt = \\ &= \frac{\partial \phi(x)}{\partial x} \dot{x}(t) \Big|_{t_{\text{end}}} + \int_0^{t_{\text{end}}} \left[ H_x^T \dot{x} + H_u^T \dot{u} + (H_\lambda - \dot{x})^T \dot{\lambda} - \lambda^T \ddot{x} \right] dt \end{aligned} \quad (3.2.26)$$

To eliminate  $\ddot{x}$  we integrate by parts<sup>||</sup>:

$$\int_0^{t_{\text{end}}} -\lambda^T \ddot{x} dt = -\lambda^T \dot{x} \Big|_{t_{\text{end}}} + \lambda^T \dot{x} \Big|_0 + \int_0^{t_{\text{end}}} \dot{\lambda}^T \dot{x} dt \quad (3.2.30)$$

where we assume  $\lambda^T(t) \dot{x}(t) \Big|_{t=0} = 0$ .

Then Eq. (3.2.26) becomes

$$\frac{d}{dt} J'(u, x, \lambda) = (\phi_x - \lambda)^T \dot{x} \Big|_{t_{\text{end}}} + \int_0^{t_{\text{end}}} \left[ (H_x + \dot{\lambda})^T \dot{x} + H_u^T \dot{u} + (H_\lambda - \dot{x})^T \dot{\lambda} \right] dt = 0 \quad (3.2.31)$$

||

$$D ab = \dot{a}\dot{b} + a\ddot{b} \quad (3.2.27)$$

$$ab \Big|_{t_{\text{end}}} - ab \Big|_0 = \int_0^{t_{\text{end}}} \dot{a}\dot{b} + \int_0^{t_{\text{end}}} a\ddot{b} \quad (3.2.28)$$

$$\int_0^{t_{\text{end}}} -ab = -ab \Big|_{t_{\text{end}}} + ab \Big|_0 + \int_0^{t_{\text{end}}} \dot{a}\dot{b} \quad (3.2.29)$$

$$\begin{aligned}
 & \left( \phi_x - \lambda \right) \Big|_{t_{\text{end}}} = 0 \\
 & \left( H_x + \dot{\lambda} \right) = 0 \\
 & H_u = 0 \\
 & \left( H_\lambda - \dot{x} \right) = \left( f(x, u) - \dot{x} \right) = 0 \\
 & x(0) \quad \text{given}
 \end{aligned} \tag{3.2.32}$$

which result in the following optimal control problem

$$\left[ \frac{\partial \phi(x(t))}{\partial x(t)} - \lambda(t) \right] \Big|_{t_{\text{end}}} = 0 \tag{3.2.33}$$

$$H(x, u) = L(x, u) + \lambda^T(t) f(x, u) \tag{3.2.34}$$

$$\frac{\partial H}{\partial x} = \frac{\partial f^T}{\partial x} \lambda + \frac{\partial L}{\partial x} = -\dot{\lambda}(t), \quad t \leq t_{\text{end}} \tag{3.2.35}$$

$$\frac{\partial H}{\partial u} = \frac{\partial L}{\partial u} + \frac{\partial f^T}{\partial u} \lambda = 0 \tag{3.2.36}$$

$$\dot{x} = f(x, u), \quad t \geq 0 \tag{3.2.37}$$

$$x(0) \quad \text{given} \tag{3.2.38}$$

**Example 3.2.** In this example it is desired to heat a room using the least possible energy. If  $\vartheta(t)$  is the temperature in the room,  $\vartheta_a$  the ambient air temperature outside (as constant), and  $u(t)$  the rate of heat supply (power) to the room, then the dynamic equation is

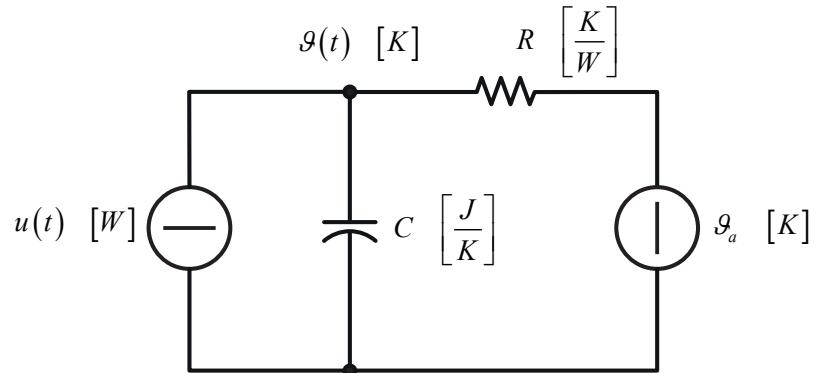


Figure 3.2: .

$$\dot{\vartheta} = -a(\vartheta - \vartheta_a) + bu \tag{3.2.39}$$

where  $a = 1/(RC)$  and  $b = 1/C$ . By defining the state as

$$x(t) = \vartheta(t) - \vartheta_a \quad (3.2.40)$$

we can write the state equation

$$\dot{x} = -ax + bu \quad (3.2.41)$$

To control the temperature on the fixed time interval  $[0 t_{\text{end}}]$  with the least possible supplied energy, we define the performance index as

$$J = \frac{1}{2} \int_0^{t_{\text{end}}} u^2(t) dt \quad (3.2.42)$$

The Hamiltonian is

$$H = \frac{u^2}{2} + \lambda(-ax + bu) \quad (3.2.43)$$

The optimal control problem is determined by solving the following system of equations

$$\dot{x} = H_\lambda = -ax + bu \quad (3.2.44)$$

$$\dot{\lambda} = -H_x = a\lambda \quad (3.2.45)$$

$$0 = H_u = u + b\lambda \quad (3.2.46)$$

From Eq. (3.2.46) the optimal control is given by

$$u(t) = -b\lambda(t) \quad (3.2.47)$$

so to determine  $u^*(t)$  we need only find the optimal costate  $\lambda^*(t)$ .

Substituting Eq. (3.2.47) into Eq. (3.2.44) yields the state-costate equations

$$\dot{x} = -ax - b^2\lambda \quad (3.2.48)$$

$$\dot{\lambda} = a\lambda \quad (3.2.49)$$

which must now be solved for  $\lambda^*(t)$  and the optimal state trajectory  $x^*(t)$ . We do not yet know the final costate  $\lambda(t_{\text{end}})$ , but let us solve Eq. (3.2.48) and Eq. (3.2.49) as if we did. The solution of Eq. (3.2.49) is

$$\lambda(t) = \exp \left[ -a(t_{\text{end}} - t) \right] \lambda(t_{\text{end}}) \quad (3.2.50)$$

using this solution into Eq. (3.2.48) yields

$$\dot{x} = -ax - b^2\lambda(t_{\text{end}}) \exp \left[ -a(t_{\text{end}} - t) \right] \quad (3.2.51)$$

which results as follows

$$x(t) = x(0) \exp(-at) - \frac{b^2}{a} \lambda(t_{\text{end}}) \exp(-at_{\text{end}}) \sinh(at) \quad (3.2.52)$$

Eq. (3.2.49) and Eq. (3.2.52) give the optimal costate  $\lambda^*(t)$  and state  $x^*(t)$  in terms of the as yet unknown final costate  $\lambda(t_{\text{end}})$ . The initial state  $x(0)$  is given.

Now we consider two possible control objectives, which will give two ways to determine  $\lambda(t_{\text{end}})$ .

#### Fixed final state.

Suppose that the initial temperature of the room is equal to  $\vartheta_a = 60^\circ\text{C}$ . Then

$$x(0) = 0^\circ\text{C} \quad (3.2.53)$$

Let our control objective be to drive the final temperature  $\vartheta(t_{\text{end}})$  exactly to  $70^\circ\text{C}$  at the given final time  $t_{\text{end}}$ . Then the final state is required to take on the fixed value of

$$x(t_{\text{end}}) = 10^\circ\text{C} \quad (3.2.54)$$

Using Eq. (3.2.53) and Eq. (3.2.54), we must determine  $\lambda(t_{\text{end}})$ , then we can find  $\lambda(t)$  by using Eq. (3.2.49) and optimal control by using Eq. (3.2.47). To find  $\lambda(t_{\text{end}})$ , use Eq. (3.2.52) to write

$$x(t_{\text{end}}) = x(0) \exp(-at_{\text{end}}) - \frac{b^2}{2a} \lambda(t_{\text{end}}) \left( 1 - \exp(-2at_{\text{end}}) \right) \quad (3.2.55)$$

Taking into account Eq. (3.2.53) and Eq. (3.2.54) the final costate is

$$\lambda(t_{\text{end}}) = \frac{20a}{b^2 [1 - \exp(-2at_{\text{end}})]} \quad (3.2.56)$$

and so the optimal costate trajectory is

$$\lambda^*(t) = -\frac{10a \exp(at)}{b^2 \sinh(at_{\text{end}})} \quad (3.2.57)$$

Finally, the optimal rate of heat supply to the room is

$$u^*(t) = \frac{10a \exp(at)}{b \sinh(at_{\text{end}})} \quad 0 \leq t \leq t_{\text{end}} \quad (3.2.58)$$

applying Eq. (3.2.58) to Eq. (3.2.41) and solving the equation we obtain the following optimal trajectory

$$x^*(t) = 10 \frac{\sinh(at)}{\sinh(at_{\text{end}})}. \quad (3.2.59)$$

#### Free final state.

Now suppose that we are not so concerned that the final state  $x(t_{\text{end}})$  be exactly  $10^\circ\text{C}$ . Let

us demand only that the control  $u(t)$  minimize

$$J(x, u) = \frac{1}{2}s[x(t_{\text{end}}) - 10]^2 + \frac{1}{2} \int_0^{t_{\text{end}}} u^2(t) dt \quad (3.2.60)$$

where  $s \in \mathbb{R}^+$  is the weighting factor. If  $s$  is large then the optimal solution will have  $x(t_{\text{end}})$  close to  $10^\circ\text{C}$ . Since  $x(t_{\text{end}})$  is not fixed we have to apply Eq. 3.2.33, such that

$$\lambda(t_{\text{end}}) = \left. \frac{\partial \phi}{\partial x} \right|_{t_{\text{end}}} = s[x(t_{\text{end}}) - 10] \quad (3.2.61)$$

Note that there is no function  $\psi(x(t_{\text{end}}))$  in this problem. From Eq. (3.2.53), Eq. (3.2.55) and Eq. (3.2.61) we determinate  $\lambda(t_{\text{end}})$  as follows

$$x(t_{\text{end}}) = \frac{\lambda(t_{\text{end}})}{s} + 10 \quad (3.2.62)$$

and

$$\frac{\lambda(t_{\text{end}})}{s} + 10 = x(0) \exp(-at_{\text{end}}) - \frac{b^2}{2a} \lambda(t_{\text{end}}) \left(1 - \exp(-2at_{\text{end}})\right) \quad (3.2.63)$$

which results as follows

$$\lambda(t_{\text{end}}) = \frac{-20as}{2a + b^2s(1 - \exp(-2at_{\text{end}}))} \quad (3.2.64)$$

The optimal costate trajectory

$$\lambda^*(t) = \frac{-10as \exp(at)}{a \exp(at_{\text{end}}) + b^2s \sinh(at_{\text{end}})} \quad (3.2.65)$$

and

$$u^*(t) = \frac{10abs \exp(at)}{a \exp(at_{\text{end}}) + b^2s \sinh(at_{\text{end}})} \quad (3.2.66)$$

solving Eq. (3.2.66) with the optimal control input we obtain the following optimal state trajectory

$$x^*(t) = \frac{10b^2s \sinh(at)}{a \exp(at_{\text{end}}) + b^2s \sinh(at_{\text{end}})} \quad (3.2.67)$$

The final state  $x^*(t_{\text{end}})$  in Eq. (3.2.67) is not equal to the desired temperature  $10^\circ\text{C}$ , but is function of the weighting factor  $s$  of the performance index.

△

### 3.3 Introduction to Linear Quadratic Optimal Control

Let us consider the control system defined by the state equation

$$\vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) \quad (3.3.1)$$

where

$\vec{x}(k)$  = state vector ( $n$ -vector)

$\vec{u}(k)$  = control vector ( $r$ -vector)

$\mathbf{A} = n \times n$  matrix

$\mathbf{B} = n \times r$  matrix

In the quadratic optimal control problem we desire to determine a law for the input vector  $\vec{u}(k)$  such that a given quadratic performance index is minimized. An example of a **quadratic performance index** is

$$J = \frac{1}{2} \vec{x}^T(N) \mathbf{S} \vec{x}(N) + \frac{1}{2} \sum_{k=0}^{N-1} \left[ \vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}^T(k) \mathbf{R} \vec{u}(k) \right] \quad (3.3.2)$$

where matrices  $\mathbf{S}$  and  $\mathbf{Q}$  are symmetrical\*\* and positive definite or positive semidefinite†† matrices and  $\mathbf{R}$  is symmetrical and positive definite matrix. The first term on the right-hand side of Eq. (3.3.2) accounts for the importance of the final state. The first term in the summation brackets accounts for the relative importance of the error during the control process, and the second term accounts for the expenditure of the energy of the control signals.

The optimal control law is given by

$$\vec{u}(k) = -\mathbf{K}(k) \vec{x}(k) \quad (3.3.3)$$

where  $\mathbf{K}(k)$  is a time varying  $r \times n$  matrix. If  $N = \infty$ , then  $\mathbf{K}(k)$  becomes a constant  $r \times n$  matrix. The design of optimal control systems based on such quadratic performance indexes consist of the determination of the matrix  $\mathbf{K}(k)$ .

The initial state of the system is at some arbitrary state  $\vec{x}(0) = \vec{c}$ . If the final state  $\vec{x}(N)$  is not fixed, it can be included in the cost function in order to introduce a contribution, or better a weight, in the performance measure due to the final state. Note that in the minimization problem, the inclusion of the term  $\frac{1}{2} \vec{x}^T(N) \mathbf{S} \vec{x}(N)$  in the performance index  $J$  implies that we desire the final state  $\vec{x}(N)$  to be as close to the origin as possible.

---

\*\*  $\mathbf{A}$  is symmetric, if and only if  $\mathbf{A} = \mathbf{A}^T$  or  $a_{ij} = a_{ji} \forall i, j$ .

†† An  $n \times n$  symmetric real matrix  $\mathbf{A}$  is said to be positive definite if  $\vec{x}^T \mathbf{A} \vec{x} > 0 \forall \vec{x} \in \mathbb{R}^n - \{\vec{0}\}$ . An  $n \times n$  symmetric real matrix  $\mathbf{A}$  is said to be positive semi-definite if  $\vec{x}^T \mathbf{A} \vec{x} \geq 0 \forall \vec{x} \in \mathbb{R}^n$

The quadratic optimal control problem is a minimization problem with constraints as follows

$$\left\{ \begin{array}{l} \text{find } u(k) \text{ such that } J(\vec{x}, \vec{u}) \text{ is minimum, where} \\ J(\vec{x}, \vec{u}) = \frac{1}{2} \vec{x}^T(N) \mathbf{S} \vec{x}(N) + \frac{1}{2} \sum_{k=0}^{N-1} [\vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}^T(k) \mathbf{R} \vec{u}(k)] \\ \text{subject to the constraint: } \vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) \end{array} \right. \quad (3.3.4)$$

where  $k = 0, 1, 2, \dots, N - 1$ , and where the initial condition on the state vector is specified as

$$\vec{x}(0) = \vec{c} \quad (3.3.5)$$

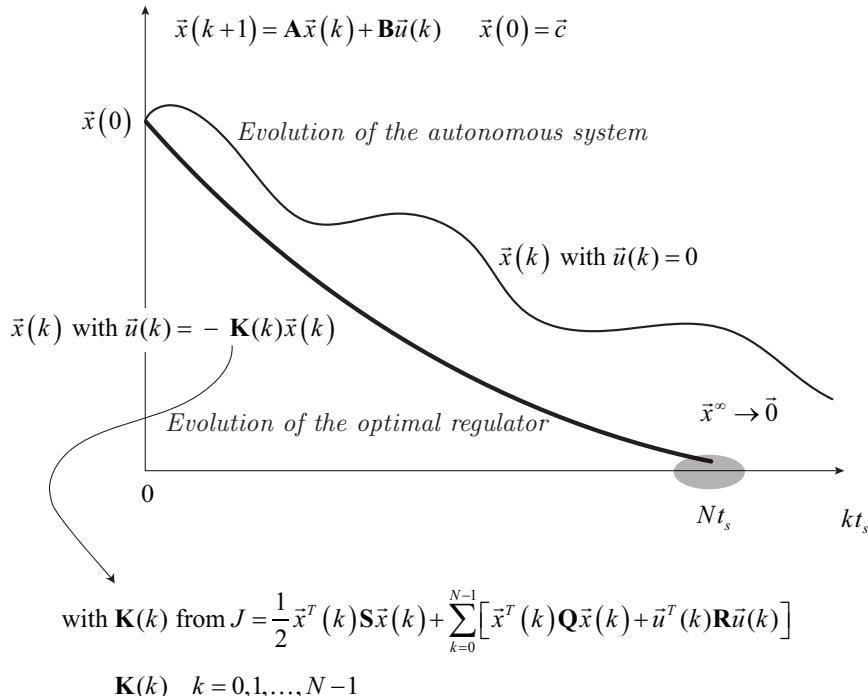


Figure 3.3: Informal visualization of the evolution of a state vector in case of autonomous system (dashed) and in the case of optimal state feedback (solid).

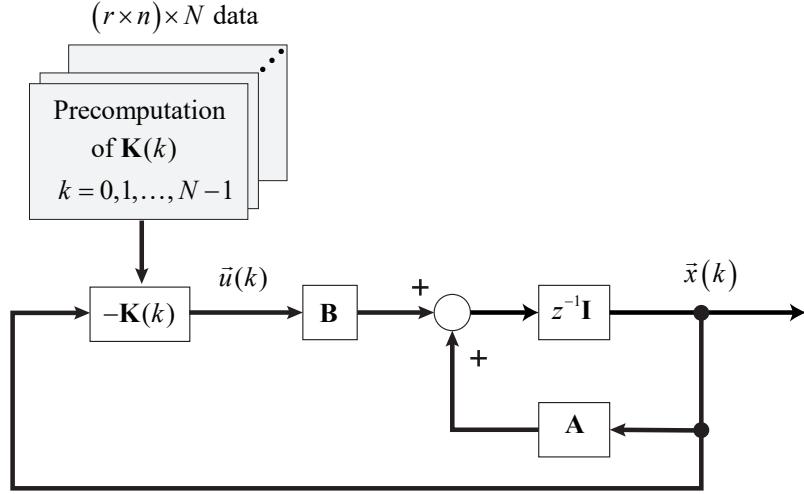


Figure 3.4: Implementation of the finite time optimal regulator.

Now, by using a set of Lagrange multiplier  $\vec{\lambda}(1), \vec{\lambda}(2), \dots, \vec{\lambda}(N)$  we define a new performance index  $J'$  as follows:

$$J'(\vec{x}, \vec{u}, \vec{\lambda}) = \frac{1}{2} \vec{x}^T(N) \mathbf{S} \vec{x}(N) + \frac{1}{2} \sum_{k=0}^{N-1} \left\{ [\vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}^T(k) \mathbf{R} \vec{u}(k)] \right. \\ \left. + 2\vec{\lambda}(k+1)^T [\mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) - \vec{x}(k+1)] \right\} \quad (3.3.6)$$

or also

$$J'(\vec{x}, \vec{u}, \vec{\lambda}) = \frac{1}{2} \vec{x}^T(N) \mathbf{S} \vec{x}(N) + \frac{1}{2} \sum_{k=0}^{N-1} \left\{ [\vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}^T(k) \mathbf{R} \vec{u}(k)] \right. \\ \left. + \vec{\lambda}(k+1)^T [\mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) - \vec{x}(k+1)] \right. \\ \left. + [\mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) - \vec{x}(k+1)]^T \vec{\lambda}(k+1) \right\} \quad (3.3.7)$$

It is well known fact that minimization of the function  $J'$  defined by Eq. (3.3.7) is equivalent to minimization of  $J$  as defined by Eq. (3.3.2) when it is subject to the equality constraint defined by Eq. (3.3.1).

In order to minimize the functional  $J'$ , we need to differentiate<sup>††</sup>  $J'$  with respect to time and equals the result to zero. This operation is equivalent to derivate the functional  $J'$  for each component of vectors  $\vec{x}(N)$ ,  $\vec{x}(k)$ ,  $\vec{u}(k)$  and  $\vec{\lambda}(k)$  and equals each term to zero, as follows

<sup>††</sup>via total differential

Remembering that, if  $\mathbf{Q}$  is real and symmetric  $n \times n$  matrix, it holds

$$\frac{\partial}{\partial \vec{x}} \vec{x}^T \mathbf{Q} \vec{x} = 2\mathbf{Q} \vec{x}$$

$$\frac{\partial}{\partial \vec{y}} \vec{x}^T \mathbf{Q} \vec{y} = \mathbf{Q}^T \vec{x} = \mathbf{Q} \vec{x} \quad \mathbf{Q} \text{ is real symmetric}$$

$$\frac{\partial}{\partial \vec{y}} \vec{x}^T \mathbf{A} \vec{y} = \mathbf{A}^T \vec{x} \quad \mathbf{A} \text{ is real matrix}$$

$$\frac{\partial}{\partial \vec{x}} \vec{x}^T \mathbf{Q} \vec{y} = \mathbf{Q} \vec{y}$$

and

$$\vec{x} = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T$$

$$\frac{\partial}{\partial \vec{x}} f = \begin{bmatrix} \frac{\partial}{\partial x_1} f & \cdots & \frac{\partial}{\partial x_n} f \end{bmatrix}$$

$$\frac{\partial}{\partial \vec{x}^T} f = \begin{bmatrix} \frac{\partial}{\partial x_1} f \\ \vdots \\ \frac{\partial}{\partial x_n} f \end{bmatrix}$$

and

$$\frac{\partial}{\partial \vec{x}} \vec{x}^T \mathbf{Q} = \frac{\partial}{\partial \vec{x}} \mathbf{Q}^T \vec{x} = \mathbf{Q}^T$$

we obtain

$$\frac{\partial J'}{\partial \vec{x}(k)} = \vec{0} \quad \Rightarrow \quad \mathbf{Q} \vec{x}(k) + \mathbf{A}^T \vec{\lambda}(k+1) - \vec{\lambda}(k) = \vec{0} \quad k = 1, 2, \dots, N-1 \quad (3.3.8)$$

$$\frac{\partial J'}{\partial \vec{x}(N)} = \vec{0} \quad \Rightarrow \quad \mathbf{S} \vec{x}(N) - \vec{\lambda}(N) = \vec{0} \quad (3.3.9)$$

$$\frac{\partial J'}{\partial \vec{u}(k)} = \vec{0} \quad \Rightarrow \quad \mathbf{R} \vec{u}(k) + \mathbf{B}^T \vec{\lambda}(k+1) = \vec{0} \quad k = 0, 1, \dots, N-1 \quad (3.3.10)$$

$$\frac{\partial J'}{\partial \vec{\lambda}(k)} = \vec{0} \quad \Rightarrow \quad \mathbf{A} \vec{x}(k-1) + \mathbf{B} \vec{u}(k-1) - \vec{x}(k) = \vec{0} \quad k = 1, 2, \dots, N \quad (3.3.11)$$

From Eq. (3.3.8) we have

$$\vec{\lambda}(k) = \mathbf{Q} \vec{x}(k) + \mathbf{A}^T \vec{\lambda}(k+1) \quad k = 1, 2, \dots, N-1 \quad (3.3.12)$$

with final condition  $\lambda(N) = \mathbf{S}\vec{x}(N)$ . From Eq. (3.3.10) we obtain

$$\vec{u}(k) = -\mathbf{R}^{-1}\mathbf{B}^T\vec{\lambda}(k+1) \quad k = 0, 1, \dots, N-1 \quad (3.3.13)$$

and

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \quad k = 0, 1, \dots, N-1 \quad (3.3.14)$$

Substituting Eq. (3.3.13) into Eq. (3.3.14) we obtain

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\vec{\lambda}(k+1) \quad k = 0, 1, \dots, N-1 \quad (3.3.15)$$

with initial state  $\vec{x}(0) = \vec{c}$ . In order to obtain the solution to the minimization problem we need to solve the Eq. (3.3.12) and Eq. (3.3.15) resulting in the following boundary value problem

$$\left\{ \begin{array}{l} \vec{\lambda}(k) = \mathbf{Q}\vec{x}(k) + \mathbf{A}^T\vec{\lambda}(k+1) \\ \vec{\lambda}(N) = \mathbf{S}\vec{x}(N) \\ \vec{x}(k+1) = \mathbf{A}\vec{x}(k) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\vec{\lambda}(k+1) \\ \vec{x}(0) \text{ given} \end{array} \right. \quad (3.3.16)$$

Now we want to find a way to express the control vector  $\vec{u}(k)$  in the following form

$$\vec{u}(k) = -\mathbf{K}(k)\vec{x}(k) \quad (3.3.17)$$

where  $\mathbf{K}(k)$  is the  $r \times n$  feedback matrix. For this purpose the Riccati transformation can be a possible solution.

**Riccati transformation** - It can be assumed that  $\vec{\lambda}(k)$  has the following form:

$$\vec{\lambda}(k) = \mathbf{P}(k)\vec{x}(k) \quad (3.3.18)$$

where  $\mathbf{P}(k)$  is an  $n \times n$  real symmetric matrix. Substitution of Eq. (3.3.18) into Eq. (3.3.12) results in

$$\mathbf{P}(k)\vec{x}(k) = \mathbf{Q}\vec{x}(k) + \mathbf{A}^T\mathbf{P}(k+1)\vec{x}(k+1) \quad (3.3.19)$$

and substitution of Eq. (3.3.18) into Eq. (3.3.15) results in

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)\vec{x}(k+1) \quad (3.3.20)$$

which can be written as follows

$$\vec{x}(k+1) = [\mathbf{I} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)]^{-1} \mathbf{A} \vec{x}(k) \quad (3.3.21)$$

substituting Eq. (3.3.21) into Eq. (3.3.19) we obtain

$$\mathbf{P}(k)\vec{x}(k) = \mathbf{Q}\vec{x}(k) + \mathbf{A}^T\mathbf{P}(k+1)[\mathbf{I} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)]^{-1}\mathbf{A}\vec{x}(k)$$

or

$$\mathbf{P}(k) = \mathbf{Q} + \mathbf{A}^T\mathbf{P}(k+1)[\mathbf{I} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)]^{-1}\mathbf{A} \quad (3.3.22)$$

The term  $[\mathbf{I} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)]^{-1}$  can be written as follows<sup>§§</sup>

$$[\mathbf{I} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)]^{-1} = \mathbf{I} - \mathbf{B}[\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1)$$

Hence Eq. (3.3.22) can be modified as follows

$$\begin{cases} \mathbf{P}(k) = \mathbf{Q} + \mathbf{A}^T\mathbf{P}(k+1)\mathbf{A} + \\ \quad - \mathbf{A}^T\mathbf{P}(k+1)\mathbf{B}[\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A} \\ \mathbf{P}(N) = \mathbf{S} \end{cases} \quad (3.3.25)$$

Eq. (3.3.25) is called the Riccati equation. The system Eq. (3.3.25) can be solved uniquely backward from  $k = N$  to  $k = 0$  obtaining  $\mathbf{P}(N), \mathbf{P}(N-1), \dots, \mathbf{P}(0)$  starting from  $\mathbf{P}(N)$  which is known.

**The optimal control problem is inherently a backward-in-time problem.**

The optimal control vector  $\vec{u}(k)$  can be represented as follows, from Eq. (3.3.13), Eq. (3.3.21),

---

<sup>§§</sup>**Sherman-Morrison Formula:** if  $\mathbf{A}_{n \times n}$  is nonsingular and if  $\mathbf{c}$  and  $\mathbf{d}$  are  $n \times 1$  columns such  $1 + \mathbf{d}^T\mathbf{A}^{-1}\mathbf{c} \neq 0$ , then the sum  $\mathbf{A} + \mathbf{c}\mathbf{d}^T$  is nonsingular, and

$$(\mathbf{A} + \mathbf{c}\mathbf{d}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{c}\mathbf{d}^T\mathbf{A}^{-1}}{1 + \mathbf{d}^T\mathbf{A}^{-1}\mathbf{c}} \quad (3.3.23)$$

The Sherman-Morrison-Woodbury formula is a generalization. If  $\mathbf{C}$  and  $\mathbf{D}$  are  $n \times k$  such that  $(\mathbf{I} + \mathbf{D}^T\mathbf{A}^{-1}\mathbf{C})^{-1}$  exists, then

$$(\mathbf{A} + \mathbf{C}\mathbf{D}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{C}(\mathbf{I} + \mathbf{D}^T\mathbf{A}^{-1}\mathbf{C})^{-1}\mathbf{D}^T\mathbf{A}^{-1} \quad (3.3.24)$$

and rules¶

$$\begin{aligned}
 \vec{u}(k) &= -\mathbf{R}^{-1}\mathbf{B}^T\vec{\lambda}(k+1) \\
 &= -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)\vec{x}(k+1) \\
 &= -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)[\mathbf{I} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)]^{-1}\mathbf{A}\vec{x}(k) \\
 &= -\mathbf{R}^{-1}\mathbf{B}^T[\mathbf{P}^{-1}(k+1) + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T]^{-1}\mathbf{A}\vec{x}(k) \\
 &= -\mathbf{K}(k)\vec{x}(k)
 \end{aligned} \tag{3.3.28}$$

where the equation

$$\mathbf{K}(k) = \mathbf{R}^{-1}\mathbf{B}^T[\mathbf{P}^{-1}(k+1) + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T]^{-1}\mathbf{A} \tag{3.3.29}$$

can be manipulated as follow\*\*\*

$$\begin{aligned}
 \mathbf{R}^{-1}\mathbf{B}^T[\mathbf{P}^{-1}(k+1) + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T]^{-1} &= \\
 &= \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1) - \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}[\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1) = \\
 &= [\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1)
 \end{aligned} \tag{3.3.30}$$

which results as follows

$$\mathbf{K}(k) = [\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1) \tag{3.3.31}$$

From Eq. (3.3.31)

**The final representation become**

$$\left\{
 \begin{array}{l}
 \vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \quad \text{with} \quad \vec{x}(0) = \vec{c} \\
 \vec{u}(k) = -\mathbf{K}(k)\vec{x}(k) \\
 \mathbf{K}(k) = [\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A} \\
 \mathbf{P}(k) = \mathbf{Q} + \mathbf{A}^T\mathbf{P}(k+1)[\mathbf{A} - \mathbf{B}\mathbf{K}(k)] \quad \text{with} \quad \mathbf{P}(N) = \mathbf{S}
 \end{array}
 \right. \tag{3.3.32}$$

¶ rules:

$$\mathbf{A}^{-1}(\mathbf{BC})^{-1} = (\mathbf{BCA})^{-1} \quad \text{or} \quad \mathbf{A}(\mathbf{BC})^{-1} = (\mathbf{BCA}^{-1})^{-1} \tag{3.3.26}$$

and

$$(\mathbf{BC})^{-1}\mathbf{A}^{-1} = (\mathbf{ABC})^{-1} \quad \text{or} \quad (\mathbf{BC})^{-1}\mathbf{A} = (\mathbf{A}^{-1}\mathbf{BC})^{-1} \tag{3.3.27}$$

$$***(\mathbf{A} + \mathbf{BD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{I} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1}$$

### 3.4 Steady State Quadratic Optimal Control

Considering the control system defined by Eq. (3.3.1):

$$\vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) \quad (3.4.1)$$

We have seen that when the control process is finite time, i.e. when  $N$  is finite, the feedback gain matrix  $\mathbf{K}(k)$  becomes a time varying matrix. Let us now consider the quadratic optimal control problem where the process continues without bound, or when  $N = \infty$ . As  $N$  approach to infinity, the optimal control solution becomes a steady state solution and the time varying gain matrix  $\mathbf{K}(k)$  becomes a constant gain matrix. Such a constant gain matrix  $\mathbf{K}(k)$  is called a steady state gain matrix and is written as  $\mathbf{K}$ .

For  $N = \infty$ , the performance index may be modified to

$$J_{ss} = \frac{1}{2} \sum_{k=0}^{\infty} [\vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}^T(k) \mathbf{R} \vec{u}(k)] \quad (3.4.2)$$

The term  $\frac{1}{2}\vec{x}^T(N)\mathbf{S}\vec{x}(N)$  is not here included because, if the optimal control is stable, the value of  $J$  converges to a constant and the term  $\frac{1}{2}\vec{x}_{\text{inf}}^T \mathbf{S} \vec{x}_{\text{inf}} = 0$  because  $\vec{x}_{\text{inf}} = 0$ .

In many practical systems, instead of using a time-varying gain matrix  $\mathbf{K}(k)$  we approximate such a matrix by the constant gain matrix  $\mathbf{K}$ .

Implementing the steady state (or time invariant) optimal controller the steady state solution of the Riccati equation must be found.

From Eq. (3.3.25)

$$\begin{cases} \mathbf{P}(k) = \mathbf{Q} + \mathbf{A}^T \mathbf{P}(k+1) \mathbf{A} + \\ \quad - \mathbf{A}^T \mathbf{P}(k+1) \mathbf{B} [\mathbf{R} + \mathbf{B}^T \mathbf{P}(k+1) \mathbf{B}]^{-1} \mathbf{B}^T \mathbf{P}(k+1) \mathbf{A} \\ \mathbf{P}(N) = \mathbf{S} \end{cases} \quad (3.4.3)$$

in steady state condition we obtain the condition

$$\mathbf{P}(k+1) = \mathbf{P}(k) = \mathbf{P} \quad (3.4.4)$$

Substituting Eq. (3.4.4) into Eq. (3.4.3) we obtain an algebraic equation in  $\mathbf{P}$  as unknown variable

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{A}^T \mathbf{P} \mathbf{B} [\mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B}]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \quad (3.4.5)$$

Once matrix  $\mathbf{P}$  has been computed, the optimal state feedback matrix  $\mathbf{K}$  is obtained as follows

$$\mathbf{K} = [\mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B}]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \quad (3.4.6)$$

Combining Eq. (3.4.5) and Eq. (3.4.6) we obtain the algebraic equation system

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{A}^T \mathbf{P} \mathbf{B} \left[ \mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B} \right]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \quad (3.4.7)$$

and the steady state feedback matrix gain is given as follows

$$\mathbf{K} = \left[ \mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B} \right]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \quad (3.4.8)$$

or we can also write the algebraic equation system as follows

$$\begin{cases} \mathbf{K} = \left[ \mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B} \right]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \\ \mathbf{P} = \mathbf{Q} + \mathbf{A}^T \mathbf{P} (\mathbf{A} - \mathbf{B} \mathbf{K}) \end{cases} \quad (3.4.9)$$

### 3.4.1 Steady state quadratic optimal control using matlab

Solution of the system of Eq. (3.4.9) can be computed using the matlab commands

$$\begin{aligned} [\mathbf{Klqr}] &= \text{lqr}(\mathbf{Atilde}, \mathbf{Btilde}, \mathbf{Q}, \mathbf{R}) \\ [\mathbf{Klqr}] &= \text{dlqr}(\mathbf{A}, \mathbf{B}, \mathbf{Q}, \mathbf{R}) \end{aligned} \quad (3.4.10)$$

first command is used for the continuous time model, while the second command is used for the discrete time model.

**Example 3.3.** Steady state Riccati equation for the double integrator

**Problem description:** Find a solution for the steady state Riccati equation for the linear quadratic optimal problem considering  $\mathbf{A} = \begin{bmatrix} 1 & t_s \\ 0 & 1 \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} 0 \\ t_s \end{bmatrix}$  and matrices  $\mathbf{Q}$  and  $\mathbf{R}$  as for your design.

**Solution.** The double integrator system

$$\ddot{x}(t) = u(t) \quad (3.4.11)$$

can be written in the form

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}} \vec{x}(t) + \tilde{\mathbf{B}} u(t) \quad (3.4.12)$$

where  $\tilde{\mathbf{A}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ ,  $\tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ .

Checking the controllability of the system we can see that the controllable matrix

$$\mathcal{C} = \begin{bmatrix} \tilde{\mathbf{B}} & \tilde{\mathbf{A}} \tilde{\mathbf{B}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (3.4.13)$$

has full rank, that means, the system is fully controllable.

Discretizing Eq. (3.4.12) we obtain the system

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}u(k) \quad (3.4.14)$$

where  $\mathbf{A} = \begin{bmatrix} 1 & t_s \\ 0 & 1 \end{bmatrix}$  and  $\mathbf{B} = \begin{bmatrix} 0 \\ t_s \end{bmatrix}$ .

Let's now consider the steady-state Riccati equation

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{A}^T \mathbf{P} \mathbf{B} \left[ \mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B} \right]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \quad (3.4.15)$$

where  $\mathbf{A} = \begin{bmatrix} 1 & t_s \\ 0 & 1 \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} 0 \\ t_s \end{bmatrix}$ ,  $\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ , and  $\mathbf{R} = 1$ .

Eq. (3.4.15) can be expanded as follows

$$\begin{bmatrix} x & z \\ z & y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ t_s & 1 \end{bmatrix} \begin{bmatrix} x & z \\ z & y \end{bmatrix} \begin{bmatrix} 1 & t_s \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ t_s & 1 \end{bmatrix} \begin{bmatrix} x & z \\ z & y \end{bmatrix} \begin{bmatrix} 0 \\ t_s \end{bmatrix} \left( 1 + \begin{bmatrix} 0 & t_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x & z \\ z & y \end{bmatrix} \begin{bmatrix} 0 \\ t_s \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 & t_s \\ t_s & 1 \end{bmatrix} \begin{bmatrix} x & z \\ z & y \end{bmatrix} \begin{bmatrix} 1 & t_s \\ 0 & 1 \end{bmatrix} \quad (3.4.16)$$

where  $\mathbf{P} = \begin{bmatrix} x & z \\ z & y \end{bmatrix}$  means it is symmetric (matrix  $\mathbf{P}$  is symmetric by definition).

Hence, with some manipulation we obtain the system equation

$$\begin{bmatrix} x & z \\ z & y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} x & t_s x + z \\ t_s x + z & t_s(t_s x + z) + t_s z + y \end{bmatrix} + \begin{bmatrix} t_s z \\ t_s(t_s z + y) \end{bmatrix} \left( 1 + t_s^2 y \right)^{-1} \begin{bmatrix} t_s z & t_s^2 z + t_s y \\ t_s^2 z(t_s z + y) & t_s^2(t_s z + y)^2 \end{bmatrix} \quad (3.4.17)$$

and

$$\begin{bmatrix} x & z \\ z & y \end{bmatrix} = \begin{bmatrix} 1 + x & t_s x + z \\ t_s x + z & 1 + t_s(t_s x + z) + t_s z + y \end{bmatrix} + \begin{bmatrix} t_s^2 z^2 & t_s^3 z^2 + t_s^2 y z \\ t_s^2 z(t_s z + y) & t_s^2(t_s z + y)^2 \end{bmatrix} - \left( 1 + t_s^2 y \right)^{-1} \quad (3.4.18)$$

and

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & t_s x \\ t_s x & 1 + t_s(t_s x + z) + t_s z \end{bmatrix} + \begin{bmatrix} t_s^2 z^2 & t_s^2 z(t_s z + y) \\ t_s^2 z(t_s z + y) & t_s^2(t_s z + y)^2 \end{bmatrix} - \left( 1 + t_s^2 y \right)^{-1} \quad (3.4.19)$$

Eq. (3.4.19) results in the following system of algebraic equations

$$\begin{cases} 1 - (1 + t_s^2 y)^{-1} t_s^2 z^2 = 0 \\ t_s x - (1 + t_s^2 y)^{-1} t_s^2 z(t_s z + y) = 0 \\ 1 + t_s(t_s x + z) + t_s z - (1 + t_s^2 y)^{-1} t_s^2 (t_s z + y)^2 = 0 \end{cases} \quad (3.4.20)$$

The system of Eq. (3.4.20) is made by three second degree equations in  $x$ ,  $y$  and  $z$ , and this seems to be correct due to the fact the Riccati equation is made by a summary of quadratic monomials.

Solution of the system of Eq. (3.4.20) shall be find imposing the matrix  $\mathbf{P}$  is definite positive (to impose the full controllability).

Supposing  $t_s = 10\text{ ms}$ , we can find two real solution as follows:

$$\begin{cases} x_1 \approx 1733.05 \\ y_1 \approx 1733.55 \\ z_1 \approx 1000.87 \end{cases} \quad \begin{cases} x_2 \approx -1731.05 \\ y_2 \approx -1730.55 \\ z_2 \approx 999.134 \end{cases} \quad (3.4.21)$$

Solution  $x_2$ ,  $y_2$ ,  $z_2$  will be rejected because brings matrix  $\mathbf{P}$  definite negative (or, due to the fact is symmetric, contains negative eigenvalues).

The same result can be achieved using the Matlab command `idare(A,B,Q,R)`.

$\trianglelefteq$

## 3.5 Optimal Control of the Mass-Spring-Mass System

Supposing the Mass-Spring-Mass system as follows

**In continuous time domain.** We consider  $\tau_2(t) = 0$

$$\begin{cases} J_1 \dot{\omega}_1(t) + b\omega_1 + b_\vartheta [\omega_1(t) - \omega_2(t)] = \tau_1(t) - \tau_\vartheta(t) \\ J_2 \dot{\omega}_2(t) + b\omega_2 + b_\vartheta [\omega_2(t) - \omega_1(t)] = \tau_\vartheta(t) \\ \dot{\tau}_\vartheta(t) = k_\vartheta [\omega_1(t) - \omega_2(t)] \end{cases}$$

which its state space representation is as follows

$$\begin{bmatrix} \dot{\omega}_1(t) \\ \dot{\omega}_2(t) \\ \dot{\tau}_\vartheta(t) \end{bmatrix} = \begin{bmatrix} -\frac{b+b_\vartheta}{J_1} & \frac{b_\vartheta}{J_1} & -\frac{1}{J_1} \\ \frac{b_\vartheta}{J_2} & -\frac{b+b_\vartheta}{J_2} & \frac{1}{J_2} \\ k_\vartheta & -k_\vartheta & 0 \end{bmatrix} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \\ \tau_\vartheta(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix} \tau_1(t)$$

where

$$\vec{x}(t) = \begin{bmatrix} \omega_1(t) & \omega_2(t) & \tau_\vartheta(t) \end{bmatrix}^T$$

$$\tilde{\mathbf{A}} = \begin{bmatrix} -\frac{b+b_\vartheta}{J_1} & \frac{b_\vartheta}{J_1} & -\frac{1}{J_1} \\ \frac{b_\vartheta}{J_2} & -\frac{b+b_\vartheta}{J_2} & \frac{1}{J_2} \\ k_\vartheta & -k_\vartheta & 0 \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}$$

we want to evaluate (in discrete time domain) the behavior of three different system configurations:

- autonomous system  $\vec{x}(k+1) = \mathbf{A} \vec{x}(k)$
- optimal state feedback  $\vec{x}(k+1) = [\mathbf{A} - \mathbf{B}\mathbf{K}(k)] \vec{x}(k)$
- steady state optimal state feedback  $\vec{x}(k+1) = [\mathbf{A} - \mathbf{B}\mathbf{K}] \vec{x}(k)$

with initial condition  $\vec{x}(0) = \vec{c}$ . The state feedback matrix gain is determined considering the following cost function

$$J = \frac{1}{2} \vec{x}^T(N) \mathbf{S} \vec{x}(N) + \frac{1}{2} \sum_{k=0}^{N-1} [\vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}(k)^T \mathbf{R} \vec{u}(k)]$$

and for the steady state case

$$J_{ss} = \frac{1}{2} \sum_{k=0}^{N-1} [\vec{x}^T(k) \mathbf{Q} \vec{x}(k) + \vec{u}(k)^T \mathbf{R} \vec{u}(k)]$$

hence we are going to consider three different systems.

### Optimal state feedback with finite time process.

$$\begin{cases} \vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) & \text{with } \vec{x}(0) = [0 \ 0 \ -1] \\ \vec{u}(k) = -\mathbf{K}(k) \vec{x}(k) & \text{without constraints} \\ \mathbf{K}(k) = [\mathbf{R} + \mathbf{B}^T \mathbf{P}(k+1) \mathbf{B}]^{-1} \mathbf{B}^T \mathbf{P}(k+1) \mathbf{A} \\ \mathbf{P}(k) = \mathbf{Q} + \mathbf{A}^T \mathbf{P}(k+1) [\mathbf{A} - \mathbf{B}\mathbf{K}(k)] & \text{with } \mathbf{P}(N) = \mathbf{S} \end{cases} \quad (3.5.1)$$

**Steady state optimal state feedback system.**

$$\begin{cases} \vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \quad \text{with} \quad \vec{x}(0) = [0 \ 0 \ -1] \\ \vec{u}(k) = -\mathbf{K}\vec{x}(k) \quad \text{without constraints} \\ \mathbf{K} = [\mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B}]^{-1} \mathbf{B}^T \mathbf{P} \mathbf{A} \\ \mathbf{P} = \mathbf{Q} + \mathbf{A}^T \mathbf{P} [\mathbf{A} - \mathbf{B} \mathbf{K}] \end{cases} \quad (3.5.2)$$

**Autonomous system.**

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) \quad \text{with} \quad \vec{x}(0) = [0 \ 0 \ -1] \quad (3.5.3)$$

The above systems can be graphically represented as follows

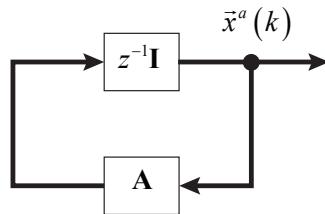


Figure 3.5: System representation for the case of autonomous system. The symbol  $\vec{x}^a(k)$  for the autonomous system is used.

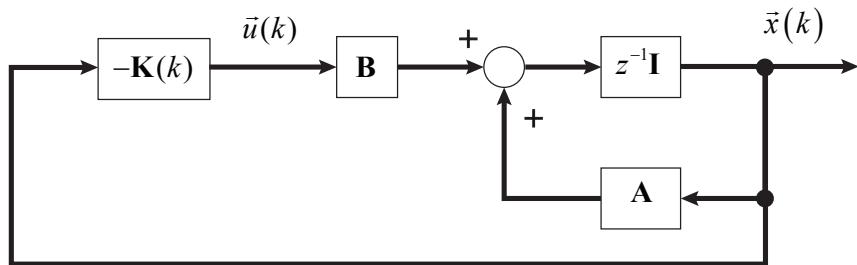


Figure 3.6: System representation for the case with optimal state feedback where matrix  $\mathbf{K}(k)$ ,  $k = 0, 1, \dots, N - 1$  has been pre-calculated.

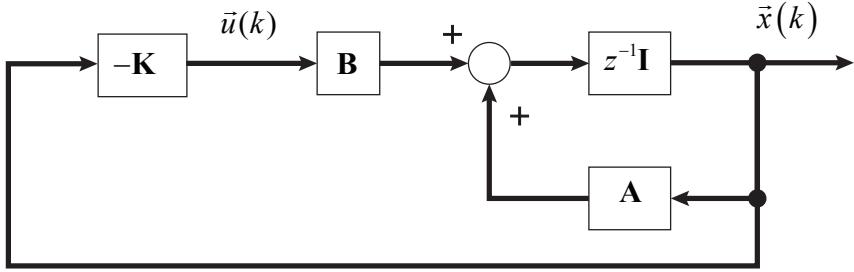


Figure 3.7: System representation for the case with steady state optimal state feedback.

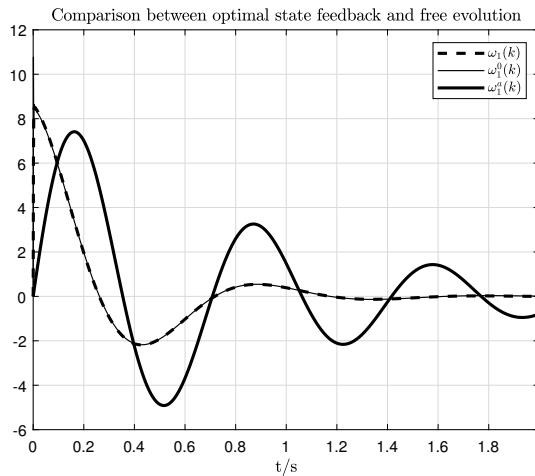
### 3.5.1 Simulation Results

In the following we can evaluate the state vector  $\vec{x}(k)$  evolution in both three cases.

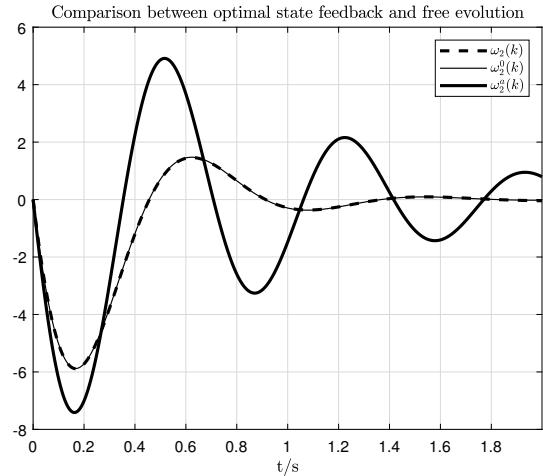
We assume:  $J_1 = J_2 = 0.025 \text{ kg m}^2$ ,  $b_\theta = 0.02 \frac{\text{Nm}}{\text{rad/s}}$ ,  $k_\theta = 1 \frac{\text{Nm}}{\text{rad}}$  and  $t_s = 1 \text{ ms}$ .

Regarding optimal state feedback we assume:  $N = 2500$ ,  $\mathbf{R} = 1$ ,  $\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$  and  $\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

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



(a) Evolution of the first element of the state vector  $\vec{x}(k)$  which is  $\omega_1$ . Where  $\omega_1^a$  represent the evolution of the autonomous system and  $\omega_1^0$  the evolution of the steady state optimal state-feedback.

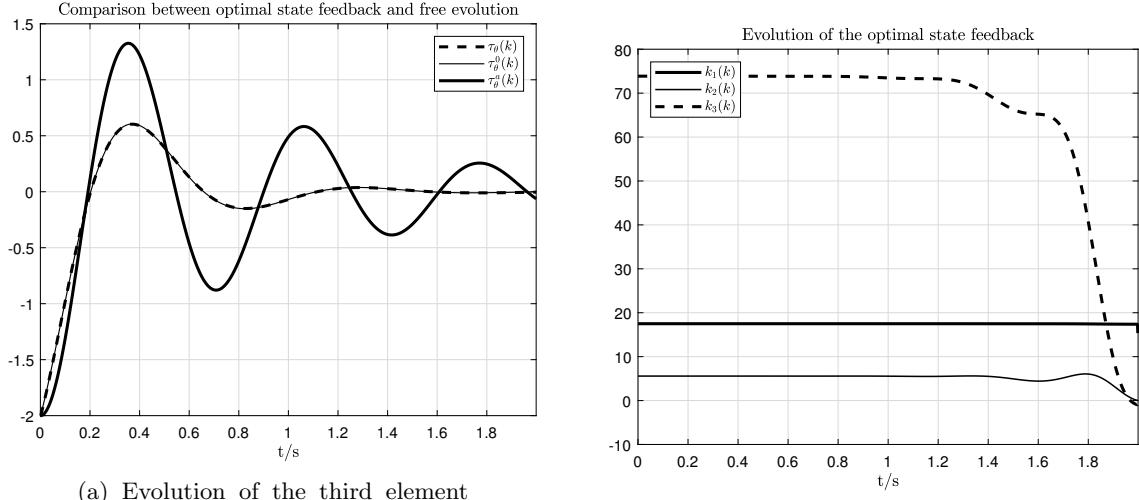


(b) Evolution of the second element of the state vector  $\vec{x}(k)$  which is  $\omega_2$ . Where  $\omega_2^a$  represent the evolution of the autonomous system and  $\omega_2^0$  the evolution of the steady state optimal state-feedback.

Figure 3.8: Simulation results.

### 3.5. Optimal Control of the Mass-Spring-Mass System

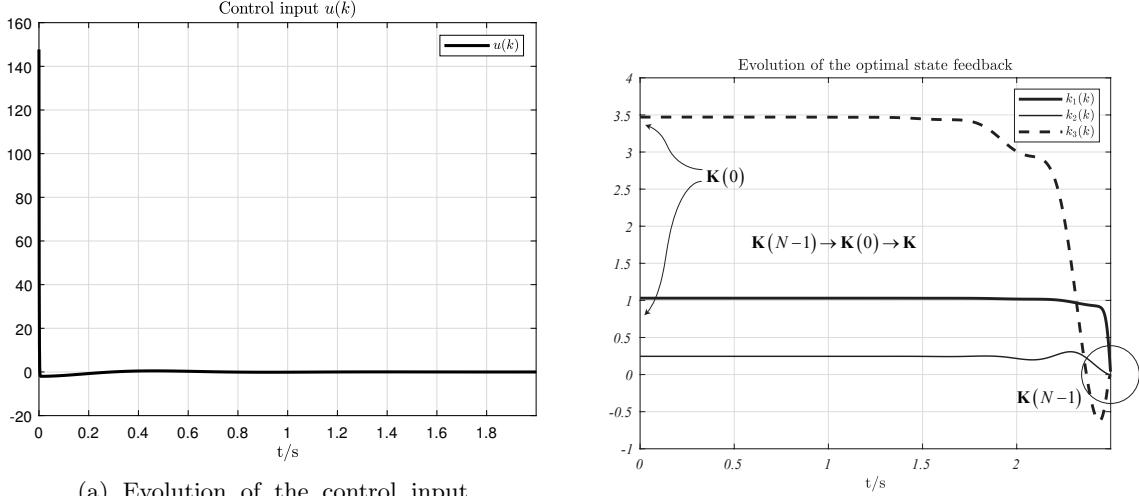
---



(a) Evolution of the third element of the state vector  $\vec{x}(k)$  which is  $\tau_\vartheta$ . Where  $\tau_\vartheta^a$  represent the evolution of the autonomous system and  $\tau_\vartheta^0$  the evolution of the steady state optimal state-feedback.

(b) Pre-calculation of the state feedback gains matrix  $\mathbf{K}(k)$ . We can note that its evolution start at time  $(N-1)t_s$  and ends to time 0 where reach a steady state condition.

Figure 3.9: Simulation results.



(a) Evolution of the control input  $u(k)$  for the case of state feedback. For the autonomous, obviously,  $u(k) = 0$

(b) The calculation of the matrix  $\mathbf{K}(k)$  start from the point  $(N-1)t_s$  and ends to the origin.

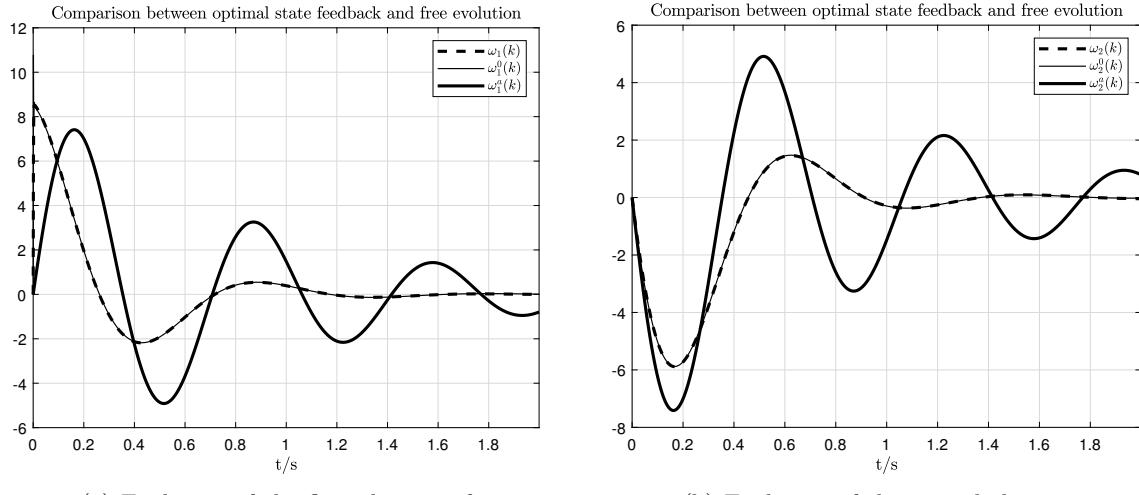
Figure 3.10: Simulation results.

From Fig. 3.9b we can see the evolution of the pre-calculation of the matrix  $\mathbf{K}(k)$  and we can note its behaviour which becomes constant in the case we consider an sufficient value of  $N$  steps.

For sake of clarity, we consider now a new set of the matrix gains as follows:

$$N = 2500, \quad \mathbf{R} = 0.001, \quad \mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.5.4)$$

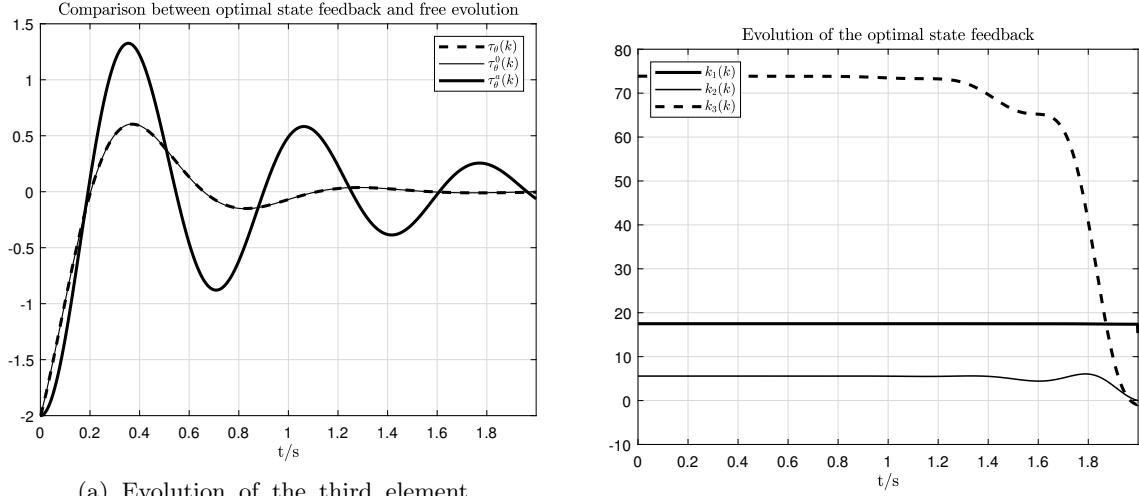
We reduce the value of  $\mathbf{R}$  that means we leave the control to be more aggressive and we aspect an increase of the control input  $u(k)$  as can be seen in the plotting results and we remove the constraint regarding the component  $\omega_1$  keeping the optimization active to  $\omega_2$  and  $\tau_\theta$ .



(a) Evolution of the first element of the state vector  $\vec{x}(k)$  which is  $\omega_1$ . Where  $\omega_1^a$  represent the evolution of the autonomous system and  $\omega_1^0$  the evolution of the steady state optimal state-feedback.

(b) Evolution of the second element of the state vector  $\vec{x}(k)$  which is  $\omega_2$ . Where  $\omega_2^a$  represent the evolution of the autonomous system and  $\omega_2^0$  the evolution of the steady state optimal state-feedback.

Figure 3.11: Simulation results.



(a) Evolution of the third element of the state vector  $\vec{x}(k)$  which is  $\tau_\vartheta$ . Where  $\tau_\vartheta^a$  represent the evolution of the autonomous system and  $\tau_\vartheta^0$  the evolution of the steady state optimal state-feedback.

(b) Pre-calculation of the state feedback gains matrix  $\mathbf{K}(k)$ . We can note that its evolution starts at time  $(N - 1)t_s$  and ends to time 0 where reach a steady state condition.

Figure 3.12: Simulation results.

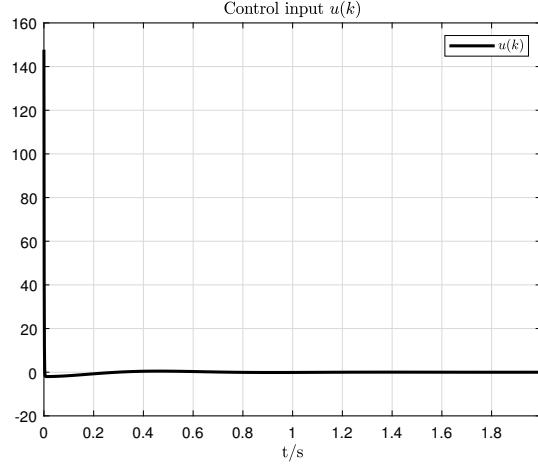


Figure 3.13: Evolution of the control input  $u(k)$  for the case of state feedback. For the autonomous, obviously,  $u(k) = 0$

### 3.6 Linear Quadratic Integral (LQI) Control

Consider the control structure of Fig. 3.14, where

$$\left\{ \begin{array}{l} \vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) \\ \vec{y}(k) = \mathbf{C} \vec{x}(k) \\ \vec{v}(k) = \vec{v}(k-1) + \vec{r}(k) - \vec{y}(k) \\ \vec{u}(k) = - [\mathbf{K}_I \quad \mathbf{K}_x] [\vec{v}(k) \quad \vec{x}(k)]^T \end{array} \right. \quad (3.6.1)$$

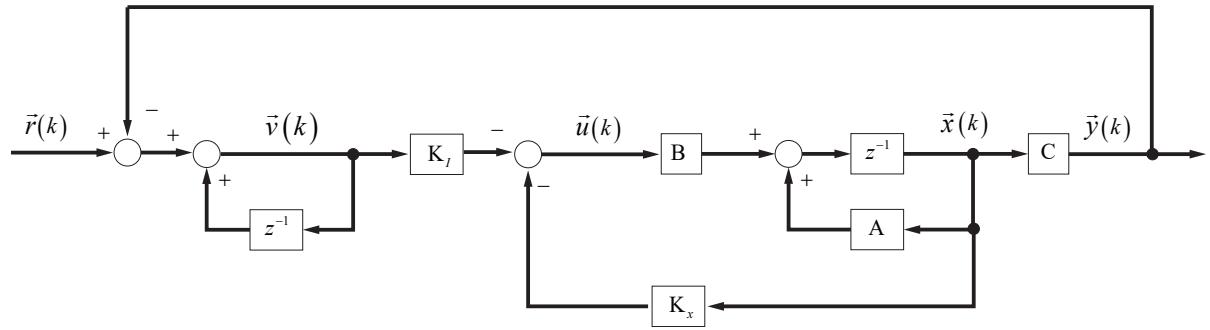


Figure 3.14: LQI controller structure.

Define the new state vector

$$\vec{z}(k) = \begin{bmatrix} \vec{v}(k) \\ \vec{x}(k) \end{bmatrix} \quad (3.6.2)$$

Eq. (3.6.1) becomes

$$\vec{z}(k+1) = \begin{bmatrix} \mathbf{I}_m & -\mathbf{C}\mathbf{A} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \vec{z}(k) + \begin{bmatrix} -\mathbf{C}\mathbf{B} \\ \mathbf{B} \end{bmatrix} \vec{u}(k) + \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(k) \quad (3.6.3)$$

which can be rewritten as

$$\vec{z}(k+1) = \mathbf{A}' \vec{z}(k) + \mathbf{B}' \vec{u}(k) + \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(k) \quad (3.6.4)$$

The optimal control problem can be solved minimizing the following performance index

$$\begin{aligned}
 J = & \frac{1}{2} \left( \vec{z}(N) - \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(N) \right)^T \mathbf{S}' \left( \vec{z}(N) - \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(N) \right) + \\
 & + \frac{1}{2} \sum_{k=0}^{N-1} \left[ \left( \vec{z}(k) - \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(k) \right)^T \mathbf{Q}' \left( \vec{z}(k) - \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(k) \right) + \right. \\
 & \left. + \vec{u}^T(k) \mathbf{R} \vec{u}(k) \right]
 \end{aligned} \tag{3.6.5}$$

Applying the solution of the Riccati equation we obtain

$$\begin{cases} \vec{z}(k+1) = \mathbf{A}' \vec{z}(k) + \mathbf{B}' \vec{u}(k) + \begin{bmatrix} \mathbf{I}_m \\ \mathbf{0} \end{bmatrix} \vec{r}(k) \\ \vec{u}(k) = -\mathbf{K}(k) \vec{z}(k) \\ \mathbf{K}(k) = [\mathbf{K}_I \quad \mathbf{K}_x] = [\mathbf{R} + \mathbf{B}'^T \mathbf{P}'(k+1) \mathbf{B}']^{-1} \mathbf{B}'^T \mathbf{P}'(k+1) \mathbf{A}' \\ \mathbf{P}'(k) = \mathbf{Q}' + \mathbf{A}'^T \mathbf{P}'(k+1) (\mathbf{A}' - \mathbf{B}' \mathbf{K}(k)) \quad \text{with} \quad \mathbf{P}'(N) = \mathbf{S}' \end{cases} \tag{3.6.6}$$

### 3.6.1 LQI Control of Mass-Spring-Mass

In the following the simulation results of the LQI control applied to the MSM system is shown.

We assume:

$$J_1 = J_2 = 0.025 \text{ kg m}^2, \quad b_\vartheta = 0.02 \text{ N m rad}^{-1} \text{ s}, \quad k_\vartheta = 1 \text{ N m rad},$$

$$t_s = 1 \text{ ms}, \quad N = 5000, \quad \mathbf{R} = 1,$$

$$\mathbf{S}' = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{Q}' = \begin{bmatrix} t_s & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The simulation are performed with the steady state value of  $\mathbf{K}_x(k) \rightarrow \mathbf{K}_x = [1.79 \quad 9.1 \quad 15.5]$  and  $\mathbf{K}_I(k) \rightarrow \mathbf{K}_I = -0.03$ .

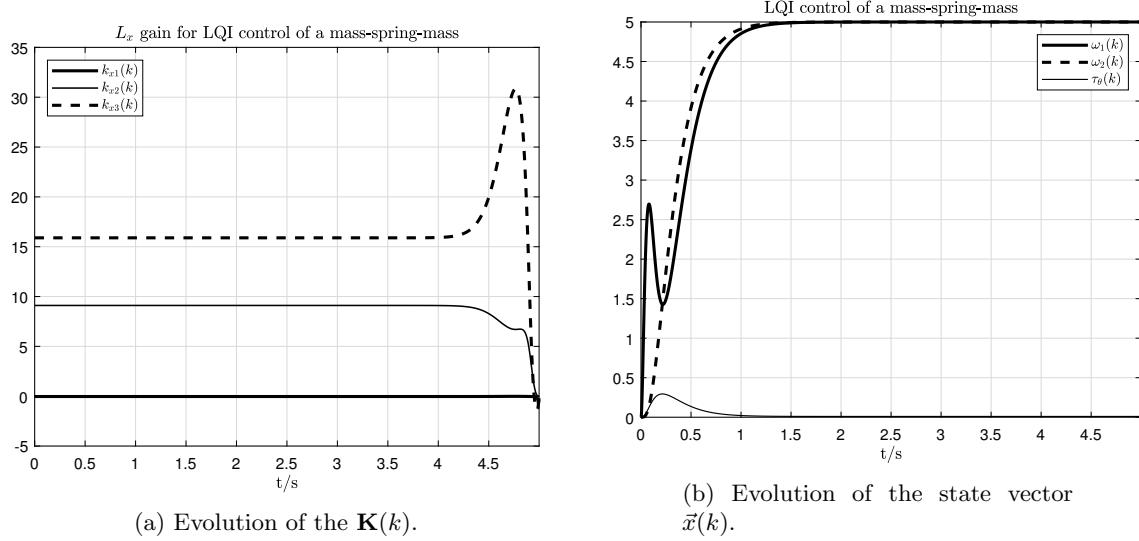


Figure 3.15: Simulation results.

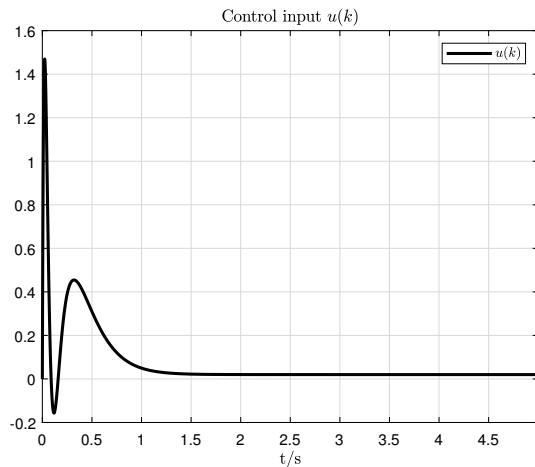


Figure 3.16: Control input  $u(k)$ .

## Chapter 4

# Least Squares Parameters Estimation for Dynamical Systems

In this paragraph a different approach, on system identification is investigated, in particular the least squares method is used in the case of parameter estimation of discrete-time Dynamical systems.

### 4.1 Non-Recursive Method of Least Squares (LS)

In the following, the classical method of least square for discrete-time linear process will be derived. Let  $G(z^{-1})$  be the transfer function of the discrete-time linear process:

$$\begin{aligned} G(z^{-1}) &= \frac{y_u(z)}{u(z)} = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} = \\ &= \frac{b_1 z^{-1} + \cdots + b_m z^{-m}}{a_1 z^{-1} + \cdots + a_m z^{-m}} z^{-d} \end{aligned} \quad (4.1.1)$$

Scope of the method is to determine the unknown parameters  $a_1, \dots, a_m, b_1, \dots, b_m$ . Figure 4.1

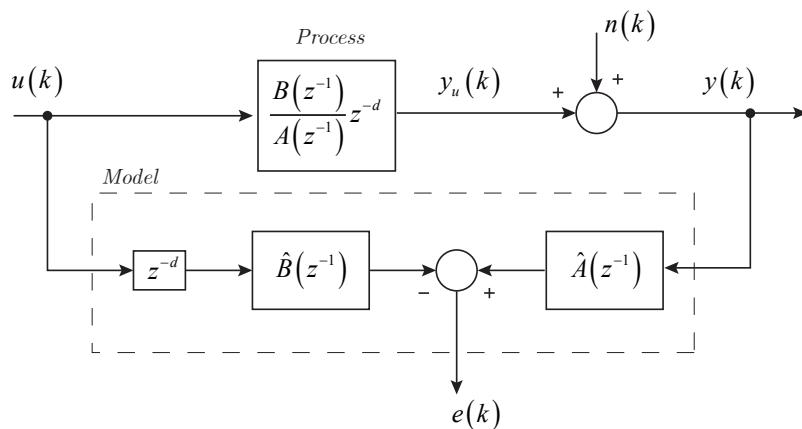


Figure 4.1: Non-recursive scheme for least squares method.

shows the structure of the least squares method, where the measurable signal  $y(k)$  is assumed to be a superposition of the real process output  $y_u(k)$  and the stochastic signal  $n(k)$ .

$$y(k) = y_u(k) + n(k) \quad (4.1.2)$$

Now the fundamental task is to define a way to estimate the parameters of the linear process by the only measure of the input  $u(k)$  and output  $y(k) = y_u(k) + n(k)$  of the system. Task of the algorithm is to find a way to converge our estimation quantities to the real one applied on the system.

$$\begin{aligned} \hat{\theta} &= (\hat{a}_1 \dots \hat{a}_m \hat{b}_1 \dots \hat{b}_m) \rightarrow \theta_0 = (a_{10} \dots a_{m0} b_{10} \dots b_{m0}) \\ \hat{y}_u(k) &= f(\hat{\theta}) \rightarrow y_u(k) = f(\theta_0) \end{aligned}$$

The real process equation is given by:

$$\begin{aligned} y_u(k) + a_{10}(k-1)y_u(k-1) + \dots + a_{m0}(k-m)y_u(k-m) + \\ - b_{10}(k-1)u(k-1-d) - \dots - b_{m0}(k-m)u(k-m-d) = 0 \end{aligned} \quad (4.1.3)$$

Now we use the measured quantity  $y(k)$  instead of  $y_u(k)$  and we introduce the estimating model parameters

$$\boxed{\frac{\hat{B}(z^{-1})}{\hat{A}(z^{-1})} z^{-d}}$$

$$\begin{aligned} y(k) + \hat{a}_1(k-1)y_u(k-1) + \dots + \hat{a}_m(k-m)y_u(k-m) + \\ - \hat{b}_1(k-1)u(k-1-d) - \dots - \hat{b}_m(k-m)u(k-m-d) = e(k) \end{aligned} \quad (4.1.4)$$

**Where  $e(k)$  represents the generalized equation error.**

The Eq. (4.1.4) can be interpreted as the prediction  $\hat{y}(k|k-1)$  of the output signal  $y(k)$  for one step ahead noised on the measurements available at time step  $k-1$ .

$$\begin{aligned} \hat{y}(k|k-1) &= \hat{y}(k) = -\hat{a}_1(k-1)y(k-1) - \dots - \hat{a}_m(k-m)y(k-m) + \\ &\quad + \hat{b}_1(k-1)u(k-1-d) + \dots + \\ &\quad + \hat{b}_m(k-m)u(k-m-d) = \\ &= \Psi^T(k)\hat{\theta}(k-1) \end{aligned} \quad (4.1.5)$$

$$\boxed{\hat{y}(k) = \Psi^T(k)\hat{\theta}(k-1)} \quad (4.1.6)$$

Where

$$\begin{aligned}\Psi^T(k) &= \begin{bmatrix} -y(k-1) & \dots & -y(k-m) & u(k-d-1) & \dots & u(k-d-m) \end{bmatrix} \\ \hat{\theta}(k) &= \begin{bmatrix} \hat{a}_1 & \dots & \hat{a}_m & \hat{b}_1 & \dots & \hat{b}_m \end{bmatrix}\end{aligned}\quad (4.1.7)$$

The measured output can be described as:

$$y(k) = \Psi^T(k)\hat{\theta}(k-1) + e(k) \quad (4.1.8)$$

Now, the input  $u(k)$  and the output  $y(k)$  are sampled for the data points  $k = 1, 2, \dots, m+d+N$  and then **stored**. In order to determine the  $2m$  parameters,  $N \geq 2m-1$  equations are necessary (but for noise suppression is better using  $N \gg 2m-1$ ).

These  $N+1$  equations can be written in matrix form as:

$$\mathbf{y}(m+d+N) = \Psi^T(m+d+N)\hat{\theta}(m+d+N-1) + \mathbf{e}(m+d+N) \quad (4.1.9)$$

where

$$\mathbf{y}^T(m+d+N) = \begin{bmatrix} y(m+d) & y(m+d+1) & \dots & y(m+d+N) \end{bmatrix}$$

and

$$\Psi(m+d+N) = \begin{bmatrix} -y(m+d-1) & \dots & -y(d) & u(m-1) & \dots & u(0) \\ -y(m+d) & \dots & -y(d+1) & u(m) & \dots & u(1) \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ -y(m+d+N-1) & \dots & -y(d+N) & u(m+N-1) & \dots & u(N) \end{bmatrix}$$

where the size of the matrices is:

$$\mathbf{y} \rightarrow (N+1) \times 1, \quad \Psi \rightarrow (N+1) \times 2m, \quad \mathbf{e} \rightarrow (N+1) \times 1$$

The cost function can be written as:

$$V = \mathbf{e}^T \mathbf{e} = \sum_{k=m+d}^{m+d+N} e^2(k) \quad (4.1.10)$$

To calculate the minimum of the cost function, we have to calculate the derivative of the function, taking into account the relation

$$\frac{\partial}{\partial \mathbf{x}}(\mathbf{A}\mathbf{x}) = \mathbf{A}^T, \quad \frac{\partial}{\partial \mathbf{x}}\mathbf{x}^T \mathbf{A} = \mathbf{A}, \quad \frac{\partial}{\partial \mathbf{x}}(\mathbf{x}^T \mathbf{A}\mathbf{x}) = 2\mathbf{A}\mathbf{x}$$

we obtain:

$$\begin{aligned} \frac{dV}{d\theta} \Big|_{\theta=\hat{\theta}} &= \frac{d}{d\hat{\theta}} \left( \mathbf{y}^T \mathbf{y} - \hat{\theta}^T \Psi^T \mathbf{y} - \mathbf{y}^T \Psi \hat{\theta} + \hat{\theta}^T \Psi^T \hat{\theta} \Psi \right) \\ &= -2\Psi^T \mathbf{y} + 2\Psi^T \Psi \hat{\theta} \end{aligned} \quad (4.1.11)$$

The minimum of the cost function is derived by equating to zero the derivate function:

$$\boxed{\begin{aligned} \frac{dV}{d\theta} \Big|_{\theta=\hat{\theta}} &= -2\Psi^T \mathbf{y} + 2\Psi^T \Psi \hat{\theta} \doteq 0 \\ \Downarrow \\ \hat{\theta} &= \mathbf{P} \Psi^T \mathbf{y} \\ \Downarrow \\ \mathbf{P} &= (\Psi^T \Psi)^{-1} \end{aligned}} \quad (4.1.12)$$

The equation  $\hat{\theta} = \mathbf{P} \Psi^T \mathbf{y}$  represents the estimation of the parameters of the system under investigation. Moreover, the symmetric matrix  $\mathbf{P} = (\Psi^T \Psi)^{-1}$  as well as the vector  $\Psi^T \mathbf{y}$  contains very important information about the properties of the least square method. By a simple verification is it possible to see that the elements of the matrix  $\mathbf{P} = (\Psi^T \Psi)^{-1}$  as well as of the matrix  $\Psi^T \mathbf{y}$  are auto and cross-correlation functions with different starting and ending points.

$$(N+1)^{-1} \Psi^T \Psi =$$

$$= \begin{bmatrix} R_{yy}(0) & R_{yy}(1) & \dots & R_{yy}(m-1) & -R_{uy}(d) & \dots & -R_{uy}(d+m-1) \\ R_{yy}(1) & R_{yy}(0) & \dots & R_{yy}(m-2) & -R_{uy}(d-1) & \dots & -R_{uy}(d+m-2) \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ & \dots & R_{yy}(0) & -R_{uy}(d-m-1) & \dots & -R_{uy}(d) \\ & & & R_{uu}(0) & \dots & R_{uu}(m-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ & & & & & & R_{uu}(0) \end{bmatrix}$$

where

$$\begin{aligned}
 R_{yy}(0) &= \sum_{k=m+d-1}^{m+d+N-1} y^2(k) \\
 R_{yy}(1) &= \sum_{k=m+d-2}^{m+d+N-2} y(k)y(k+1) \\
 R_{yy}(m-1) &= \sum_{k=m+d-1}^{m+d+N-1} y(k)y(k-m+1) \\
 R_{uy}(d) &= \sum_{k=m+d-1}^{m+d+N-1} y(k)u(k-d) \\
 &\vdots
 \end{aligned}$$

## 4.2 Recursive Method of Least Squares (LS)

In order to implement an on-line estimation, the above implementation of least square method, requires storing  $N + 1$  data and to calculate for each step the matrix inversion  $\mathbf{P} = (\Psi^T \Psi)^{-1}$ . The computational effort makes this method not useful for an online parameter estimator. As proposed by Gauss in the original work (1821), least squares method can be manipulated in a way to achieve a recursive algorithm. The recursive method of least squares belongs to the topic of the adaptive filter.

From non-recursive method, we obtain the fundamental equation of the least squares estimation

$$\hat{\theta}(k) = \mathbf{P}(k)\Psi^T(k)\mathbf{y}(k) \quad (4.2.1)$$

$$\mathbf{P}(k) = (\Psi^T \Psi)^{-1} \quad (4.2.2)$$

$$\mathbf{y}(k) = \begin{bmatrix} y(1) \\ \vdots \\ y(k) \end{bmatrix} \quad (4.2.3)$$

$$\Psi(k) = \begin{bmatrix} \Psi^T(1) \\ \vdots \\ \Psi^T(k) \end{bmatrix} \quad (4.2.4)$$

where

$$\Psi^T(k) = \begin{bmatrix} -y(k-1) & \dots & -y(k-m) & u(k-d-1) & \dots & u(k-d-m) \end{bmatrix} \quad (4.2.5)$$

Accordingly the parameter estimate for sample step  $k + 1$  is given as

$$\begin{aligned}\hat{\theta}(k) &= \mathbf{P}(k) \begin{bmatrix} \Psi(k-1) \\ \Psi^T(k) \end{bmatrix}^T \begin{bmatrix} \mathbf{y}(k-1) \\ y(k) \end{bmatrix} = \\ &= \mathbf{P}(k) (\Psi^T(k-1)\mathbf{y}(k-1) + \Psi(k)y(k))\end{aligned}\quad (4.2.6)$$

Since  $\mathbf{P}^{-1}(k)\hat{\theta}(k) = \Psi^T(k)\mathbf{y}(k)$  we can write as follows

$$\begin{aligned}\hat{\theta}(k) &= \mathbf{P}(k) (\Psi^T(k-1)\mathbf{y}(k-1) + \Psi(k)y(k)) \\ &\Downarrow \\ \mathbf{P}^{-1}(k)\hat{\theta}(k) &= \Psi^T(k)\mathbf{y}(k) \\ &\Downarrow \\ \hat{\theta}(k) &= \mathbf{P}(k) (\mathbf{P}^{-1}(k-1)\hat{\theta}(k-1) + \Psi(k)y(k)) \\ &\Downarrow \\ \hat{\theta}(k) &= \mathbf{P}(k)\mathbf{P}^{-1}(k-1)\hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)y(k) \\ &\Downarrow \\ \hat{\theta}(k) &= \hat{\theta}(k-1) + \mathbf{P}(k)\mathbf{P}^{-1}(k-1)\hat{\theta}(k-1) - \hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)y(k) \\ &\Downarrow \\ \hat{\theta}(k) &= \hat{\theta}(k-1) + (\mathbf{P}(k)\mathbf{P}^{-1}(k-1) - \mathbf{I})\hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)y(k)\end{aligned}\quad (4.2.7)$$

The above equation gives the predicted value of the  $\hat{\theta}(k+1)$  as function of  $\hat{\theta}(k)$  and  $\mathbf{P}(k+1)$ . Now it is necessary to find a way to calculate  $\mathbf{P}(k)$  in a recursive way. As above we can write  $\mathbf{P}(k+1)$  as:

$$\begin{cases} \mathbf{P}(k) = \left( \begin{bmatrix} \Psi(k-1) \\ \Psi^T(k) \end{bmatrix}^T \begin{bmatrix} \Psi(k-1) \\ \Psi^T(k) \end{bmatrix} \right)^{-1} = (\mathbf{P}^{-1}(k-1) + \Psi(k)\Psi^T(k))^{-1} \\ \mathbf{P}^{-1}(k-1) = \mathbf{P}^{-1}(k) + \Psi(k)\Psi^T(k) \end{cases}\quad (4.2.8)$$

Hence considering that  $\mathbf{P}^{-1}(k-1) = \mathbf{P}^{-1}(k) + \Psi(k)\Psi^T(k)$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + (\mathbf{P}(k)\mathbf{P}^{-1}(k-1) - \mathbf{I})\hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)y(k)$$

↓

$$\mathbf{P}^{-1}(k-1) = \mathbf{P}^{-1}(k) + \Psi(k)\Psi^T(k)$$

↓

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)\Psi^T(k)\hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)y(k)$$

↓

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \mathbf{P}(k)\Psi(k)(y(k) - \Psi^T(k)\hat{\theta}(k-1))$$

**Now the question is how to calculate  $\mathbf{P}(k) = (\mathbf{P}^{-1}(k-1) + \Psi(k)\Psi^T(k))^{-1}$**

Considering the matrix inversion lemma:

$$\mathbf{E} = (\mathbf{A}^{-1} + \mathbf{B}\mathbf{B}^T)^{-1} = \mathbf{A} - \mathbf{A}\mathbf{B}(\mathbf{B}^T\mathbf{A}\mathbf{B} + \mathbf{1})^{-1}\mathbf{B}^T\mathbf{A} \quad (4.2.9)$$

We obtain

$$\begin{aligned} \mathbf{P}(k) &= (\mathbf{P}^{-1}(k-1) + \Psi(k)\Psi^T(k))^{-1} = \\ &= \mathbf{P}(k-1) + \Psi^T(k)\mathbf{P}(k-1) \frac{\mathbf{P}(k-1)\Psi(k)}{\Psi^T(k)\mathbf{P}(k-1)\Psi(k) + 1} \end{aligned}$$

↓

$$\mathbf{P}(k)\Psi(k) = \mathbf{P}(k-1)\Psi(k) \left( 1 - \frac{\Psi^T(k)\mathbf{P}(k-1)\Psi(k)}{\Psi^T(k)\mathbf{P}(k-1)\Psi(k) + 1} \right)$$

↓

$$\mathbf{P}(k)\Psi(k) = \mathbf{P}(k-1)\Psi(k) \frac{1}{\Psi^T(k)\mathbf{P}(k-1)\Psi(k) + 1}$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \gamma(k-1) (y(k) - \Psi^T(k)\hat{\theta}(k-1))$$

$$e(k) = y(k) - \Psi^T(k)\hat{\theta}(k-1) \quad (4.2.10)$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \gamma(k-1)e(k) \quad (4.2.11)$$

$$\begin{aligned} \gamma(k) &= \mathbf{P}(k)\Psi(k) = \\ &= \mathbf{P}(k-1)\Psi(k) \frac{1}{\Psi^T(k)\mathbf{P}(k-1)\Psi(k) + 1} \end{aligned} \quad (4.2.12)$$

$$\mathbf{P}(k) = \mathbf{P}(k-1) - \gamma(k)\Psi^T(k)\mathbf{P}(k-1) \quad (4.2.13)$$

Where equations (4.2.10), (4.2.11), (4.2.12) and (4.2.13) represents the algorithm of the recursive least square method.

## DESCRIPTION OF THE ALGORITHM

1. Start cycle  $k$
2. Update  $y(k)$  and  $u(k)$
3. Calculation of the error

$$e(k) = y(k) - \Psi^T(k)\hat{\theta}(k-1)$$

4. Estimation of the parameter vector

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \gamma(k-1)e(k)$$

5.

$$\mathbf{P}(k)\Psi(k) = \begin{bmatrix} p_{11}(k-1) & \dots & p_{12m}(k-1) \\ \vdots & \ddots & \vdots \\ p_{2m1}(k-1) & \dots & p_{2m2m}(k-1) \end{bmatrix} \begin{bmatrix} -y(k-1) \\ \vdots \\ -y(k-m) \\ u(k-2) \\ \vdots \\ u(k-m-1) \end{bmatrix} =$$

$$= \begin{bmatrix} i_1(k) \\ \vdots \\ i_{2m}(k) \end{bmatrix} = \mathbf{i}(k)$$

6.

$$\gamma(k) = \frac{1}{j+1} \mathbf{i}(k),$$

$$\Psi^T(k)\mathbf{P}(k-1)\Psi(k) = j$$

7. Calculation of the correlation matrix

$$\mathbf{P}(k) = \mathbf{P}(k-1) - \gamma(k)\mathbf{i}^T(k) =$$

$$= \begin{bmatrix} p_{11}(k-1) - \gamma_1 i_1 & \dots & p_{12m}(k-1) - \gamma_1 i_{2m} \\ \vdots & \ddots & \vdots \\ p_{2m1}(k-1) - \gamma_{2m} i_1 & \dots & p_{2m2m}(k-1) - \gamma_{2m} i_{2m} \end{bmatrix}$$

8. Update the matrix  $\Psi^T$  for the next cycle

$$\Psi^T(k+1) = (-y(k) \ \dots \ -y(k-m) \ \ u(k-1) \ \dots \ u(k-m-1))$$

9. End of the cycle  $k$  and go back to position (1.) for cycle  $k+1$

# Chapter 5

## The Kalman Theory

### 5.1 The linear least-squares problem

Least square problems have a long history in science, they arise in solving an over-determined set of equations. Consider  $\mathbf{F} \in \mathbb{R}^{m \times n}$ ,  $\vec{y} \in \mathbb{R}^m$ ,  $\vec{x} \in \mathbb{R}^n$  and the equation

$$\mathbf{F}\vec{x} = \vec{y} \quad (5.1.1)$$

which is a compact matrix notation for the set of linear equations

$$\begin{aligned} f_{11}x_1 + f_{12}x_2 + \cdots + f_{1n}x_n &= y_1 \\ f_{21}x_1 + f_{22}x_2 + \cdots + f_{2n}x_n &= y_2 \\ &\vdots \\ f_{m1}x_1 + f_{m2}x_2 + \cdots + f_{mn}x_n &= y_m \end{aligned}$$

The matrix-vector product  $\mathbf{F}\vec{x}$  geometrically means that we seek linear combination of the columns of the matrix  $\mathbf{F}$  that equal the vector  $\vec{y}$ . Therefore, a solution  $\vec{x}$  to Eq. (5.1.1) exists only provided that the vector  $\vec{y}$  lies in the column space of the matrix  $\mathbf{F}$ . When the vector  $\vec{y}$  satisfies this condition, we say that the set of equation in Eq. (5.1.1) *consistent*, otherwise, it is called *inconsistent*.

To find a solution to an inconsistent set of equations, we pose the following least-square problem:

$$\min_{\vec{x}} \vec{e}^T \vec{e} \quad \text{subject to } \vec{y} = \mathbf{F}\vec{x} + \vec{e} \quad (5.1.2)$$

That is, we seek a vector  $\vec{x}$  that minimizes the norm of the residual vector  $\vec{e}$ , such that  $\vec{y} \approx \mathbf{F}\vec{x}$ . The residual vector  $\vec{e} \in \mathbb{R}^m$  can be eliminated from the above problem formulation, thus reducing it to the more standard form

$$\min_{\vec{x}} \|\mathbf{F}\vec{x} - \vec{y}\|_2^2 \quad (5.1.3)$$

The cost function of the least-squares minimization problem Eq. (5.1.3) can be expanded as

$$\begin{aligned} J(\vec{x}) &= \|\mathbf{F}\vec{x} - \vec{y}\|_2^2 \\ &= (\mathbf{F}\vec{x} - \vec{y})^T (\mathbf{F}\vec{x} - \vec{y}) \\ &= \vec{x}^T \mathbf{F}^T \mathbf{F}\vec{x} - \vec{x}^T \mathbf{F}^T \vec{y} - \vec{y}^T \mathbf{F}\vec{x} + \vec{y}^T \vec{y} \end{aligned} \quad (5.1.4)$$

$$\frac{\partial}{\partial \vec{x}^T} J(\vec{x}) = \begin{bmatrix} \frac{\partial J(\vec{x})}{\partial x_1} \\ \vdots \\ \frac{\partial J(\vec{x})}{\partial x_n} \end{bmatrix} = 2\mathbf{F}^T \mathbf{F}\vec{x} - 2\mathbf{F}^T \vec{y} \quad (5.1.5)$$

The solution  $\hat{\vec{x}}$  to the least squares problem Eq. (5.1.3) is found by setting the gradient  $\partial J(\vec{x})/\partial \vec{x}^T = \vec{0}$ . This yields the so called *normal equations*,

$$\mathbf{F}^T \mathbf{F}\hat{\vec{x}} = \mathbf{F}^T \vec{y} \quad (5.1.6)$$

and if the matrix  $\mathbf{F}$  has full rank

$$\hat{\vec{x}} = (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \vec{y} \quad (5.1.7)$$

Define

$$\vec{e} = \vec{y} - \hat{\vec{y}} = \vec{y} - \mathbf{F}\hat{\vec{x}} \quad (5.1.8)$$

we have

$$\vec{e}^T \mathbf{F} = \vec{0} \quad (5.1.9)$$

which means the error vector  $\vec{e} = \vec{y} - \hat{\vec{y}}$  is orthogonal to the plane spanned by  $\vec{f}_1$  and  $\vec{f}_2$ .

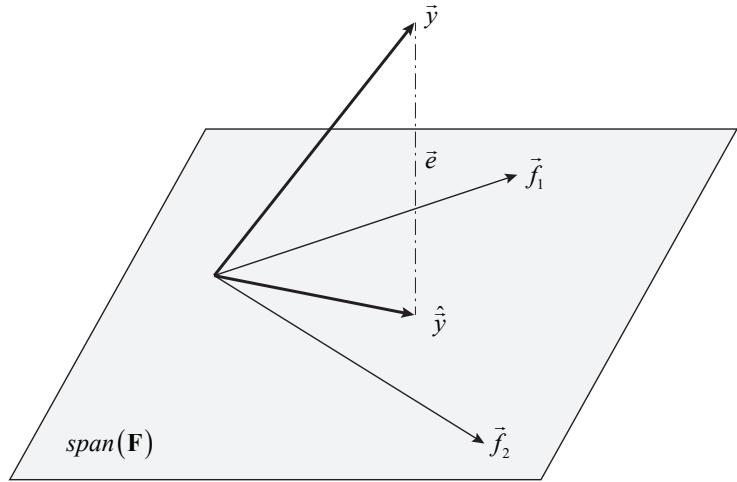


Figure 5.1: The solution to the least squares problem. The matrix  $\mathbf{F}$  is  $3 \times 2$ ; its two column vectors  $\vec{f}_1$  and  $\vec{f}_2$  span a two-dimensional plane in a three-dimensional space. The vector  $\vec{y}$  does not lie in this plane. The projection of  $\vec{y}$  onto the plane spanned by  $\vec{f}_1$  and  $\vec{f}_2$  is the vector  $\hat{\vec{y}} = \mathbf{F}\hat{\vec{x}}$ . The residual  $\vec{e} = \vec{y} - \hat{\vec{y}}$  is orthogonal to the plane spanned by  $\vec{f}_1$  and  $\vec{f}_2$ .

**Example 5.1 (Over-determined set of linear equations).** Given

$$\mathbf{F} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \\ 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},$$

consider the set of three equations in two unknowns,  $\mathbf{F}\vec{x} = \vec{y}$ . The least-squares solution is given by

$$\begin{aligned} \hat{\vec{x}} &= (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \vec{y} \\ &= \begin{bmatrix} \frac{2}{3} & -2 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -\frac{1}{2} \\ 1 \end{bmatrix} \end{aligned}$$

and

$$\vec{e}^T \mathbf{F} = \vec{0}$$

Which means the residual error  $\vec{e}$  is orthogonal to the span\* of  $(\vec{f}_1, \vec{f}_2)$ . Where span of  $\mathbf{F}$  is the subspace generated by the linear combination of the vectors  $(\vec{f}_1, \vec{f}_2)$ .  $\triangleleft \triangleleft$

---

\*The column space of  $\mathbf{F}$  is *spanned* by the columns. Their combination produce the whole space. if a vector space  $\mathbf{V}$  consist of all linear combinations of  $\vec{f}_1, \dots, \vec{f}_n$  then these vectors **span** the space. Every vector  $\vec{v} \in \mathbf{V}$  is some combination of the vectors  $\vec{f}_i$ , such that  $\vec{v} = \lambda_1 \vec{f}_1 + \dots + \lambda_n \vec{f}_n$

Hence, the best estimation  $\hat{\vec{x}} = (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \vec{y}$  generates a projection  $\hat{\vec{y}} = \mathbf{F} \hat{\vec{x}}$  such that  $(\vec{y} - \mathbf{F} \hat{\vec{x}})^T \mathbf{F} = \vec{0}$

## 5.2 The Kalman-filter problem

Like the observer, the Kalman filter is a filter that approximates the state vector of a dynamical system from measurements of the input and output sequences. The main difference from the observer is that the Kalman filter takes noise disturbances into account. Consider the following LTI system corrupted by two noise sequences  $q(k)$  and  $r(k)$

$$\begin{aligned}\vec{x}(k+1) &= \mathbf{A} \vec{x}(k) + \mathbf{B} \vec{u}(k) + \vec{q}(k) \\ \vec{y}(k) &= \mathbf{C} \vec{x}(k) + \vec{r}(k)\end{aligned}\tag{5.2.1}$$

The vector  $\vec{q}(k) \in \mathbb{R}^m$  is called the *process noise* and  $\vec{r}(k) \in \mathbb{R}^l$  is called the *measurement noise*.

If we now use the observer equation (see Eq. (5.2.2)) to reconstruct the state of the system Eq. (5.2.1),

$$\hat{\vec{x}}(k+1) = \mathbf{A} \hat{\vec{x}}(k) + \mathbf{B} \vec{u}(k) + \mathbf{L}(k) \left( \vec{y}(k) - \mathbf{C} \hat{\vec{x}}(k) \right)\tag{5.2.2}$$

the difference  $\vec{e}(k) = \vec{x}(k) - \hat{\vec{x}}(k)$  between the estimated state  $\hat{\vec{x}}(k+1)$  and the real state  $\vec{x}(k+1)$  satisfies

$$\vec{e}(k+1) = (\mathbf{A} - \mathbf{L}(k) \mathbf{C}) \vec{e}(k) + \vec{q}(k) - \mathbf{L} \vec{r}(k)\tag{5.2.3}$$

Even if  $\mathbf{A} - \mathbf{L}(k) \mathbf{C}$  were asymptotic stable,  $\vec{e}(k)$  would not go to zero for  $k \rightarrow \infty$ , because of the presence of  $\vec{q}(k)$  and  $\vec{r}(k)$ . Now, the goal is to make  $\vec{e}(k)$  *small*, because then the state estimate will be close to the real state. Since  $\vec{e}(k)$  is affected by a random signal, we can try to make the mean of  $\vec{e}(k)$  equal to zero:  $E[\vec{e}(k)] = E[\hat{\vec{x}}(k) - \vec{x}(k)] = 0$ . However, this does not mean that  $\vec{e}(k)$  will be *small*; it can still very wildly around its mean zero. Therefore in addition, we want to make the state-error covariance matrix  $E[\vec{e}(k) \vec{e}^T(k)]^\dagger$  as small as possible; that is, we are looking for a minimum-error variance estimate. Due to the stochastic nature of the signal

$${}^\dagger \vec{e} \vec{e}^T = \begin{bmatrix} e_1 e_1 & \dots & e_1 e_n \\ \vdots & \ddots & \vdots \\ e_n e_1 & \dots & e_n e_n \end{bmatrix}$$

$\vec{q}(k)$  and  $\vec{r}(k)$  we can write the following working hypothesis:

$$\begin{aligned} E[\vec{q}(k)] &= \vec{0} \\ E[\vec{q}(i)\vec{q}^T(j)] &= \mathbf{Q}\delta_{ij} \quad \delta_{ij} = 0 \text{ for } i \neq j \text{ and } \delta_{ii} = 1 \\ E[\vec{r}(k)] &= \vec{0} \\ E[\vec{r}(i)\vec{r}^T(j)] &= \mathbf{R}\delta_{ij} \\ E[\vec{r}(i)\vec{q}^T(j)] &= \mathbf{0} \quad \text{for all } i, j \end{aligned} \tag{5.2.4}$$

The predicted (*a priori*) state vector at time  $k t_s$  or  $k$  is based on a state estimate at time  $k-1$ . Letting  $\hat{\vec{x}}(k|k-1)$  denote the *a posteriori* state estimate at time  $k-1$  based on measurements up to and including time  $k-1$ , the *a priori* state at time  $k$  is

$$\hat{\vec{x}}(k|k) = \mathbf{A}\hat{\vec{x}}(k|k-1) + \mathbf{B}\vec{u}(k|k-1) \quad \text{a priori} \tag{5.2.5}$$

The *a priori* calculation of the error covariant matrix is computed as follows

$$\begin{aligned} \mathbf{P}(k|k) &= E[\vec{e}(k|k)\vec{e}^T(k|k)] = \\ &= E\left[\left(\mathbf{A}\vec{e}(k|k-1) + \vec{q}(k-1)\right)\left(\vec{e}^T(k|k-1)\mathbf{A}^T + \vec{q}^T(k-1)\right)\right] \\ &= \mathbf{A}E[\vec{e}(k|k-1)\vec{e}^T(k|k-1)]\mathbf{A}^T + E[\vec{q}(k-1)\vec{q}^T(k-1)] \\ &= \mathbf{A}\mathbf{P}(k|k-1)\mathbf{A}^T + \mathbf{Q} \end{aligned} \tag{5.2.6}$$

where

$$\begin{aligned} \vec{e}(k|k) &= \vec{x}(k) - \hat{\vec{x}}(k|k) = \\ &= \mathbf{A}\vec{x}(k-1) + \mathbf{B}\vec{u}(k-1) + \vec{q}(k-1) - \mathbf{A}\hat{\vec{x}}(k|k-1) + \mathbf{B}\vec{u}(k|k-1) \\ &= \mathbf{A}\left[\vec{x}(k-1) - \hat{\vec{x}}(k|k-1)\right] + \vec{q}(k-1) = \mathbf{A}\vec{e}(k|k-1) + \vec{q}(k-1) \end{aligned}$$

and  $\vec{e}(k|k-1) = \vec{x}(k) - \hat{\vec{x}}(k|k-1)$ .

The output estimation error becomes

$$\Delta\vec{y}(k) = \vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k) = \mathbf{C}\vec{x}(k) + \vec{r}(k) - \mathbf{C}\hat{\vec{x}}(k|k) = \mathbf{C}\vec{e}(k|k) + \vec{r}(k) \tag{5.2.7}$$

where  $\vec{r}(k)$  is white noise in discrete time domain.

Notice that  $E[\Delta \vec{y}(k)] = \vec{0}$  and the covariance of the measurement residual  $\Delta \vec{y}(k)$  is

$$E[\Delta \vec{y}(k) \Delta \vec{y}^T(k)] = \mathbf{C} \mathbf{P}(k|k) \mathbf{C}^T + \mathbf{R} \quad (5.2.8)$$

the *a posteriori* estimation of the vector state is calculated as follows

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \Delta \vec{y}(k) \quad a \text{ posteriori} \quad (5.2.9)$$

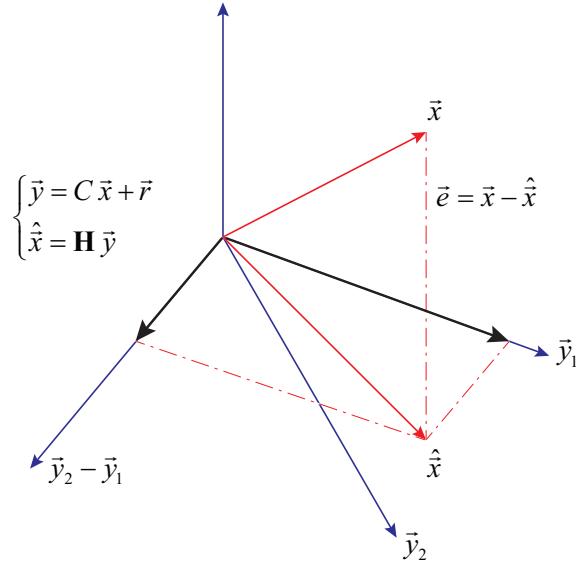


Figure 5.2: Illustration of the orthogonality principle.

The gain matrix  $\mathbf{L}(k)$  which minimizes the covariance of  $\vec{e}(k+1|k) = \vec{x}(k) - \hat{\vec{x}}(k+1|k)$  is computed using the **orthogonality principle** (see Figure 5.2):

$$\begin{aligned} E[(\vec{x}(k) - \hat{\vec{x}}(k+1|k)) \vec{y}^T(k)] &= E[(\vec{e}(k|k) - \mathbf{L}(k)(\mathbf{C}\vec{e}(k|k) + \vec{r}(k))) (\mathbf{C}\vec{x}(k) + \vec{r}(k))^T] \\ &= E[( (\mathbf{I} - \mathbf{L}(k)\mathbf{C})\vec{e}(k|k) - \mathbf{L}\vec{r}(k)) ((\hat{\vec{x}}(k|k) + \vec{e}(k|k))^T \mathbf{C}^T + \vec{r}^T(k)) ] \\ &= (\mathbf{I} - \mathbf{L}(k)\mathbf{C}) \mathbf{P}(k|k) \mathbf{C}^T - \mathbf{L}(k) \mathbf{R} \\ &= \mathbf{0} \end{aligned} \quad (5.2.10)$$

where

$$[\mathbf{C}\vec{x}(k) + \vec{r}(k)]^T = [\hat{\vec{x}}(k|k) + \vec{e}(k|k)]^T \mathbf{C}^T + \vec{r}^T(k)$$

and where

$$\begin{aligned}\vec{x}(k) - \hat{\vec{x}}(k+1|k) &= \vec{e}(k|k) - \mathbf{L}(k) [\vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k)] = \\ &= \vec{e}(k|k) - \mathbf{L}(k) [\mathbf{C}\vec{x}(k) - \mathbf{C}\hat{\vec{x}}(k|k) + \vec{r}(k)] = \\ &= [\mathbf{I} - \mathbf{L}(k)\mathbf{C}] \vec{e}(k|k) - \mathbf{L}(k)\vec{r}(k)\end{aligned}$$

We have assumed  $E[\vec{e}(k|k) \hat{\vec{x}}^T(k|k)] = \mathbf{0}$  (from **orthogonality principle**), hence the  $n \times m$  gain matrix is given as follows

$$\mathbf{L}(k) = \mathbf{P}(k|k)\mathbf{C}^T (\mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R})^{-1} \quad (5.2.11)$$

hence the *a posteriori* state estimate becomes

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \Delta \vec{y}(k) \quad a \text{ posteriori} \quad (5.2.12)$$

and the *a posteriori* state error covariance is  $\vec{e}(k+1|k) = \vec{x}(k) - \hat{\vec{x}}(k+1|k)$

$$\begin{aligned}\mathbf{P}(k+1|k) &= E[\vec{e}(k+1|k) \vec{e}^T(k+1|k)] = \\ &= E\left[\left(\left(\mathbf{I} - \mathbf{L}\mathbf{C}\right)\vec{e}(k|k) - \mathbf{L}(k)\vec{r}(k)\right)\left(\left(\mathbf{I} - \mathbf{L}\mathbf{C}\right)\vec{e}(k|k) - \mathbf{L}(k)\vec{r}(k)\right)^T\right] \quad (5.2.13) \\ &= \left(\mathbf{I} - \mathbf{L}\mathbf{C}\right)\mathbf{P}(k|k)\left(\mathbf{I} - \mathbf{L}\mathbf{C}\right)^T + \mathbf{L}(k)\mathbf{R}\mathbf{L}^T(k)\end{aligned}$$

substituting Eq. (5.2.11) into Eq. (5.2.13) we obtain

$$\begin{aligned}\mathbf{P}(k+1|k) &= \mathbf{P}(k|k) - \mathbf{L}(k)\mathbf{C}\mathbf{P}(k|k) - \mathbf{P}(k|k)\mathbf{C}^T\mathbf{L}^T(k) + \mathbf{L}(k)\left(\mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R}\right)\mathbf{L}^T(k) \\ &= \mathbf{P}(k|k) - \mathbf{L}(k)\mathbf{C}\mathbf{P}(k|k) - \mathbf{P}(k|k)\mathbf{C}^T\mathbf{L}^T(k) \\ &\quad + \mathbf{P}(k|k)\mathbf{C}^T\left(\mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R}\right)^{-1}\left(\mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R}\right)\mathbf{L}^T(k) \\ &= \left(\mathbf{I} - \mathbf{L}(k)\mathbf{C}\right)\mathbf{P}(k|k) \quad (5.2.14)\end{aligned}$$

In summary

### Kalman Filter Algorithm

$$\hat{\vec{x}}(k|k) = \mathbf{A}\hat{\vec{x}}(k|k-1) + \mathbf{B}\vec{u}(k|k-1) \quad a \text{ priori} \quad (5.2.15)$$

$$\mathbf{P}(k|k) = \mathbf{A}\mathbf{P}(k|k-1)\mathbf{A}^T + \mathbf{Q} \quad a \text{ priori} \quad (5.2.16)$$

$$\mathbf{L}(k) = \mathbf{P}(k|k)\mathbf{C}^T (\mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R})^{-1} \quad (5.2.17)$$

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \left( \vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k) \right) \quad a \text{ posteriori} \quad (5.2.18)$$

$$\mathbf{P}(k+1|k) = (\mathbf{I} - \mathbf{L}(k)\mathbf{C})\mathbf{P}(k|k) \quad a \text{ posteriori} \quad (5.2.19)$$

where in Figure 5.3 graphical representation is shown.

Finally we mention one more important practical aspect of the Kalman filter. We see from Eq. (5.2.23), Eq. (5.2.24) and Eq. (5.2.26) that the calculation of  $\mathbf{P}(k|k)$ ,  $\mathbf{L}(k)$  and  $\mathbf{P}(k+1|k)$  does not depend on the measurement  $\vec{y}(k)$ , but depends only on the system parameters  $\mathbf{A}$ ,  $\mathbf{C}$ ,  $\mathbf{Q}$  and  $\mathbf{R}$ . That means that the Kalman gain  $\mathbf{L}(k)$  can be calculated offline as  $\mathbf{L}^{kalman}$ . Then the only closed loop observer can be implemented resulting as follows

$$\hat{\vec{x}}(k+1|k) = \mathbf{A}\hat{\vec{x}}(k|k-1) + \mathbf{B}\vec{u}(k|k-1) + \mathbf{L}^{kalman} \left( \vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k) \right) \quad (5.2.20)$$

In contrast, the Kalman filter gain  $\mathbf{L}(k)$  and covariance  $\mathbf{P}(k)$  for nonlinear systems cannot (in general) be computed offline because they depend on the measurement as can be seen as follows: Consider the nonlinear system

$$\begin{aligned} \vec{x}(k+1) &= f(\vec{x}(k)) + \mathbf{B}\vec{u}(k) + \vec{q}(k) \\ \vec{y}(k) &= \mathbf{C}\vec{x}(k) + \vec{r}(k) \end{aligned} \quad (5.2.21)$$

where we apply the concept of the Extended Kalman filter

### Extended Kalman Filter Algorithm

$$\hat{\vec{x}}(k|k) = f(\hat{\vec{x}}(k|k-1)) + \mathbf{B}\vec{u}(k|k-1) \quad a \text{ priori} \quad (5.2.22)$$

$$\mathbf{P}(k|k) = \mathbf{A}(\hat{\vec{x}}(k|k-1))\mathbf{P}(k|k-1)\mathbf{A}^T(\hat{\vec{x}}(k|k-1)) + \mathbf{Q} \quad a \text{ priori} \quad (5.2.23)$$

$$\mathbf{L}(k) = \mathbf{P}(k|k)\mathbf{C}^T \left( \mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R} \right)^{-1} \quad (5.2.24)$$

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \left( \vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k) \right) \quad a \text{ posteriori} \quad (5.2.25)$$

$$\mathbf{P}(k+1|k) = (\mathbf{I} - \mathbf{L}(k)\mathbf{C})\mathbf{P}(k|k) \quad a \text{ posteriori} \quad (5.2.26)$$

where

$$\mathbf{A}(\hat{\vec{x}}(k|k-1)) = \frac{\partial}{\partial \vec{x}} f(\vec{x}) \Big|_{\vec{x}=\hat{\vec{x}}} = \mathbf{A}(k) \quad (5.2.27)$$

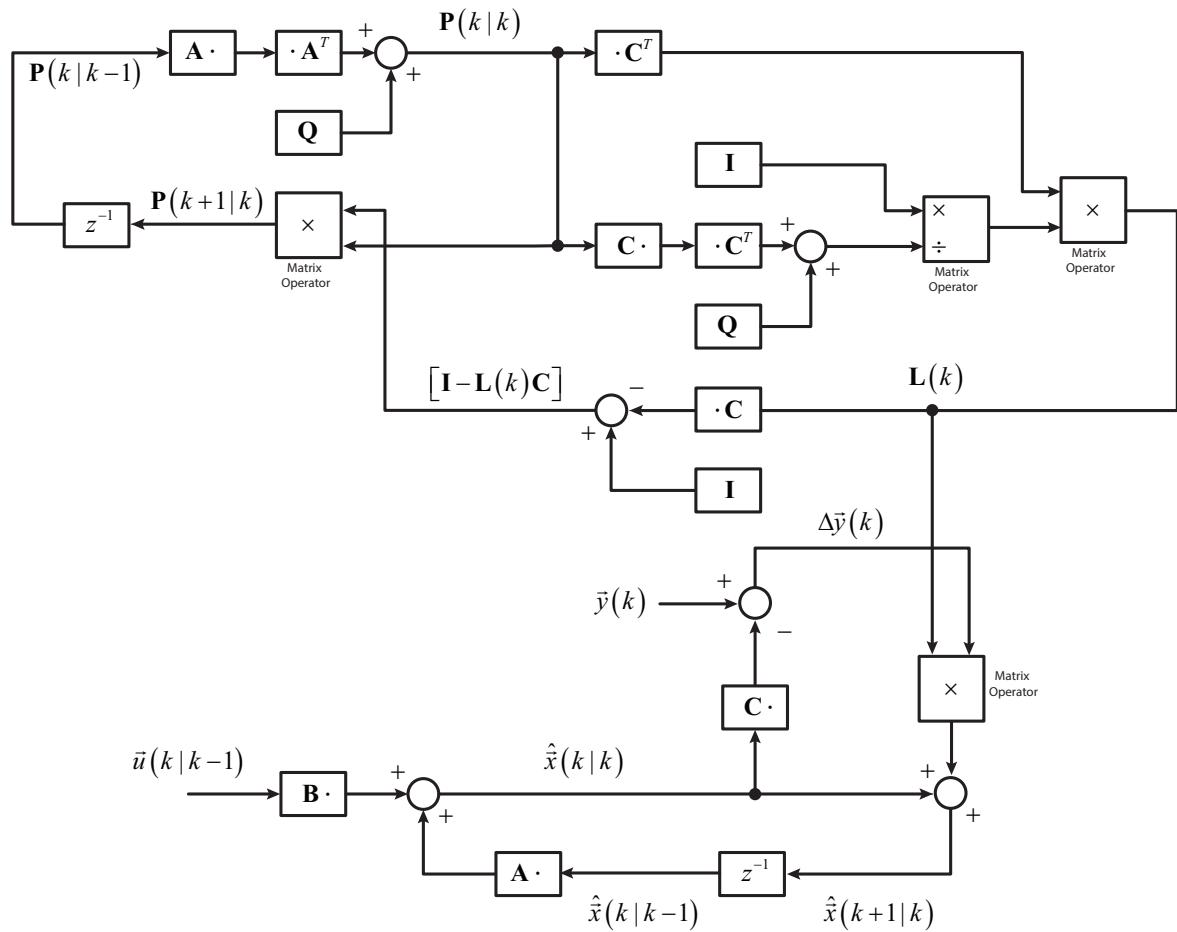


Figure 5.3: Kalman observer implementation.

# Chapter 6

## Model Predictive Control

### 6.1 Predictive Current Control of a Single-Phase RL Load

This section refers to [12].

We start by introducing the notion of predictive control for a single-phase system. Specifically, H-Bridge converter is considered, as shown in Figure 6.1. An  $RL$  load is connected between the phase leg's terminal A and B. Let  $u_{dc}$  denote the instantaneous dc-link voltage. The H-Bridge converter can produce the set of voltage  $\{-u_{dc}, 0, u_{dc}\}$ . We use the integer variable  $u \in \{-1, 0, 1\}$  to denote the switch position of the H-Bridge, hence, the output voltage. The values  $\{-1, 0, 1\}$  correspond to the phase voltages  $\{-u_{dc}, 0, u_{dc}\}$ , respectively.

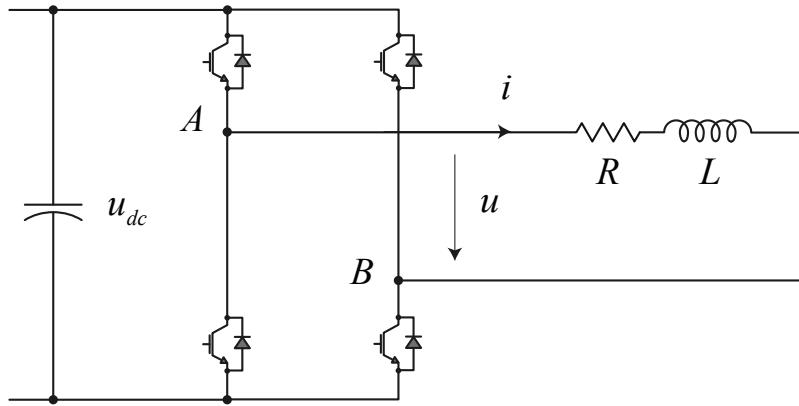


Figure 6.1: H-Bridge inverter with  $RL$  load.

#### 6.1.1 Control Problem

Let  $i^{ref}$  denote the reference for the instantaneous current through the  $RL$  load, and  $i$  the actual current. The aim is to design a controller that manipulates the switches in the H-Bridge such that the load current  $i$  closely tracks its reference  $i^{ref}$ . **Another requirement is to switch as little as possible. Switching directly between  $u = 1$  and  $u = -1$  is prohibited.**

The block diagram of the controller to be designed is shown in Figure 6.2. The controller consist of two parts. Starting from the current measurement, the first part predicts future

current trajectories for different possible choices of control inputs. The second part is the optimization stage, in which the cost function, capturing the control objectives, is minimized. This yields the optimal control input  $u_{opt}$ , which is applied to the inverter. The controller operates in the discrete time domain with the sampling interval  $t_s$ . Each of these blocks is explained in the detail hereafter.

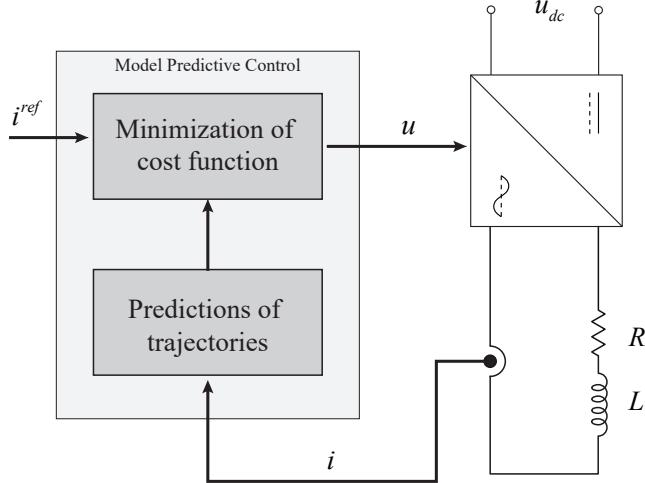


Figure 6.2: Predictive current control.

### 6.1.2 Prediction of Current Trajectories

To predict the future current trajectories, predictive control schemes requires a model that capture the dynamics of the inverter system and its load. Starting from  $u(t) = Ri(t) + L\frac{di(t)}{dt}$ , such a model in the continuous time domain is directly obtained as follows

$$\frac{di(t)}{dt} = \tilde{\mathbf{A}}i(t) + \tilde{\mathbf{B}}u(t) \quad (6.1.1)$$

with

$$\tilde{\mathbf{A}} = -\frac{R}{L} \quad \text{and} \quad \tilde{\mathbf{B}} = \frac{1}{L} \quad (6.1.2)$$

The current  $i$  is the state variable in the state space model of Eq. (7.2.1), and  $u$  is the input variable.

As the controller operates at time instants  $t = kt_s$ , with  $k \in \mathbb{N}$ , this model needs to be translated from the continuous to the discrete-time domain. This is achieved by integrating Eq. (7.2.1) from  $t = kt_s$  to  $t = (k + 1)t_s$ . During this time interval,  $u(t)$  is constant and equal to  $u(k)$ , where  $k$  refers to the  $k$ th sampling interval. The integration yields

$$i(k + 1) = \mathbf{A}i(k) + \mathbf{B}u(k) \quad (6.1.3)$$

with

$$\mathbf{A} = \exp(\tilde{\mathbf{A}}t_s) \quad \text{and} \quad \mathbf{B} = \int_0^{t_s} \exp(\tilde{\mathbf{A}}\tau) d\tau \tilde{\mathbf{B}} \quad (6.1.4)$$

This provides the exact current evolution at the discrete-time instants  $t = kt_s$ . For short sampling intervals, Eq. (7.2.4) can be approximated using e.g. Forward Euler, which yields Eq. (7.2.3) into

$$\mathbf{A} \approx 1 + \tilde{\mathbf{A}}t_s \quad \text{and} \quad \mathbf{B} \approx \tilde{\mathbf{B}}t_s \quad (6.1.5)$$

### 6.1.3 Optimization Problem

The control problem, at time step  $k$ , of tracking the current reference can be mapped into the cost function

$$J = (i^{ref}(k+1) - i(k+1))^2 + \rho (\Delta u(k))^2 \quad (6.1.6)$$

The cost function consist of two terms. The first term penalizes the predicted current error at the next time step  $k+1$  quadratically. The current error is given by the difference between the current reference and the predicted current in the *RL* circuit. The second term penalizes the switching effort, which is the absolute value of  $\Delta u(k) = u(k) - u(k-1)$ . The penalty  $\rho \geq 0$  is a tuning parameter that adjusts the trade-off between the tracking accuracy (deviation of the current from its reference) and the switching effort, that is, whether switching is performed or not.

The current at  $k+1$  depends on the choice of  $u(k)$ . The discrete-time model of Eq. (7.2.3) can be used to predict the current at  $k+1$  for all admissible  $u(k)$ . The admissible  $u(k)$  are those that are either  $-1, 0$  or  $1$  and differ by at most one step from the previous switches state (output voltage), that is,  $u(k-1)$ . Thi optimization problem can formally be state as

$$\begin{cases} u_{opt}(k) = \arg \underbrace{\min}_{u(k)} J \\ \text{subject to } i(k+1) = \mathbf{A}i(k) + \mathbf{B}u(k) \\ u(k) \in \{-1, 0, 1\}, |\Delta u(k)| \leq 1 \end{cases} \quad (6.1.7)$$

The optimal control input, that is, the optimal switch state of the H-Bridge at time step  $k$ ,  $u_{opt}(k)$ , is obtained by minimizing the cost function  $J$ . Note that the expression  $\arg \min J$  minimizes the cost function  $J$  subject to the constraints. This leads to the minimum (or optimum) value  $J_{opt}$ . On the other hand,  $\arg \min J$  yields the argument that minimizes the cost function. The argument is also referred to as the optimal solution or the optimizer, which is, in this case, the control input  $u_{opt}$ .

The optimization problem 7.2.7 and hence  $u_{opt}(k)$  depend on the current  $i(k)$ , the previously chosen switch position  $u(k-1)$ , and the current reference  $i^{ref}(k+1)$ . If the latter is not available, it can be approximated using the current reference at time step  $k$ ,  $i^{ref}(k)$ .

### 6.1.4 Control Algorithm

The optimization problem 7.2.7 can be solved using tools from mathematical programming or a simple alternative is to use the concept of enumeration. For each admissible output voltage,

the model response is predicted and the corresponding cost is computed by evaluating the cost function. The output voltage with the smallest cost is, by definition, the optimal one and chosen as the control input. Depending on  $u(k-1)$ , the set of admissible switch position  $\mathcal{U}(k)$  consists of two or three elements, as shown in table 6.1.

$u(k-1)$	$\mathcal{U}(k)$
1	{0, 1}
0	{-1, 0, 1}
-1	{-1, 0}

Table 6.1: Set of admissible output voltage  $\mathcal{U}(k)$  that meets the switching constraint  $|\Delta u(k)| \leq 1$ .

At time step  $k$ , the predictive current algorithm computes  $u_{opt}(k)$  according to the following procedure:

1. Given the previously applied output voltage  $u(k-1)$  and taking into account the constraints on the switching transitions, the set of admissible output voltage at time step  $k$ ,  $\mathcal{U}(k)$ , is determined.
2. For each output voltage  $u(k) \in \mathcal{U}(k)$ , the current at time step  $k+1$ ,  $i(k+1)$ , is predicted using the model 7.2.3.
3. For each output voltage  $u(k) \in \mathcal{U}(k)$ , the cost function  $J$  is computed according to Eq. (7.2.6).
4. The output voltage  $u_{opt}(k)$  with the minimum cost is determined and applied to the inverter.
5. At the next time step, this procedure is repeated.

Considering the situation depicted in Fig. 6.3. The current at time step  $k$  is closed to its reference. Assuming that  $u(k-1)$  is zero, the set of admissible output voltages at time step  $k+1$  is  $\mathcal{U}(k) = \{-1, 0, 1\}$ . For each  $u(k) \in \mathcal{U}(k)$ , the predicted  $i(k+1)$  is shown in the Fig. 6.3.

The corresponding costs are summarized in table 6.2. Switching to  $u(k) = 1$  minimizes the predicted current error but incurs a switching penalty, which is given by  $\rho$ . Refraining from change the output voltage, that is using  $u(k) = u(k-1) = 0$ , leads to a slightly larger current error while incurring no switching penalty. The third choice,  $u(k) = -1$ , is clearly sub-optimal because it entails a large current error and also requires switching. Whether  $u(k) = 0$

or  $u(k) = 1$  minimizes the cost and is thus optimal depends on the choice for  $\rho$ . For large  $\rho$ , the controller refrains from frequent switching and tolerates relatively large current errors. For small  $\rho$ , the controller switches more often to minimize the deviations of the current from its reference.

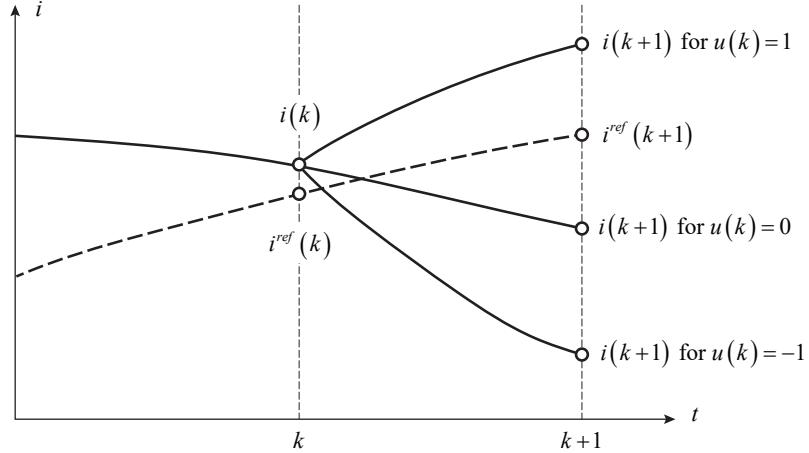


Figure 6.3: Prediction of the current at the next time step  $i(k + 1)$  as a function of the output voltage  $u(k)$ . The current reference  $i^{ref}$  is indicated by the dashed line.

$u(k)$	$(i^{ref}(k + 1) - i(k + 1))^2$	$\rho (\Delta u(k))^2$
1	0.044 <sup>2</sup>	$\rho$
0	0.051 <sup>2</sup>	0
-1	0.085 <sup>2</sup>	$\rho$

Table 6.2: Cost corresponding to the three admissible  $u(k)$ .

In the following simulation results are reported.

## 6.2. Theory of Model Predictive Control

---

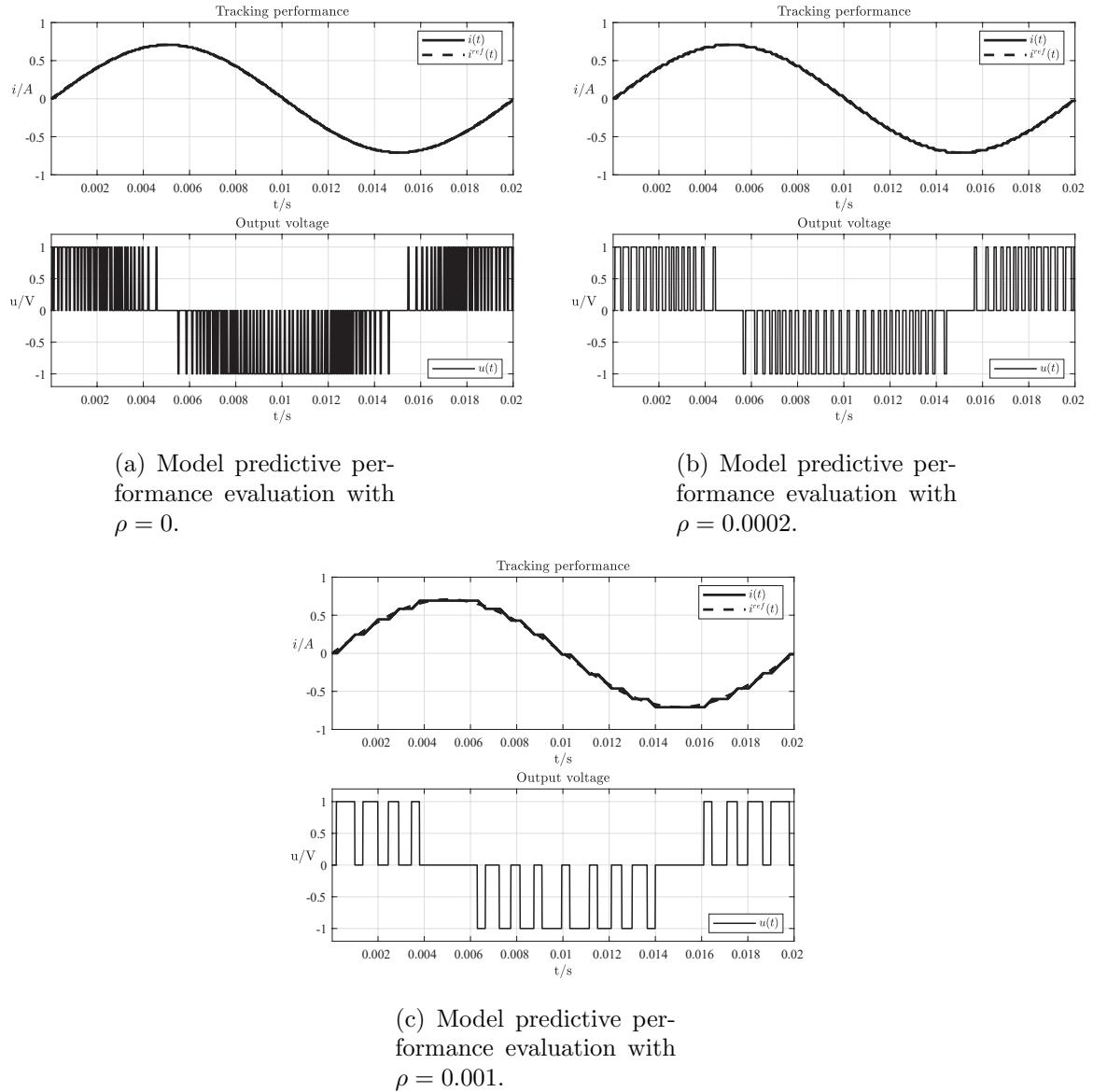


Figure 6.4: Simulation results.

## 6.2 Theory of Model Predictive Control

Given the cost function

$$J = \frac{1}{2} \sum_{k=1}^{N_p} \Delta \vec{y}^T(k) \mathbf{Q} \Delta \vec{y}(k) + \frac{1}{2} \sum_{k=1}^{N_c} \Delta \vec{u}^T(k) \mathbf{R} \Delta \vec{u}(k) \quad (6.2.1)$$

where  $N_p$  is the prediction horizon and  $N_c$  is the control horizon.

Let's try to write our linear system Eq. (7.2.9) in term of  $\Delta\vec{y}$  and  $\Delta\vec{u}$

$$\begin{aligned}\vec{x}(k+1) &= \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k) \\ \vec{y}(k) &= \mathbf{C}\vec{x}(k)\end{aligned}\tag{6.2.2}$$

Hence, the control vector

$$\vec{u}(k) = \vec{u}(k-1) + \Delta\vec{u}(k)\tag{6.2.3}$$

yields the following closed-loop state equation

$$\vec{x}(k+1) = \mathbf{A}\vec{x}(k) + \mathbf{B}\vec{u}(k-1) + \mathbf{B}\Delta\vec{u}(k)\tag{6.2.4}$$

with the outputs until the control horizon  $N_c$

$$\begin{aligned}\vec{y}(k) &= \mathbf{C}\vec{x}(k) \\ \vec{y}(k+1|k) &= \mathbf{C}\mathbf{A}\vec{x}(k) + \mathbf{C}\mathbf{B}\vec{u}(k-1) + \mathbf{C}\mathbf{B}\Delta\vec{u}(k) \\ \vec{y}(k+2|k) &= \mathbf{C}\mathbf{A}^2\vec{x}(k) + \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B}\vec{u}(k-1) + \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B}\Delta\vec{u}(k) + \mathbf{C}\mathbf{B}\Delta\vec{u}(k+1) \\ \vec{y}(k+3|k) &= \mathbf{C}\mathbf{A}^3\vec{x}(k) + \mathbf{C}(\mathbf{A}^2 + \mathbf{A} + \mathbf{I})\mathbf{B}\vec{u}(k-1) + \mathbf{C}(\mathbf{A}^2 + \mathbf{A} + \mathbf{I})\mathbf{B}\Delta\vec{u}(k) + \\ &\quad + \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B}\Delta\vec{u}(k+1) + \mathbf{C}\mathbf{B}\Delta\vec{u}(k+2) \\ &\vdots \\ \vec{y}(k+N_c|k) &= \mathbf{C}\mathbf{A}^{N_c}\vec{x}(k) + \mathbf{C}(\mathbf{A}^{N_c-1} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B}\vec{u}(k-1) + \\ &\quad + \mathbf{C}(\mathbf{A}^{N_c-1} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B}\Delta\vec{u}(k) + \mathbf{C}(\mathbf{A}^{N_c-2} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B}\Delta\vec{u}(k+1) + \dots \\ &\quad + \mathbf{C}\mathbf{B}\Delta\vec{u}(k+N_c-1) = \\ &= \mathbf{C}\mathbf{A}^{N_c}\vec{x}(k) + \mathbf{C}\left(\sum_{j=0}^{N_c-1} \mathbf{A}^j\right)\mathbf{B}\vec{u}(k-1) + \mathbf{C}\left(\sum_{j=0}^{N_c-1} \mathbf{A}^j\right)\mathbf{B}\Delta\vec{u}(k) + \\ &\quad + \mathbf{C}\left(\sum_{j=0}^{N_c-2} \mathbf{A}^j\right)\mathbf{B}\vec{u}(k+1) + \dots + \mathbf{C}\mathbf{B}\Delta\vec{u}(k+N_c-1) = \\ &= \mathbf{C}\mathbf{A}^{N_c}\vec{x}(k) + \mathbf{C}\left(\sum_{j=0}^{N_c-1} \mathbf{A}^j\right)\mathbf{B}\vec{u}(k-1) + \mathbf{C}\sum_{n=1}^{N_c}\left(\sum_{j=0}^{N_c-n} \mathbf{A}^j\right)\mathbf{B}\Delta\vec{u}(k+n-1)\end{aligned}\tag{6.2.5}$$

For any further time-step beyond the control horizon  $N_c$  until the prediction horizon  $N_p \geq N_c$  the input is assumed to be constant, hence  $\Delta\vec{u}(k+i-1) = \vec{0}$  for  $i \geq N_c$ .

Hence, we obtain

$$\begin{aligned}
 \vec{y}(k + N_c + 1|k) &= \mathbf{C} \mathbf{A}^{N_c+1} \vec{x}(k) + \mathbf{C} \left( \sum_{j=0}^{N_c} \mathbf{A}^j \right) \mathbf{B} \vec{u}(k-1) + \\
 &\quad + \mathbf{C} \sum_{n=0}^{N_c} \left( \sum_{j=0}^{N_c-n} \mathbf{A}^j \right) \mathbf{B} \Delta \vec{u}(k+n-1) \\
 &\quad \vdots \\
 \vec{y}(k + N_p|k) &= \mathbf{C} \mathbf{A}^{N_p} \vec{x}(k) + \mathbf{C} \left( \sum_{j=0}^{N_p-1} \mathbf{A}^j \right) \mathbf{B} \vec{u}(k-1) + \\
 &\quad + \mathbf{C} \sum_{n=1}^{N_c} \left( \sum_{j=0}^{N_p-n} \mathbf{A}^j \right) \mathbf{B} \Delta \vec{u}(k+n-1)
 \end{aligned} \tag{6.2.6}$$

Define vectors

$$\mathbf{Y} = \begin{bmatrix} \vec{y}(k+1|k) \\ \vec{y}(k+2|k) \\ \vec{y}(k+3|k) \\ \vdots \\ \vec{y}(k+N_p|k) \end{bmatrix}, \quad \Delta \mathbf{U} = \begin{bmatrix} \Delta \vec{u}(k) \\ \Delta \vec{u}(k+1) \\ \Delta \vec{u}(k+2) \\ \vdots \\ \Delta \vec{u}(k+N_c-1) \end{bmatrix} \tag{6.2.7}$$

where in SISO case, the dimension of  $\mathbf{Y}$  is  $N_p$  and the dimension of  $\Delta \mathbf{U}$  is  $N_c$ .

We collect Eq. (7.2.12) and Eq. (7.2.13) together in a compact matrix form as

$$\mathbf{Y} = \mathbf{F} \vec{x}(k) + \mathbf{G} \vec{u}(k-1) + \mathbf{H} \Delta \mathbf{U} \tag{6.2.8}$$

where

$$\mathbf{F} = \begin{bmatrix} \mathbf{C} \mathbf{A} \\ \mathbf{C} \mathbf{A}^2 \\ \mathbf{C} \mathbf{A}^3 \\ \vdots \\ \mathbf{C} \mathbf{A}^{N_p} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{C} \mathbf{B} \\ \mathbf{C} (\mathbf{A} + \mathbf{I}) \mathbf{B} \\ \mathbf{C} (\mathbf{A}^2 + \mathbf{A} + \mathbf{I}) \mathbf{B} \\ \vdots \\ \mathbf{C} (\mathbf{A}^{N_p-1} + \dots + \mathbf{A} + \mathbf{I}) \mathbf{B} \end{bmatrix} \tag{6.2.9}$$

and

$$\mathbf{H} = \begin{bmatrix} \mathbf{CB} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{CB} & \dots & \mathbf{0} \\ \mathbf{C}(\mathbf{A}^2 + \mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}(\mathbf{A}^{N_c-1} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{C}(\mathbf{A}^{N_c-2} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} & \dots & \mathbf{CB} \\ \mathbf{C}(\mathbf{A}^{N_c} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{C}(\mathbf{A}^{N_c-1} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} & \dots & \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}(\mathbf{A}^{N_p-1} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{C}(\mathbf{A}^{N_p-2} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} & \dots & \mathbf{C}(\mathbf{A}^{N_p-N_c} + \dots + \mathbf{A} + \mathbf{I})\mathbf{B} \end{bmatrix} \quad (6.2.10)$$

We can define the error between reference and output as follows

$$\Delta \mathbf{Y} = \mathbf{Y} - \mathbf{Y}_{\text{ref}} \quad (6.2.11)$$

The cost function of Eq. (7.2.8) can be represented as follows

$$J = \frac{1}{2} \Delta \mathbf{Y}^T \mathbf{Q} \Delta \mathbf{Y} + \frac{1}{2} \Delta \mathbf{U}^T \mathbf{R} \Delta \mathbf{U} \quad (6.2.12)$$

We can express the term  $\Delta \mathbf{Y}$  as  $\Delta \mathbf{Y} = \mathbf{F}\vec{x}(k) + \mathbf{G}\vec{u}(k-1) + \mathbf{H}\Delta \mathbf{U} - \mathbf{Y}_{\text{ref}}$  and we obtain

$$\begin{aligned} J &= \frac{1}{2} \left( \mathbf{F}\vec{x}(k) + \mathbf{G}\vec{u}(k-1) + \mathbf{H}\Delta \mathbf{U} - \mathbf{Y}_{\text{ref}} \right)^T \mathbf{Q} \left( \mathbf{F}\vec{x}(k) + \mathbf{G}\vec{u}(k-1) + \mathbf{H}\Delta \mathbf{U} - \mathbf{Y}_{\text{ref}} \right) \\ &\quad + \frac{1}{2} \Delta \mathbf{U}^T \mathbf{R} \Delta \mathbf{U} \end{aligned} \quad (6.2.13)$$

By proper manipulation\* we can obtain the following expression

$$\begin{aligned} J &= \frac{1}{2} \left( \mathbf{F}\vec{x}(k) + \mathbf{G}\vec{u}(k-1) - \mathbf{Y}_{\text{ref}} \right)^T \mathbf{Q} \left( \mathbf{F}\vec{x}(k) + \mathbf{G}\vec{u}(k-1) - \mathbf{Y}_{\text{ref}} \right) \\ &\quad + \frac{1}{2} \left( 2\vec{x}^T(k)\mathbf{F}^T \mathbf{Q} \mathbf{H} \Delta \mathbf{U} + 2\vec{u}^T(k-1)\mathbf{G}^T \mathbf{Q} \mathbf{H} \Delta \mathbf{U} + \Delta \mathbf{U}^T \mathbf{H}^T \mathbf{Q} \mathbf{H} \Delta \mathbf{U} + \right. \\ &\quad \left. - 2\Delta \mathbf{U}^T \mathbf{H}^T \mathbf{Q} \mathbf{Y}_{\text{ref}} \right) + \frac{1}{2} \Delta \mathbf{U}^T \mathbf{R} \Delta \mathbf{U} \end{aligned} \quad (6.2.14)$$

Applying the  $\frac{\partial J}{\partial \Delta \mathbf{U}} = \mathbf{0}$  we obtain

$$\begin{aligned} \frac{\partial J}{\partial \Delta \mathbf{U}} &= \mathbf{H}^T \mathbf{Q} \mathbf{F} \vec{x}(k) + \mathbf{H}^T \mathbf{Q} \mathbf{G} \vec{u}(k-1) - \mathbf{H}^T \mathbf{Q} \mathbf{Y}_{\text{ref}} + [\mathbf{H}^T \mathbf{Q} \mathbf{H} + \mathbf{R}] \Delta \mathbf{U} = \\ &= \mathbf{H}^T \mathbf{Q} [\mathbf{F} \vec{x}(k) + \mathbf{G} \vec{u}(k-1) - \mathbf{Y}_{\text{ref}}] + [\mathbf{H}^T \mathbf{Q} \mathbf{H} + \mathbf{R}] \Delta \mathbf{U} = \mathbf{0} \end{aligned} \quad (6.2.15)$$

\*  $(a+b+c-d)(a+b+c-d) = (a+b-d)(a+b-d) + 2ac + c^2 - 2cd + 2bc$

which yields to

$$\Delta \mathbf{U} = -[\mathbf{H}^T \mathbf{Q} \mathbf{H} + \mathbf{R}]^{-1} \mathbf{H}^T \mathbf{Q} [\mathbf{F} \vec{x}(k) + \mathbf{G} \vec{u}(k-1) - \mathbf{Y}_{\text{ref}}] \quad (6.2.16)$$

We can define

$$\mathbf{E} = \mathbf{F} \vec{x}(k) + \mathbf{G} \vec{u}(k-1) - \mathbf{Y}_{\text{ref}} \quad (6.2.17)$$

and obtain the following control output

$$\Delta \mathbf{U} = -\mathbf{K} \mathbf{E} = -[\mathbf{H}^T \mathbf{Q} \mathbf{H} + \mathbf{R}]^{-1} \mathbf{H}^T \mathbf{Q} \mathbf{E} \quad (6.2.18)$$

but we are interested in  $\Delta \vec{u}(k)$  which is the first element of  $\Delta \mathbf{U}$  and the final control structure is as follows

$$\begin{aligned} \Delta \vec{u}(k) &= -[\mathbf{I}_m \ 0 \ \dots \ 0] \mathbf{K} \mathbf{E} = \\ &= -[\mathbf{I}_m \ 0 \ \dots \ 0] [\mathbf{H}^T \mathbf{Q} \mathbf{H} + \mathbf{R}]^{-1} \mathbf{H}^T \mathbf{Q} [\mathbf{F} \vec{x}(k) + \mathbf{G} \vec{u}(k-1) - \mathbf{Y}_{\text{ref}}] \end{aligned} \quad (6.2.19)$$

Where in case of SISO  $\mathbf{I}_m = 1$ .

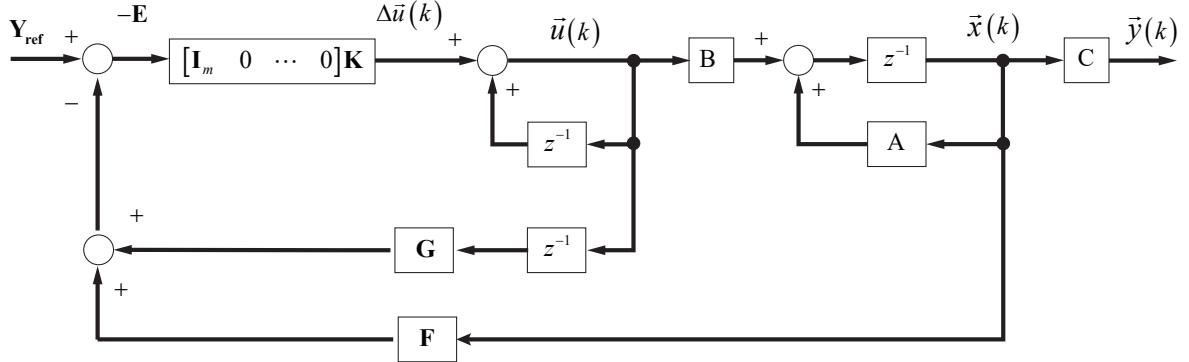


Figure 6.5: Block diagram for MPC of a linear system (without observer).

# Chapter 7

## Adaptive Control

This lecture covers the topic regarding the adaptive control. In particular we will focus on *Model Reference Adaptive Control* in which a reference model is taken as benchmark and a defined set of parameters are adapted in order to keep the answer of the system close to the reference one.

### 7.1 Model Reference Adaptive Control.

Model reference adaptive control (MRAC) is one of the main approaches to adaptive control. The basic structure of a MRAC scheme is shown in Figure 7.1. The reference model is chosen to generate the desired trajectory,  $\vec{y}_m(t)$ , that the plant output  $\vec{y}_p(t)$  has to follow. The trajectory error  $\vec{e}(t) = \vec{y}_p(t) - \vec{y}_m(t)$  represents the deviation of the plant output from the desired trajectory. The controller of the plant is composed by an ordinary controller the a set of parameters  $\vec{\vartheta}(t)$  is adjusted by an proper adaptation mechanism

$$\dot{\vec{\vartheta}}(t) = C(\hat{\vec{\vartheta}}, \vec{\vartheta}, \vec{e}, \vec{y}_p, \vec{u}_p)$$

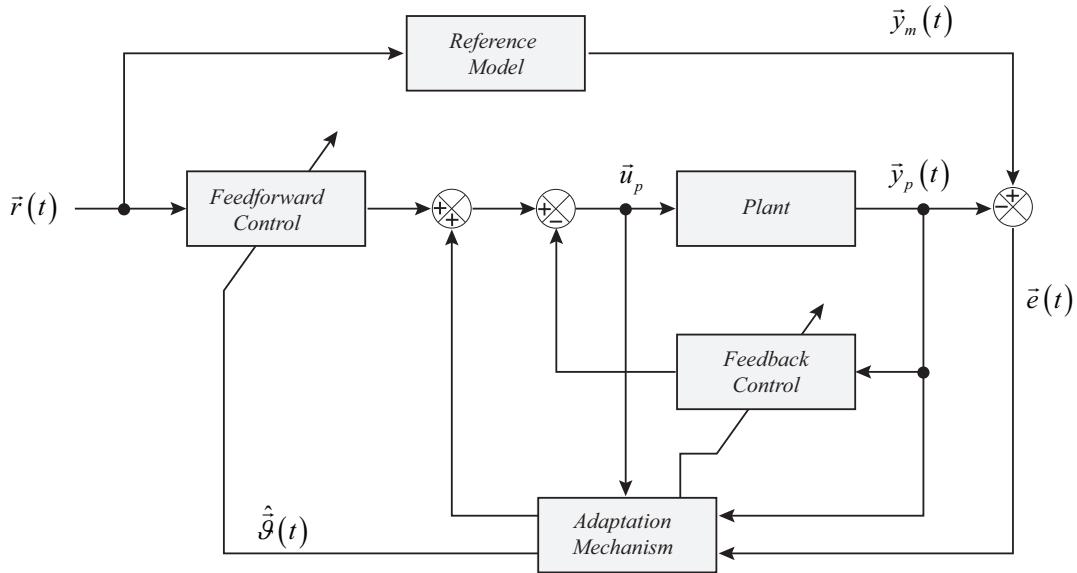


Figure 7.1: Block diagram of a model-reference adaptive control (MRAC).

MRAC schemes can be characterized as *direct* or *indirect* adaptive laws. In direct MRAC, the parameters vector  $\hat{\vartheta}$  of the controller  $C(\hat{\vartheta}, \vec{\vartheta}, \vec{e}, \vec{y}_p, \vec{u}_p)$  is updated directly by an adaptive law, whereas in indirect MRAC  $\hat{\vartheta}$  is calculated at each time  $t$  by the previous on-line estimation of the plant parameters.

## 7.2 Simple Adaptive Control Systems

In this section some simple examples are used to illustrate basic concepts and design steps for adaptive control systems. Only one class of adaptive control system will be shown: that based on *direct* adaptive control.

### 7.2.1 Direct Adaptive Control

In direct adaptive control systems the controller parameters are *directly* updated from an adaptive law.

#### 7.2.1.1 Lyapunov Design

**Example 7.1.** Let us first consider a first-order linear time-invariant plant

$$\dot{y}(t) = a_p y(t) + u(t), \quad t \geq 0 \quad (7.2.1)$$

where the constant  $a_p$  is the plant parameter,  $y(t)$  is the plant output with initial value  $y(0) = y_0$  and  $u(t)$  is the control input.

The control objective is to design a feedback control  $u(t)$  such that all closed-loop system signals are bounded\* and the plant output  $y(t)$  tracks, asymptotically, the output  $y_m(t)$  of the chosen reference model.

$$\dot{y}_m(t) = -a_m y_m(t) + r(t), \quad t \geq 0 \quad (7.2.2)$$

with  $y_m(0) = y_{m0}$ , where  $a_m > 0$  for stability and performance and  $r(t)$  is a bounded external input which characterizes desired response.

**Consider  $a_p$  known.** - When the plant parameter  $a_p$  is known, we use the feedback controller

$$u(t) = -k^* y(t) + r(t) \quad (7.2.3)$$

where

$$k^* = a_p + a_m. \quad (7.2.4)$$

With this controller, the closed-loop system is

$$\dot{y}(t) = -a_m y(t) + r(t). \quad (7.2.5)$$

Define the tracking error as  $e(t) = y(t) - y_m(t)$ . Then from Eq. 7.2.2 and Eq. 7.2.5, we have the tracking error equation

$$\dot{e}(t) = -a_m e(t) \quad (7.2.6)$$

with  $e(0) = y(0) - y_m(0)$ . The solution to this equation is  $e(t) = e^{-a_m t} e(0)$ ,  $t \geq 0$ , which has the desired property:  $e(t)$  is bounded, and so are  $y(t)$  and  $u(t)$ . Moreover,  $\lim_{t \rightarrow \infty} e(t) = 0$ . Hence, we have achieved the control objective.

**Consider  $a_p$  unknown.** - When plant parameter  $a_p$  is unknown, we cannot implement the control law Eq. 7.2.3 because  $k^*$  is unknown. Instead, we use an estimate  $k(t)$  for  $k^*$  to implement the adaptive controller,

$$u(t) = -k(t)y(t) + r(t) \quad (7.2.7)$$

In view of Eq. 7.2.3, this controller, when applied to the plant Eq. 7.2.1, results in the closed loop system

$$\dot{y}(t) = -a_m y(t) + r(t) + (k(t) - k^*)y(t), \quad t \geq 0. \quad (7.2.8)$$

In terms of the tracking error  $e(t)$ , we have

$$\dot{e}(t) = -a_m e(t) + \tilde{k}(t)y(t), \quad t \geq 0 \quad (7.2.9)$$

where  $\tilde{k}(t) = k(t) - k^*$  is the parameter error.

The design task is to choose an adaptive law to update the estimate  $k(t)$  (i.e. to specify

---

\* A signal  $x(t) \in \mathbb{R}$  is bounded if  $|x(t)| \leq c$ ,  $\forall t \geq 0$ , for some constant  $c > 0$

$\dot{k}(t)$ , the time derivative of  $k(t)$  so that the stated control objective is still achievable even if the plant parameter  $a_p$  is unknown.

Let us introduce a measure for the errors  $e(t)$  and  $\tilde{k}(t)$ :

$$V(e, \tilde{k}) = e^2 + \tilde{k}^2, \quad (7.2.10)$$

which is positive whenever  $e \neq 0$  and/or  $\tilde{k} \neq 0$ , and examine the time derivative  $\dot{V} = \frac{dV}{dt}$ .

The time derivative of  $V(e, \tilde{k})$  is

$$\begin{aligned} \dot{V}(e, \tilde{k}) &= \frac{\partial V(e, \tilde{k})}{\partial e} \dot{e}(t) + \frac{\partial V(e, \tilde{k})}{\partial \tilde{k}} \dot{\tilde{k}}(t) \\ &= 2(e(t)\dot{e}(t) + \tilde{k}\dot{\tilde{k}}). \end{aligned} \quad (7.2.11)$$

With Eq. 7.2.9 and the fact that  $\dot{\tilde{k}}(t) = \dot{k}(t) - \dot{k}^* = \dot{k}(t)$  because  $k^*$  is constant, Eq. 7.2.11 becomes

$$\dot{V}(e, \tilde{k}) = -2a_m e^2(t) + 2e(t)y(t)\tilde{k} + 2\tilde{k}(t)\dot{k}(t). \quad (7.2.12)$$

If an adaptive law  $\dot{k}(t)$  ensures  $\dot{V} \leq 0, \forall e(t), \tilde{k}(t)$ , the errors  $e(t), \tilde{k}(t)$  will stay inside a circle centered at the origin. If, in addition,  $\dot{V} < 0$  for any  $e(t) \neq 0$ , the tracking error  $e(t)$  may be forced to go to zero asymptotically. To make  $\dot{V} \leq 0$ , we choose the following adaptive law for  $k(t)$ :

$$\dot{k}(t) = -e(t)y(t), \quad t \geq 0, \quad (7.2.13)$$

which leads Eq. 7.2.11 to

$$\dot{V} = -2a_m e^2(t) \leq 0. \quad (7.2.14)$$

The implementation of the adaptive law Eq. 7.2.13 is

$$k(t) = - \int_0^t e(\tau)y(\tau)d\tau + k(0) \quad (7.2.15)$$

where  $k(0)$  is an initial estimate of the unknown parameter  $k^*$ .

▫

**Example 7.2.** Consider the control of a mass on a frictionless surface by a motor force  $u$ , with the plant dynamics being

$$m\ddot{x}(t) = u(t) \quad (7.2.16)$$

Assume that a human operator provides the positioning command  $r(t)$  to the control system (possibly through a joystick). A reasonable way of specifying the ideal response of the controlled mass to the external command  $r(t)$  is to use the following reference model

$$\ddot{x}_m(t) + \lambda_1 \dot{x}_m(t) + \lambda_2 x_m(t) = \lambda_2 r(t) \quad (7.2.17)$$

with the positive constants  $\lambda_1$  and  $\lambda_2$  chosen to reflect the performance specifications, and the

reference model output  $x_m(t)$  being the ideal output of the control system (i.e., ideally, the mass should go to the specified position  $r(t)$  like a well-damped mass-spring-damper system).

if the mass  $m$  is known exactly, we can use the following control law to achieve perfect tracking

$$u(t) = m \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] \quad (7.2.18)$$

with  $\tilde{x}(t) = x(t) - x_m(t)$  representing the tracking error and  $\lambda$  a strictly positive number. This control law leads to the exponentially convergent tracking error dynamics

$$\ddot{\tilde{x}}(t) + 2\lambda \dot{\tilde{x}}(t) + \lambda^2 \tilde{x}(t) = 0 \quad (7.2.19)$$

Now let us assume that the mass **is not known exactly**. We may use the following control law

$$u(t) = \hat{m} \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] \quad (7.2.20)$$

which contains the adjustable parameter  $\hat{m}$ . Considering  $\tilde{m} = \hat{m} - m$  Eq. 7.2.20 becomes

$$u(t) = \tilde{m} \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] + m \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] \quad (7.2.21)$$

substituting Eq. 7.2.21 into Eq. 7.2.16 we obtain

$$m \ddot{x}(t) = \tilde{m} \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] + m \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] \quad (7.2.22)$$

or

$$m \ddot{\tilde{x}}(t) + 2m\lambda \dot{\tilde{x}}(t) + m\lambda^2 \tilde{x}(t) = \tilde{m} \left[ \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \right] \quad (7.2.23)$$

Substituting the quantity

$$\begin{aligned} s(t) &= \dot{\tilde{x}}(t) + \lambda \tilde{x}(t) \\ v(t) &= \ddot{x}_m(t) - 2\lambda \dot{\tilde{x}}(t) - \lambda^2 \tilde{x}(t) \end{aligned} \quad (7.2.24)$$

into Eq. 7.2.23 we obtain

$$\begin{cases} s(t) = \dot{\tilde{x}}(t) + \lambda \tilde{x}(t) \\ m \dot{s}(t) + \lambda m s(t) = \tilde{m}(t) v(t) \end{cases} \quad (7.2.25)$$

Considering the following adaptive law

$$\dot{\tilde{m}}(t) = -\gamma v(t) s(t) \quad (7.2.26)$$

and considering that  $\dot{\tilde{m}} = \dot{\tilde{m}}$ , we obtain the following **closed loop system**

**Closed loop system**

$$\begin{cases} s(t) = \dot{\tilde{x}}(t) + \lambda\tilde{x}(t) \\ v(t) = \ddot{x}_m(t) - 2\lambda\dot{\tilde{x}} - \lambda^2\tilde{x} \\ m\dot{s}(t) + \lambda ms(t) = \tilde{m}(t)v(t) \\ \dot{\tilde{m}} = \dot{m} = -\gamma v(t)s(t) \end{cases} \quad (7.2.27)$$

From Eq. 7.2.20 and substituting with Eq. 7.2.24 and Eq. 7.2.26 we obtain the following control law

**Control law**

$$\begin{cases} s(t) = \dot{\tilde{x}}(t) + \lambda\tilde{x}(t) \\ v(t) = \ddot{x}_m(t) - 2\lambda\dot{\tilde{x}} - \lambda^2\tilde{x} \\ \dot{\tilde{m}}(t) = -\gamma v(t)s(t) \\ u(t) = \hat{m}(t)v(t) \end{cases} \quad (7.2.28)$$

Resulting in the control architecture shown in Figure 7.2. The stability and convergence of this adaptive law can be analysed using Lyapunov theory. Considering the closed loop system reported in Eq. 7.2.27 where  $s(t)$  and  $\tilde{m}(t)$  represent the state variables we can build the following candidate Lyapunov function

$$V(s, \tilde{m}) = \frac{1}{2} \left[ ms^2(t) + \frac{1}{\gamma} \tilde{m}^2(t) \right] \quad (7.2.29)$$

its derivative is

$$\dot{V}(s, \tilde{m}) = -\lambda ms^2(t) \quad (7.2.30)$$

That means the system is stable. Now we try to explore the behaviour of the quantity  $s(t)$  and its implication. The convergence to zero of the quantity  $s(t)$  can be assumed by the following lemma which derive from the set of corollaries of **Barbalat's Lemma**:

**Lemma 7.1.** *If a scalar function  $V(\vec{x}, t)$  satisfies the following conditions*

- $V(\vec{x}, t)$  is lower bounded
- $\dot{V}(\vec{x}, t)$  is negative semi-definite
- $\dot{V}(\vec{x}, t)$  is uniformly continuous in time

then  $\dot{V}(\vec{x}, t) \rightarrow 0$  as  $t \rightarrow \infty$

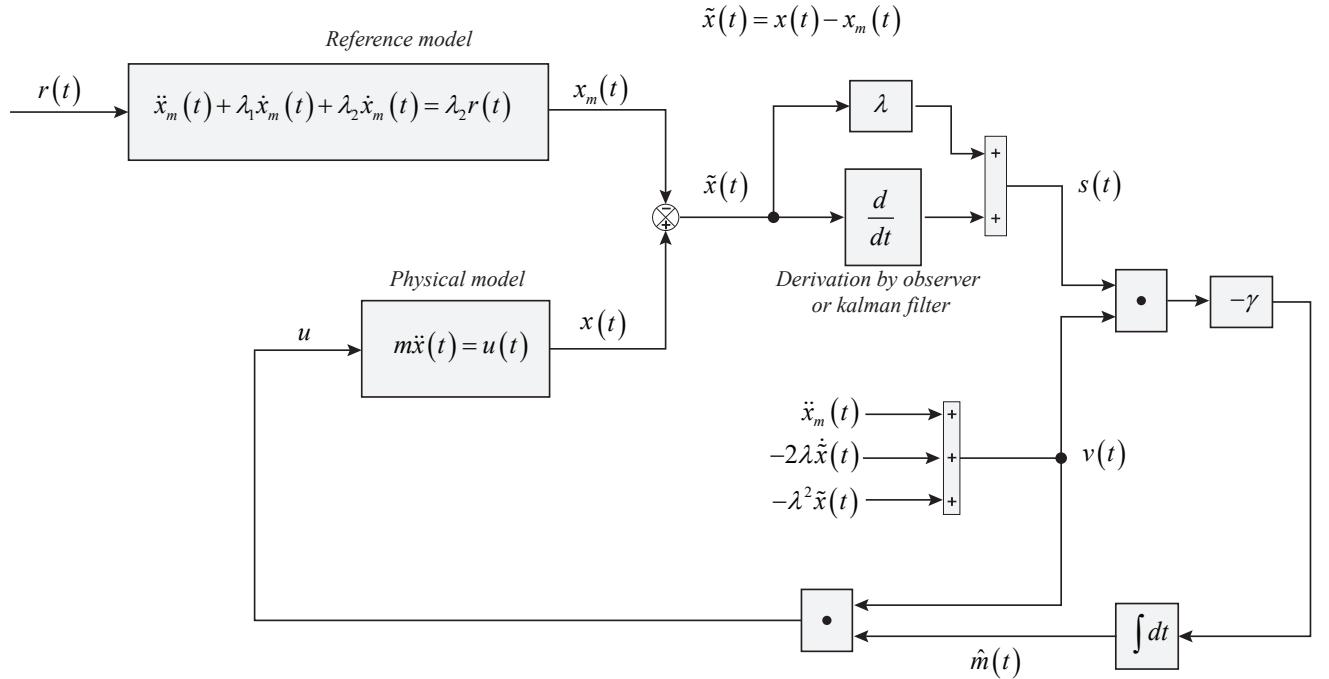


Figure 7.2: Control architecture.

That means  $s(t) \rightarrow 0$  as  $t \rightarrow \infty$  and that means, from Eq. 7.2.28, also the quantities  $\dot{\tilde{x}}(t)$  and  $\tilde{x}(t)$  converge to zero. As we will see later, the convergence to zero of the quantity  $s(t)$  means the system  $x(t)$  will behave as the system  $x_m(t)$  but none information about the convergence of the mass estimation  $\hat{m}$  can be derived.

A case study is here reported. Consider the following data:

- $\lambda_1 = 10$ ,  $\lambda_2 = 25$
- $m = 2$
- $\lambda = 6$ ,  $\gamma = 0.5$

As previously mentioned Figure 7.3a and Figure 7.3b show an asymptotic behaviour in term of tracking and mass estimation, while Figure 7.4a and Figure 7.4b show an asymptotic bias-free behaviour only for the tracking performance and the mass estimation remains with a steady state error. The concept behind this behaviour is covered by the persistency of excitation where only in the first case is achieved. In the theory of the parameters identification (*identifiability*) the concept of persistent excitation is a necessary condition for the asymptotically bias-free estimation.

The control architecture includes a derivative block which can be implemented by an observer or by a Kalman filter (robust against noise).

Consider the plant

$$\ddot{x}(t) = 0 \quad (7.2.31)$$

or

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \quad (7.2.32)$$

$$y(t) = x_1(t)$$

The observer can be written as follows

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 + l_1(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_2 &= l_2(x_1 - \hat{x}_1) \end{aligned} \quad (7.2.33)$$

where  $\hat{x}_2(t)$  represents the derivative of the quantity  $x_1(t)$ .

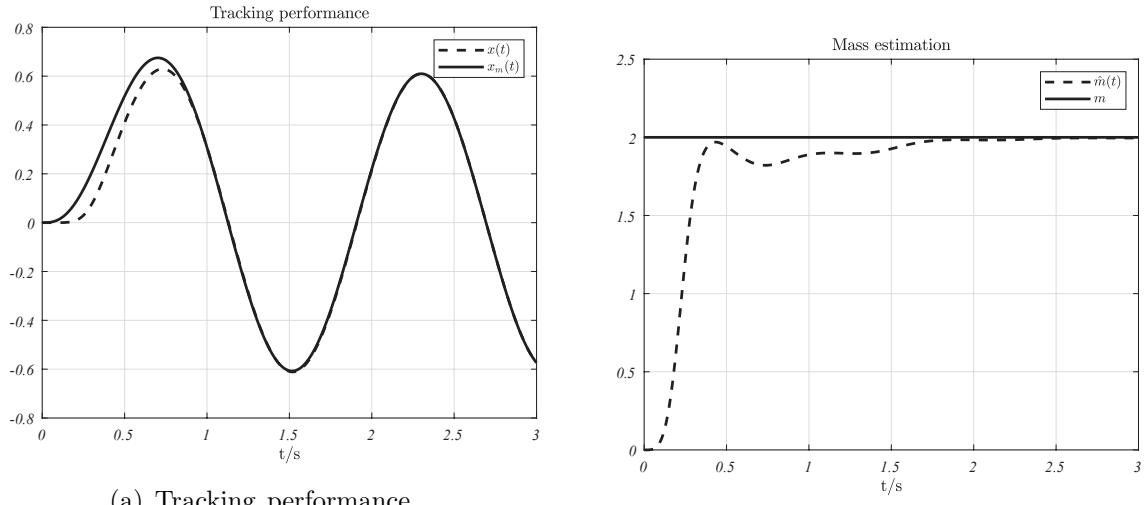


Figure 7.3: Simulation results.

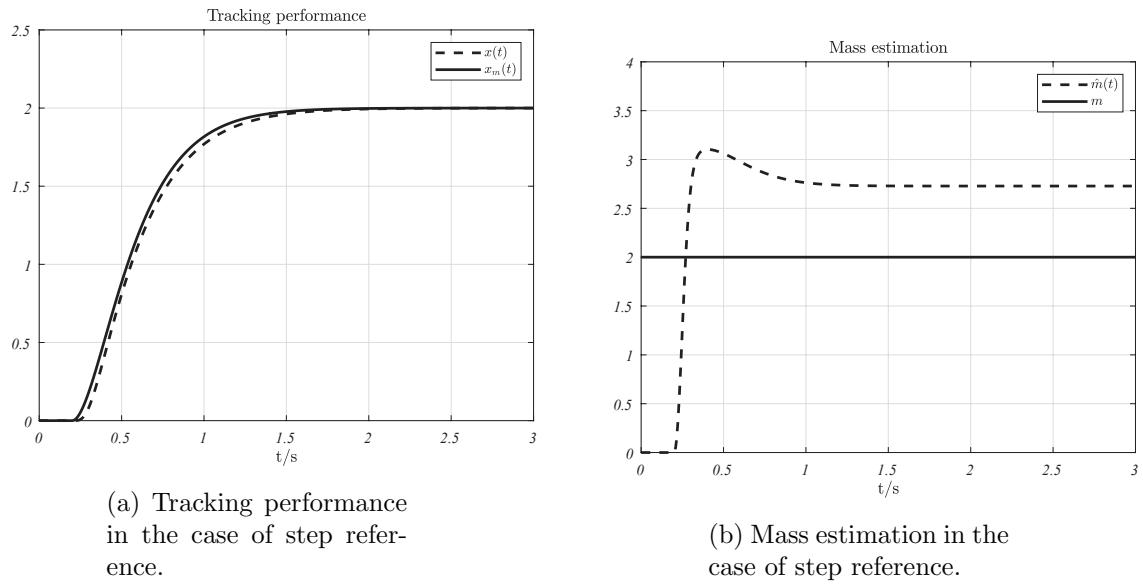


Figure 7.4: Simulation results.

△

### 7.3 Adaptive Control for Active Suspension System

The control strategy which will be pursued can be schematized as follows

- The control design is based on *model reference adaptive control* MRAC, where sky-hook damping model is assumed (see Figure 7.5).
- During working life of the suspension, parameters like vehicle mass, damping coefficient can vary considerably.
- The aim of the control is to develop an adaptive law which makes the system to behave as all parameters were known. We don't estimate parameters, but we want our system works as we know.

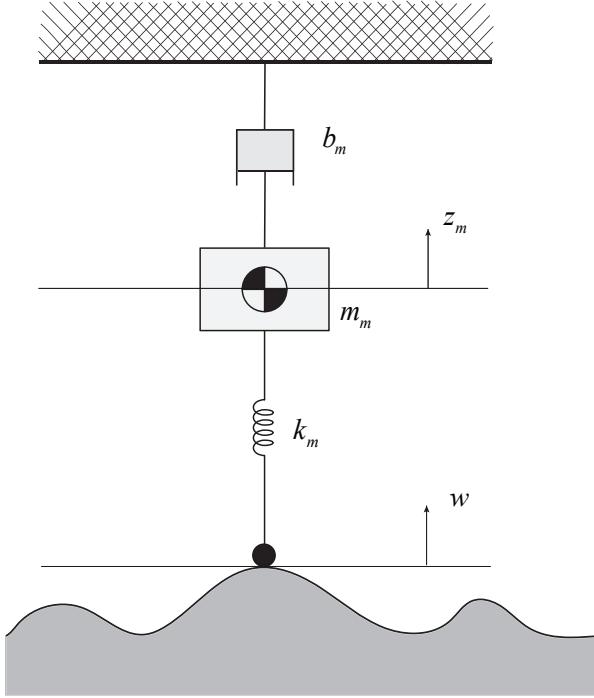


Figure 7.5: Sky-hook model reference.

The sky-hook can be represented as follows

$$m_m \ddot{z}_m = -k_m(z_m - w) - b_m \dot{z}_m \quad (7.3.1)$$

where  $w(t)$  represents the profile of the floor height. In the following control design the knowledge of the quantities  $\ddot{w}(t)$ ,  $\dot{w}(t)$  are necessary and that can be achieved by the use of an accelerometer along the vertical axis. The derivation of the quantity  $\dot{w}(t)$  from  $\ddot{w}(t)$  can be done using a Kalman filter.

Set

$$\begin{aligned} x_m &= z_m - w \Rightarrow z_m = x_m + w \\ \dot{x}_m &= \dot{z}_m - \dot{w} \Rightarrow \dot{z}_m = \dot{x}_m + \dot{w} \\ \ddot{x}_m &= \ddot{z}_m - \ddot{w} \Rightarrow \ddot{z}_m = \ddot{x}_m + \ddot{w} \\ m_m \ddot{x}_m + m_m \ddot{w} &= -k_m x_m - b_m \dot{x}_m - b_m \dot{w} \end{aligned} \quad (7.3.2)$$

setting the following vector state

$$\vec{x}_m = \begin{bmatrix} x_m^m & x_2^m \end{bmatrix}^T \quad (7.3.3)$$

Eq. 7.3.2 results as follows

$$\dot{\vec{x}}_m = \begin{bmatrix} 0 & 1 \\ -\frac{k_m}{m_m} & -\frac{b_m}{m_m} \end{bmatrix} \vec{x}_m + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \ddot{w} + \begin{bmatrix} 0 \\ -\frac{b_m}{m_m} \end{bmatrix} \dot{w} \quad (7.3.4)$$

or

$$\dot{\vec{x}}_m = \mathbf{A}_m \vec{x}_m + \mathbf{B}_m \dot{w} + \mathbf{C}_m \ddot{w} \quad (7.3.5)$$

where

$$\begin{aligned} \mathbf{A}_m &= \begin{bmatrix} 0 & 1 \\ -\frac{k_m}{m_m} & -\frac{b_m}{m_m} \end{bmatrix} & \mathbf{B}_m &= \begin{bmatrix} 0 \\ -\frac{b_m}{m_m} \end{bmatrix} \\ \mathbf{C}_m &= \begin{bmatrix} 0 \\ -1 \end{bmatrix} \end{aligned} \quad (7.3.6)$$

**Eq. 7.3.5 and Eq. 7.3.6 represent the sky-hook reference model.**

Consider now the plant model which can be depicted as in Figure 7.6.

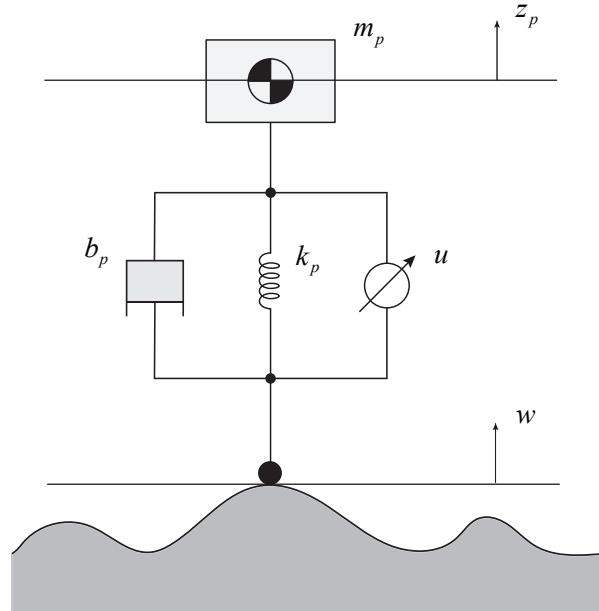


Figure 7.6: Active suspension model.

**The quarter suspension model can be represented as follows**

$$m_p \ddot{z}_p = -k_p(z_p - w) - b_p(\dot{z}_p - \dot{w}) + u \quad (7.3.7)$$

set the following change of coordinates

$$x_p = z_p - w \Rightarrow z_p = x_p + w$$

$$\dot{x}_p = \dot{z}_p - \dot{w} \Rightarrow \dot{z}_p = \dot{x}_p + \dot{w}$$

$$\ddot{x}_p = \ddot{z}_p - \ddot{w} \Rightarrow \ddot{z}_p = \ddot{x}_p + \ddot{w}$$

$$m_p \ddot{x}_p + m_p \ddot{w} = -k_p x_p - b_p \dot{x}_p + u \quad (7.3.8)$$

setting the following vector state

$$\vec{x}_p = \begin{bmatrix} x_1^p & x_2^p \end{bmatrix}^T \quad (7.3.9)$$

Eq. 7.3.8 results as follows

$$\begin{aligned} \dot{\vec{x}}_p &= \begin{bmatrix} 0 & 1 \\ -\frac{k_p}{m_p} & -\frac{b_p}{m_p} \end{bmatrix} \vec{x}_p + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \ddot{w} \\ &\quad + \begin{bmatrix} 0 \\ \frac{1}{m_p} \end{bmatrix} u \end{aligned} \quad (7.3.10)$$

or

$$\dot{\vec{x}}_p = \mathbf{A}_p \vec{x}_p + \mathbf{B}_p u + \mathbf{C}_p \ddot{w} \quad (7.3.11)$$

where

$$\begin{aligned} \mathbf{A}_p &= \begin{bmatrix} 0 & 1 \\ -\frac{k_p}{m_p} & -\frac{b_p}{m_p} \end{bmatrix} & \mathbf{B}_p &= \begin{bmatrix} 0 \\ \frac{1}{m_p} \end{bmatrix} \\ \mathbf{C}_p &= \mathbf{C}_m = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \end{aligned} \quad (7.3.12)$$

**Eq. 7.3.11 and Eq. 7.3.12 represent the quarter suspension model in new coordinates.**

### 7.3.1 Adaptive Control Design

Consider the following control law

$$u = k_r (\mathbf{K}_p \vec{x}_p + \dot{w}) \quad (7.3.13)$$

where

$$\mathbf{K}_p = [k_{p1} \ k_{p2}] . \quad (7.3.14)$$

Substituting Eq. 7.3.13 into Eq. 7.3.11 we obtain

$$\dot{\vec{x}}_p = \left( \mathbf{A}_p + k_r \mathbf{B}_p \mathbf{K}_p \right) \vec{x}_p + k_r \mathbf{B}_p \dot{w} + \mathbf{C}_p \ddot{w} \quad (7.3.15)$$

comparing Eq. 7.3.15 with Eq. 7.3.5 we find that there exists a set of unknown **constant** gains  $k_r^*$  and  $\mathbf{K}_p^*$  such that

$$\begin{cases} \mathbf{A}_m = \mathbf{A}_p + k_r^* \mathbf{B}_p \mathbf{K}_p^* \\ \mathbf{B}_m = k_r^* \mathbf{B}_p \end{cases} \quad (7.3.16)$$

Define the vector

$$\vec{e} = \vec{x}_p - \vec{x}_m \quad (7.3.17)$$

the vector  $\dot{\vec{e}}$  can be written as follows (using Eq. 7.3.16 and  $\dot{w} = \frac{1}{k_r} u - \mathbf{K}_p \vec{x}_p$ )

$$\begin{aligned} \dot{\vec{e}} &= \mathbf{A}_m \vec{e} + (\mathbf{A}_p - \mathbf{A}_m) \vec{x}_p + \mathbf{B}_p u - \mathbf{B}_m \dot{w} \\ &= \mathbf{A}_m \vec{e} - \mathbf{B}_p k_r^* \mathbf{K}_p^* \vec{x}_p + \mathbf{B}_p u - \frac{1}{k_r} \mathbf{B}_m u + \mathbf{B}_m \mathbf{K}_p \vec{x}_p \end{aligned} \quad (7.3.18)$$

considering

$$\mathbf{B}_p - \frac{1}{k_r} \mathbf{B}_m = \mathbf{B}_m \left( \frac{1}{k_r^*} - \frac{1}{k_r} \right) = \mathbf{B}_m \psi \quad (7.3.19)$$

$$\mathbf{B}_m = k_r^* \mathbf{B}_p \Rightarrow \mathbf{B}_p = \frac{1}{k_r^*} \mathbf{B}_m \quad (7.3.20)$$

$$\Phi = \mathbf{K}_p - \mathbf{K}_p^* \Rightarrow \dot{\Phi} = \dot{\mathbf{K}}_p \quad (7.3.21)$$

$$\psi = \frac{1}{k_r^*} - \frac{1}{k_r} \Rightarrow \dot{\psi} = \frac{1}{k_r^2} \dot{k}_r \quad (7.3.22)$$

Eq. 7.3.18 can be written as follows

$$\begin{aligned} \dot{\vec{e}} &= \mathbf{A}_m \vec{e} - \mathbf{B}_m \mathbf{K}_p^* \vec{x}_p + \psi \mathbf{B}_m u + \mathbf{B}_m \mathbf{K}_p \vec{x}_p \\ &= \mathbf{A}_m \vec{e} + \mathbf{B}_m \Phi \vec{x}_p + \psi \mathbf{B}_m u \end{aligned} \quad (7.3.23)$$

Let a candidate for a Lyapunov function be defined as

$$V = \vec{e}^T \mathbf{P} \vec{e} + (\Phi \mathbf{M} \Phi^T + \psi^2 n) \quad (7.3.24)$$

In order to calculate the quantity  $\dot{V}$ , consider that

$$\begin{aligned} \frac{d}{dt} \vec{e}^T \mathbf{P} \vec{e} &= \vec{e}^T \mathbf{P} \dot{\vec{e}} + \dot{\vec{e}}^T \mathbf{P} \vec{e} \\ \frac{d}{dt} (\Phi \mathbf{M} \Phi^T + \psi^2 n) &= 2(\Phi \mathbf{M} \dot{\Phi}^T + \psi \dot{\psi} n) \\ \dot{\vec{e}} &= \mathbf{A}_m \vec{e} + \mathbf{B}_m \Phi \vec{x}_p + \psi \mathbf{B}_m u \end{aligned}$$

hence

$$\begin{aligned}
 \dot{V} &= \vec{e}^T \mathbf{P} \mathbf{A}_m \vec{e} + \vec{e}^T \mathbf{A}_m^T \mathbf{P} \vec{e} \\
 &\quad + \vec{e}^T \mathbf{P} \mathbf{B}_m \Phi \vec{x}_p + \vec{e}^T \mathbf{P} \mathbf{B}_m \psi u + \vec{x}_p^T \Phi^T \mathbf{B}_m^T \mathbf{P} \vec{e} \\
 &\quad + u \psi \mathbf{B}_m^T \mathbf{P} \vec{e} + 2(\Phi \mathbf{M} \dot{\Phi}^T + \psi \dot{\psi} n) \\
 &= \vec{e}^T (\mathbf{P} \mathbf{A}_m + \mathbf{A}_m^T \mathbf{P}) \vec{e} \\
 &\quad + 2(\vec{e}^T \mathbf{P} \mathbf{B}_m \Phi \vec{x}_p + \vec{e}^T \mathbf{P} \mathbf{B}_m \psi u) \\
 &\quad + 2(\Phi \mathbf{M} \dot{\Phi}^T + \psi \dot{\psi} n) \\
 &= -\vec{e}^T \mathbf{Q} \vec{e} + 2\vec{e}^T \mathbf{P} (\mathbf{B}_m \Phi \vec{x}_p + \mathbf{B}_m \psi u) \\
 &\quad + 2(\Phi \mathbf{M} \dot{\Phi}^T + \psi \dot{\psi} n)
 \end{aligned} \tag{7.3.25}$$

We choose  $\Phi$  and  $\psi$  in order to satisfy the following equation

$$\begin{aligned}
 &\vec{e}^T \mathbf{P} (\mathbf{B}_m \Phi \vec{x}_p + \mathbf{B}_m \psi u) \\
 &\quad + (\Phi \mathbf{M} \dot{\Phi}^T + \psi \dot{\psi} n) = 0
 \end{aligned} \tag{7.3.26}$$

or

$$\begin{cases} \vec{e}^T \mathbf{P} \mathbf{B}_m \Phi \vec{x}_p + \Phi \mathbf{M} \dot{\Phi}^T = 0 \\ \vec{e}^T \mathbf{P} \mathbf{B}_m \psi u + \psi \dot{\psi} n = 0 \end{cases} \tag{7.3.27}$$

That results in

$$\dot{\Phi}^T = -\mathbf{M}^{-1} (\vec{e}^T \mathbf{P} \mathbf{B}_m \vec{x}_p) \tag{7.3.28}$$

or

$$\dot{\Phi} = -(\vec{x}_p^T \mathbf{B}_m^T \mathbf{P} \vec{e}) \mathbf{M}^{-1} \tag{7.3.29}$$

or

$$\dot{\mathbf{K}}_p = -(\vec{x}_p^T \mathbf{B}_m^T \mathbf{P} \vec{e}) \mathbf{M}^{-1} \tag{7.3.30}$$

and

$$\dot{\psi} = -n^{-1} u (\vec{e}^T \mathbf{P} \mathbf{B}_m) = -n^{-1} u (\mathbf{B}_m^T \mathbf{P} \vec{e}) \tag{7.3.31}$$

or

$$\dot{k}_r = -n^{-1} k_r^2 u (\mathbf{B}_m^T \mathbf{P} \vec{e}) \tag{7.3.32}$$

Hence the adaptive control law results as follows

$$\begin{cases} u = k_r(\mathbf{K}_p \vec{x}_p + \dot{w}) \\ \vec{e} = \vec{x}_p - \vec{x}_m \\ \dot{k}_r = -n^{-1}k_r^2 u (\mathbf{B}_m^T \mathbf{P} \vec{e}) \\ \dot{\mathbf{K}}_p = -(\vec{x}_p^T \mathbf{B}_m^T \mathbf{P} \vec{e}) \mathbf{M}^{-1} \end{cases} \quad (7.3.33)$$

and the time derivative of the Lyapunov function become

$$\dot{V} = -\vec{e}^T \mathbf{Q} \vec{e} \quad (7.3.34)$$

where, due to the fact  $\mathbf{A}_m$  is stable,  $\mathbf{Q}$  is positive definite and if persistent excitation exists such that  $\dot{V} = -\vec{e}^T \mathbf{Q} \vec{e} < 0$  (negative definite), then  $\vec{e}$  will converge asymptotically to the origin.

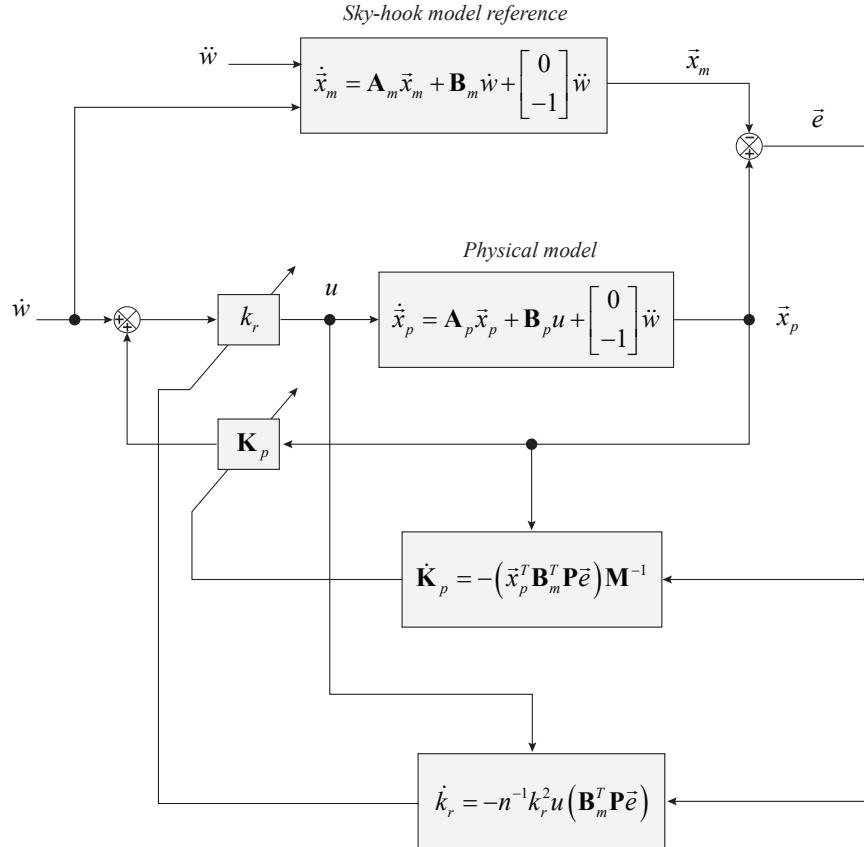


Figure 7.7: Control architecture for direct MRAC.

**Consider the following case study:**

- Model Reference

- $k_m = 50$

- $b_m = 20$
  - $m_m = 1$
  - Plant
    - $k_p = 25$
    - $b_p = 4$
    - $m_p = 2$
  - Control gain
    - $\mathbf{M} = \begin{bmatrix} 10^{-4} & 0 \\ 0 & 10^{-4} \end{bmatrix}$
    - $n = 20$
- Dynamic during street bump - without adaptive control

Dynamic during street bump
- (a) Behaviour of the quarter-suspension during a street bump in the case of none control action.

(b) Behaviour of the quarter-suspension during a street bump in the case of activation of the adaptive control.

Figure 7.8: Simulation results.

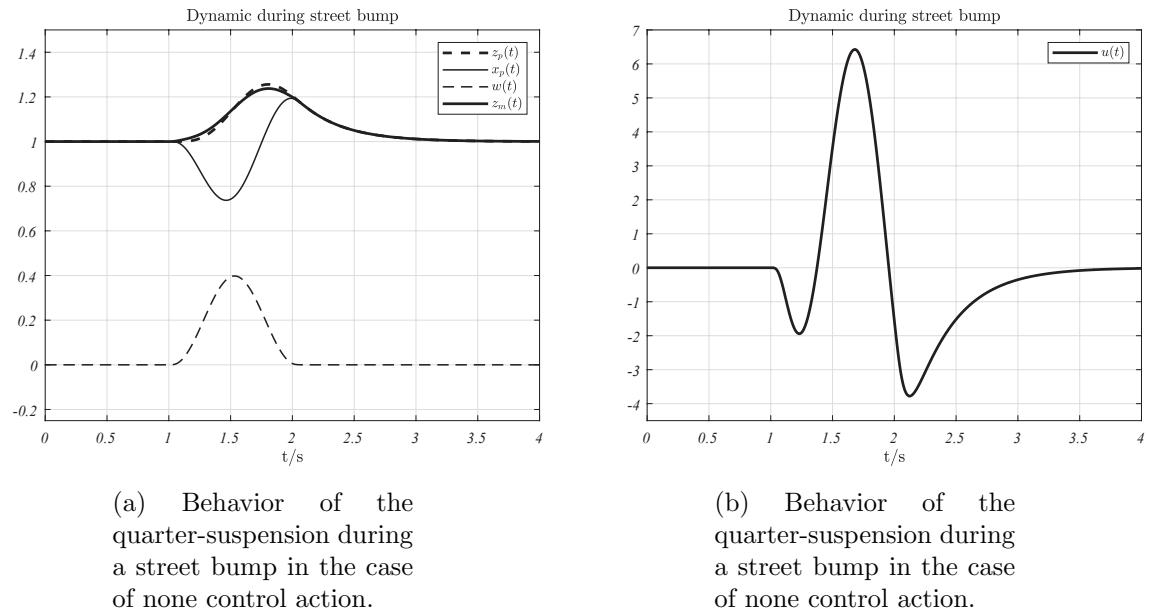


Figure 7.9: Simulation results.

## Chapter 8

# Dynamical System Models Derivation

### 8.1 Permanent-magnet Linear Actuator - Moving Coil

The moving coil was already studied at the beginning of this document. Here we want to represent the moving coil model using a different approach, we would like to use a more mathematical (and less physical) approach to show a more formal representation. Most of the following material has been taken from the monumental work of [21].

We define two reference frames: one stationary  $\mathcal{O}$  and one which lies on the moving coil  $\mathcal{O}'$  and we define the following meaning of the electrical field  $\vec{E}$  induced across the coil

- $\vec{E}'$  is the induced electrical field observed from the moving reference frame  $\mathcal{O}'$ .
- $\vec{E}$  is the induced electrical field observed from the stationary reference frame  $\mathcal{O}$ .

The Faraday's law applied to such that circuit can be written as follows

$$\oint_{\mathcal{C}} \vec{E}' \cdot d\vec{l} = -\frac{d}{dt} \int_{\mathcal{S}} \vec{B} \cdot \hat{n} da \quad (8.1.1)$$

where  $\vec{E}'$  is the electrical field applied to the coil respect the moving reference frame.

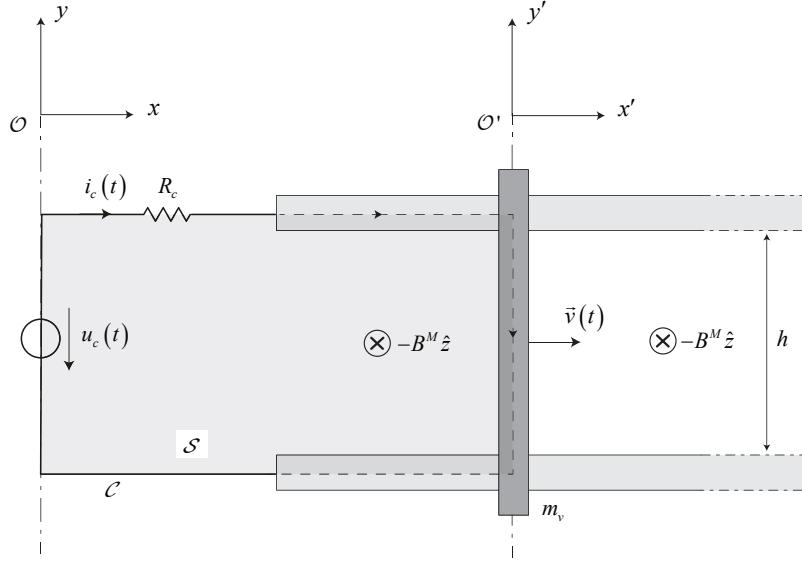


Figure 8.1: Moving coil actuator.

The circuit is moving at a certain speed  $\vec{v}$ , hence, the total time derivation of the linkage flux at the right hand side of Eq. 8.1.1 shall take into account that the linkage flux can change due to change of  $\vec{B}$  and due to the change of the contour  $\mathcal{C}$  (or  $\mathcal{S}$ ) due to the movement of the coil.

The total time derivative\* of the flux through the moving coil is (see [?])

$$\frac{d}{dt} \int_{\mathcal{S}} \vec{B} \cdot \hat{n} da = \int_{\mathcal{S}} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da + \oint_{\mathcal{C}} (\vec{B} \times \vec{v}) \cdot d\vec{l} \quad (8.1.2)$$

Hence Eq. 8.1.1 can now be written in the form

$$\oint_{\mathcal{C}} [\vec{E}' - \vec{v} \times \vec{B}] \cdot d\vec{l} = \int_{\mathcal{S}} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da \quad (8.1.3)$$

Eq. 8.1.3 can also be obtained as follows (see also Figure 8.2)

$$\oint_{\mathcal{C}} \vec{E}' \cdot d\vec{l} = -\frac{d}{dt} \int_{\mathcal{S}} \vec{B} \cdot \hat{n} da = -\frac{1}{dt} \left[ \int_{\mathcal{S}(t+dt)} \vec{B}(t+dt) \cdot \hat{n} da - \int_{\mathcal{S}(t)} \vec{B}(t) \cdot \hat{n} da \right] \quad (8.1.4)$$

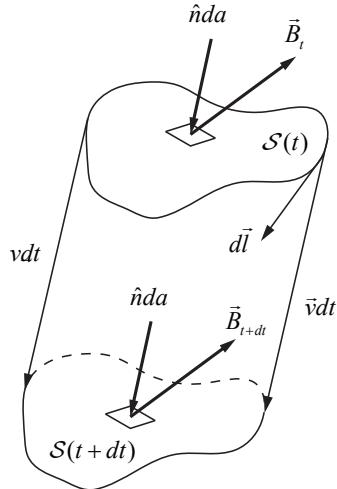
\*The concept of the convective derivative can be used

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \vec{v} \cdot \vec{\nabla}$$

we obtain that

$$\frac{d\vec{B}}{dt} = \frac{\partial \vec{B}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{B} = \frac{\partial \vec{B}}{\partial t} + \vec{\nabla} \times (\vec{B} \times \vec{v}) + \vec{v}(\vec{\nabla} \cdot \vec{B})$$

where  $\vec{v}$  is considered a constant vector. Applying Stokes theorem we obtain Eq. 8.1.2


 Figure 8.2: Evaluation of the  $d\vec{B}/dt$ .

where

$$\vec{B}(t + dt) = \vec{B}(t) + \frac{\partial \vec{B}}{\partial t} dt \quad (8.1.5)$$

The second term of Eq. 8.1.4 become

$$\begin{aligned} & -\frac{1}{dt} \left[ \int_{\mathcal{S}(t+dt)} \left( \vec{B}(t) + \frac{\partial \vec{B}}{\partial t} \right) \cdot \hat{n} da - \int_{\mathcal{S}(t)} \vec{B}(t) \cdot \hat{n} da \right] = \\ & -\frac{1}{dt} \left[ \int_{\mathcal{S}(t)} \vec{B}(t + dt) \cdot \hat{n} da - \int_{\mathcal{S}(t)} \vec{B}(t) \cdot \hat{n} da \right] - \int_{\mathcal{S}(t+dt)} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da \end{aligned} \quad (8.1.6)$$

The first term at the right side of the equation represents the variation of the flux of  $\vec{B}$  due to the only change of the surface, and we can write

$$-\frac{1}{dt} \left[ \int_{\mathcal{S}(t)} \vec{B}(t + dt) \cdot \hat{n} da - \int_{\mathcal{S}(t)} \vec{B}(t) \cdot \hat{n} da \right] = \oint_{\mathcal{C}} (\vec{v} \times \vec{B}) \cdot d\vec{l} \quad (8.1.7)$$

while the second term represent the variation of the flux due to the change of the vector  $\vec{B}$

$$\int_{\mathcal{S}(t+dt)} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da = \int_{\mathcal{S}(t)} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da \quad dt \rightarrow 0 \quad (8.1.8)$$

and we obtain the Eq. 8.1.3.

Now we apply the Faraday's law to the circuit respect the stationary reference frame  $\mathcal{O}$  and by the fact that the circuit  $\mathcal{C}$  and the surface  $\mathcal{S}$  are fixed respect to the same stationary reference frame  $\mathcal{O}$  the Faraday's law can be written in the form

$$\oint_{\mathcal{C}} \vec{E} \cdot d\vec{l} = - \int_{\mathcal{S}} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da \quad (8.1.9)$$

Galilean invariance Eq. 8.1.3 and Eq. 8.1.9 must be equal, hence results that the electrical field of the coil measured respect to  $\mathcal{O}$  can be expressed as (see also section 10.1)

$$\vec{E} = \vec{E}' - \vec{v} \times \vec{B} \quad (8.1.10)$$

Hence the Faraday's law applied to the moving coil

$$\oint_{\mathcal{C}} \vec{E}' \cdot d\vec{l} = -\frac{d}{dt} \int_{\mathcal{S}} \vec{B} \cdot \hat{n} da \quad (8.1.11)$$

where the circuit  $\mathcal{C}$  and the surface  $\mathcal{S}$  are not considered fixed, can be written in the form

$$\oint_{\mathcal{C}} [\vec{E}' - \vec{v} \times \vec{B}] \cdot d\vec{l} = \int_{\mathcal{S}} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da \quad (8.1.12)$$

where the circuit  $\mathcal{C}$  and the surface  $\mathcal{S}$  can be considered fixed but where we already account the speed of the coil.

From another, but equivalent, point of view, we can suppose to freeze, at a certain instant, the circuit and to define the contour  $\mathcal{C}$  and its surface  $\mathcal{S}$  as fixed respect to the stationary reference frame. Even if the contour  $\mathcal{C}$  and the surface  $\mathcal{S}$  are fixed in the space, the charge present into the coil is subjected to a vector speed  $\vec{v}$  and hence is subjected to a force  $\vec{f} = q(\vec{E} + \vec{v} \times \vec{B})$  respect to the reference frame  $\mathcal{O}$ , from Galilean invariance we can impose that  $\vec{f} = \vec{f}' = q\vec{E}'$  which results in  $\vec{E}' = \vec{E} + \vec{v} \times \vec{B}$ . That means, the induced electrical field along the coil seen from a stationary reference frame is given by

$$\vec{E} = \vec{E}' - \vec{v} \times \vec{B} \quad (8.1.13)$$

For ideal conductive coil we can write

$$\vec{E} = -\vec{v} \times \vec{B}.$$

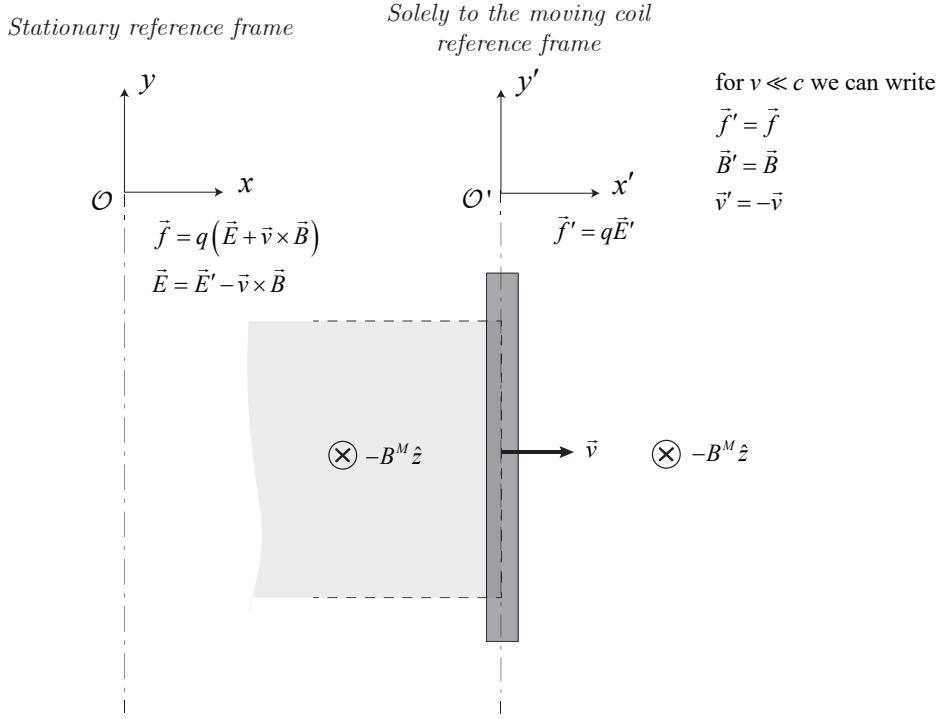


Figure 8.3: Lorentz force applied to the coil charge due to effect of the speed.

Now we can write the vector  $\vec{B}$  as a superposition of two fields: the  $\vec{B}_i$  due to the current  $i(t)$  and the external field  $\vec{B}^M$  which is supposed to be constant and isotropic,

$$\vec{B}(t) = \vec{B}_i(t) + \vec{B}^M$$

Therefore, the last term in Eq. 8.1.12 can be rewritten as

$$\int_{\mathcal{S}} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da = \int_{\mathcal{S}} \frac{\partial \vec{B}_i}{\partial t} \cdot \hat{n} da + \int_{\mathcal{S}} \frac{\partial \vec{B}^M}{\partial t} \cdot \hat{n} da \quad (8.1.14)$$

where

$$\frac{\partial \vec{B}^M}{\partial t} = 0 \quad (8.1.15)$$

$$\frac{\partial \vec{B}_i}{\partial t} = \frac{\partial \vec{B}_i(i)}{\partial i} \frac{di(t)}{dt} \quad (8.1.16)$$

Hence Eq. 8.1.14 can be written as

$$\int_{\mathcal{S}} \frac{\partial \vec{B}}{\partial t} \cdot \hat{n} da = \int_{\mathcal{S}} \left[ \frac{\partial \vec{B}_i(i)}{\partial i} \frac{di(t)}{dt} \right] \cdot \hat{n} da = \frac{\partial \phi(i)}{\partial i} \frac{di(t)}{dt} = L \frac{di(t)}{dt} \quad (8.1.17)$$

where  $\mathcal{S}$  is considered constant and  $\partial \phi(i)/\partial i = L$  is the definition of inductance.

Now we apply Eq. 8.1.12 in clockwise around the contour  $\mathcal{C}$  of the Figure 8.1 and we consider

the case that the coil is composed of  $N_c$  turns

$$\psi(t) = N_c \phi(t) = N_c \int_S \vec{B} \cdot d\vec{s} \quad (8.1.18)$$

$$L_c = N_c L$$

we obtain

$$\underbrace{\int_{(-)}^{(+)} \vec{E} \cdot d\vec{l}}_{\text{source}} + \underbrace{\int \vec{E} \cdot d\vec{l}}_{\text{resistor}} + \underbrace{\int \vec{E}' \cdot d\vec{l}}_{\text{coil}} = -\frac{d}{dt} \int_S \vec{B} \cdot \hat{n} da. \quad (8.1.19)$$

or

$$\underbrace{\int_{(-)}^{(+)} \vec{E} \cdot d\vec{l}}_{\text{source}} + \underbrace{\int \vec{E} \cdot d\vec{l}}_{\text{resistor}} + \underbrace{\int [\vec{E}' - \vec{v}_v \times \vec{B}^M N_c] \cdot d\vec{l}}_{\text{coil}} = -L_c \frac{di_c(t)}{dt}. \quad (8.1.20)$$

The integrals over the fixed components (source and resistor) follow from Equations

$$-u_c = \underbrace{\int_{(-)}^{(+)} \vec{E} \cdot d\vec{l}}_{\text{source}} \quad (8.1.21)$$

$$i_c R_c = \underbrace{\int \vec{E} \cdot d\vec{l}}_{\text{resistor}} \quad (8.1.22)$$

Concerning the coil we consider the case where the whole resistance is accounted in  $R_c$  and we set  $\vec{E}' = \vec{J}/\sigma = 0$ , we obtain

$$u_c(t) - i_c(t) R_c - L_c \frac{di_c}{dt} + \int_{\text{coil}} (\vec{v}_v \times \vec{B}^M N_c) \cdot d\vec{l} = 0 \quad (8.1.23)$$

We are considering the case where  $\vec{v}_v = v_v \hat{x}$ ,  $\vec{B}^M = -B^M \hat{z}$  and  $d\vec{l} = dy \hat{y}$ . Therefore ( $\hat{x} \times \hat{z} = -\hat{y}$ ),

$$\int_{\text{coil}} (\vec{v}_v \times \vec{B}^M N_c) \cdot d\vec{l} = \int_h^0 v_v B^M N_c dy = -N_c B^M h v_v(t) \quad (8.1.24)$$

we obtain

$$\boxed{u_c(t) - i_c(t) R_c - L_c \frac{di_c(t)}{dt} - N_c B^M h v_v(t) = 0} \quad (8.1.25)$$

To complete the linear pm-actuator model we must add the mechanical equations using the Newton's equation. To apply the Newton's law we first must calculate the corresponding force actuated by the iteration between the current  $i_c$  and the magnetic field  $\vec{B}^M$ .

The expression of the force due to the current  $i_c$  in the magnetic field  $\vec{B}^M$  is evaluated as follows

$$\vec{f}(t) = i(t) \int_{\text{wire}} d\vec{l} \times \vec{B}^{\text{ext}} \quad (8.1.26)$$

where  $\vec{B}^{\text{ext}}$  is an external field. Applying the integration of Eq. 8.1.26 it results in the following

equation (integration path follows the current direction)

$$f_v^m(t) = N_c i_c(t) \int_h^0 d\vec{l} \times \vec{B}^M = N_c B^M h i_c(t) \quad \text{positive x-direction} \quad (8.1.27)$$

where  $N_c$  by the number of turns linkage to the magnetic field. The complete set of system equations becomes

$$\begin{cases} \frac{dx_v(t)}{dt} = v_v(t) \\ \frac{dv_v(t)}{dt} = N_c \frac{B^M h}{m_v} i_c(t) - \frac{b_v}{m_v} v_v(t) - \frac{1}{m_v} f_v^e(t) \\ \frac{di_c(t)}{dt} = -\frac{R_c}{L_c} i_c(t) - \frac{1}{L_c} N_c B^M h v_v(t) + \frac{1}{L_c} u_c(t) \end{cases} \quad (8.1.28)$$

the first two equations represent the dynamic motion of the bar where the force  $f_v^m(t) = N_c B^M h i_c(t)$  is generated by the interaction of the current  $i_c(t)$  and the magnetic field  $\vec{B}^M$ . The third equation represents the Kirchhoff's voltage law of the electrical circuit which generates the current  $i_c(t)$  by applying the voltage  $u_c(t)$ . The equivalent auto-inductance  $L_c$  and the back-emf term,  $N_c B^M h v_v(t)$  represent the effect of the Faraday's law.

## 8.2 Permanent Magnet Synchronous Motor - Mathematical Model Derivation

In this chapter the permanent magnet synchronous machine control based model is derived. The following model will be used as plant to be controlled in the model predictive and Kalman-filter application.

Before to start with the pmsm modelization we recall some principle on magnetic induction law

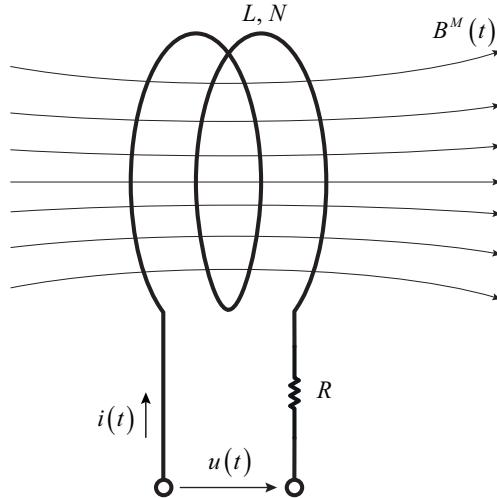


Figure 8.4: Stationary and rotating reference frame

$$e(t) = \frac{d\psi^r(t)}{dt} \quad (8.2.1)$$

where  $\psi^r(t) = N\phi(t)$  is the rotor flux linkage to the stator winding (made by  $N$  turns). The total winding flux linkage will be expressed as

$$\psi^s(t) = Li + \psi^r(t) \quad (8.2.2)$$

We can summarize the PMSM as a rotating permanent magnet coupled to three  $120^\circ$  shifted coils. During the rotation the time derivative of the magnetic flux linkages to the coils induces an electromagnetic force (emf) into the coils as

hence we can write the following generalized Kirchhoff equations

$$\begin{cases} \vec{u}_{abc} - \mathbf{R}\vec{i}_{abc} - \frac{d\vec{\psi}_{abc}^s}{dt} = 0 \\ J\dot{\omega}_m = \tau_m - \tau_l \\ \dot{\theta} = \omega = p\omega_m \end{cases} \quad (8.2.3)$$

where  $\vec{u}_{abc} = [u_a(t) \ u_b(t) \ u_c(t)]^T$ ,  $\vec{i}_{abc} = [i_a(t) \ i_b(t) \ i_c(t)]^T$  are the three phase voltage and current vector and the magnetic flux  $\vec{\psi}_{abc}^s$  represent the stator magnetic flux linkages,  $p$  is the number of pole-pairs.

Recall that

$$\vec{u}_{abc}^s - \mathbf{R}\vec{i}_{abc}^s - \frac{d\vec{\psi}_{abc}^s}{dt} = 0 \leftrightarrow \vec{u}_{abc} - \mathbf{R}\vec{i}_{abc} - \frac{d\vec{\psi}_{abc}^s}{dt} = 0$$

The flux linkages may be written as

$$\vec{\psi}_{abc}^s = \mathbf{L}_s \vec{i}_{abc} + \vec{\psi}_{abc}^r \quad (8.2.4)$$

where  $\vec{\psi}_{abc}^r$  are the rotor flux linked to the stator generated by the rotor permanent magnets

and where

$$\mathbf{L}_s = \begin{bmatrix} L_\mu + L_a - L_b \cos 2\theta & -\frac{1}{2}L_a - L_b \cos 2(\theta - \frac{\pi}{3}) & -\frac{1}{2}L_a - L_b \cos 2(\theta + \frac{\pi}{3}) \\ -\frac{1}{2}L_a - L_b \cos 2(\theta - \frac{\pi}{3}) & L_\mu + L_a - L_b \cos 2(\theta - \frac{2\pi}{3}) & -\frac{1}{2}L_a - L_b \cos 2(\theta + \pi) \\ -\frac{1}{2}L_a - L_b \cos 2(\theta + \frac{\pi}{3}) & -\frac{1}{2}L_a - L_b \cos 2(\theta + \pi) & L_\mu + L_a - L_b \cos 2(\theta + \frac{2\pi}{3}) \end{bmatrix} \quad (8.2.5)$$

is the stator inductance matrix.  $L_b$  is the anisotropic inductance term which is characteristic of internal permanent magnet machines.

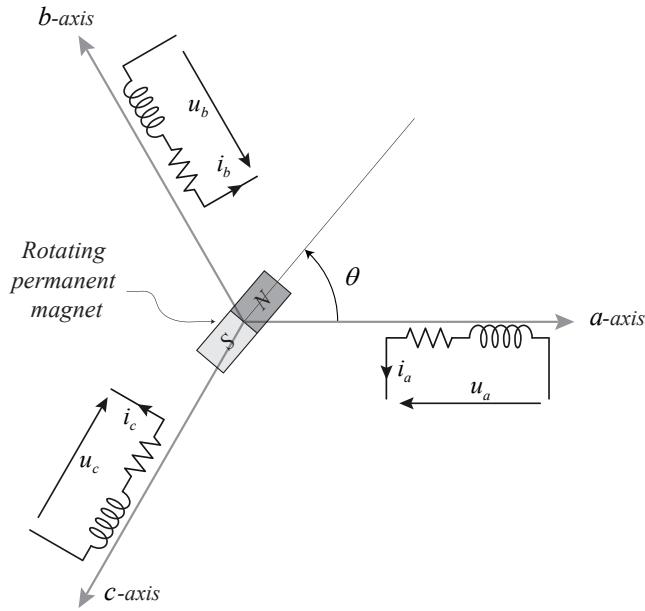


Figure 8.5: Stationary and rotating reference frame

In case of isotropic machine construction this term is null and the matrix of Eq. 8.2.5 becomes

$$\mathbf{L}_s = \begin{bmatrix} L_\mu + L_a & -\frac{1}{2}L_a & -\frac{1}{2}L_a \\ -\frac{1}{2}L_a & L_\mu + L_a & -\frac{1}{2}L_a \\ -\frac{1}{2}L_a & -\frac{1}{2}L_a & L_\mu + L_a \end{bmatrix} = \begin{bmatrix} L_s & L_m & L_m \\ L_m & L_s & L_m \\ L_m & L_m & L_s \end{bmatrix} \quad (8.2.6)$$

Where the term  $L_\mu$  is the leakage inductance and in general can be considered between one tenth and one twentieth of the synchronous inductance  $L_s \approx L_a$  and where  $L_m$  is the mutual inductance and is defined as negative value.

$$\left\{ \begin{array}{l} \vec{u}_{abc} - \mathbf{R}\vec{i}_{abc} - \omega \frac{d\mathbf{L}_s}{d\theta} \vec{i}_{abc} - \mathbf{L}_s \frac{d\vec{i}_{abc}}{dt} - \frac{d\vec{\psi}_{ab}^r}{dt} = 0 \\ J\dot{\omega}_m = \tau_m - \tau_l \\ \dot{\vartheta} = \omega = p\omega_m \end{array} \right. \quad (8.2.7)$$

where

$$\begin{cases} \psi_a^r(t) = \Psi^M \cos(\theta + \varphi_i) \\ \psi_b^r(t) = \Psi^M \cos(\theta - \frac{2\pi}{3} + \varphi_i) \\ \psi_c^r(t) = \Psi^M \cos(\theta + \frac{2\pi}{3}, \varphi_i) \end{cases} \quad (8.2.8)$$

where  $\Psi^M$  is the flux linkages generated by the permanent magnet.

### 8.2.1 PMSM model respect to a stationary reference frame

We are considering a 3-wire system which means we can apply the Kirchhoff condition  $x_a + x_b + x_c = 0$  that means we have three (not independent) vectors which lay in a plane (2-dimensions), hence the whole three phase system can be written by the use of only one independent vector and can be represented by two coordinates  $(\alpha, \beta)$ . Applying the coordinates transformation  $\mathbf{T}_1$  defined as follows:

$$\mathbf{T}_1 = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \quad \mathbf{T}_1^{-1} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \quad (8.2.9)$$

$$\mathbf{T}_1 \mathbf{T}_1^{-1} = \mathbf{I}_2 \quad (8.2.10)$$

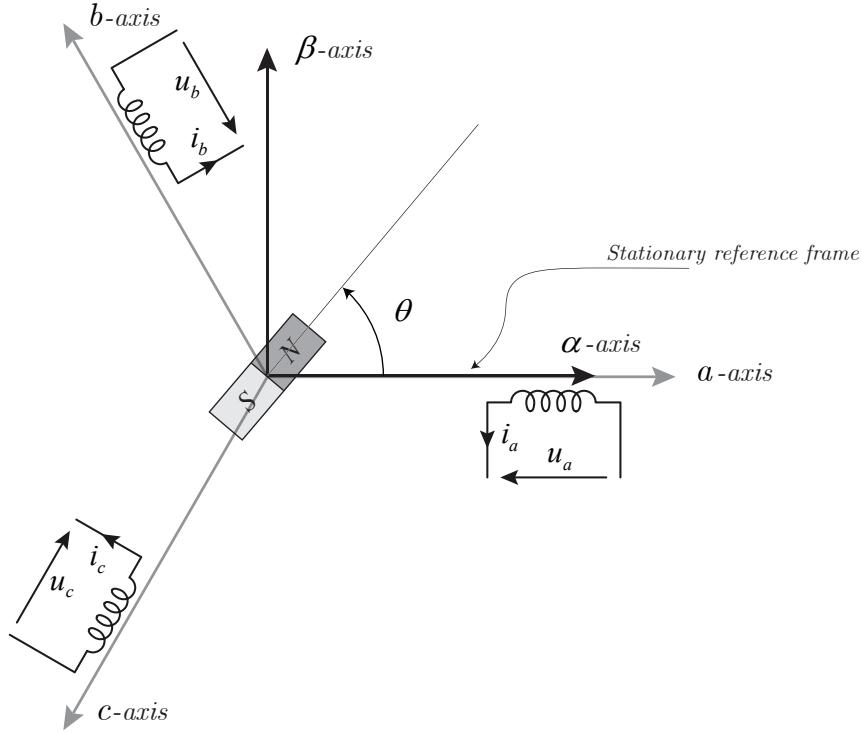
where

$$\vec{x}_{\alpha\beta} = \mathbf{T}_1 \vec{x}_{abc} \quad (8.2.11)$$

and

$$\mathbf{T}_1^{-1} \vec{x}_{\alpha\beta} = \vec{x}_{abc} \quad (8.2.12)$$

A graphical representation of the  $\alpha\beta$  reference frame is depicted in Figure 8.6.


 Figure 8.6: Stationary  $\alpha\beta$  reference frame

Applying the relation of Eq. 8.2.19 to the Eq. 8.2.4 we obtain

$$\mathbf{T}_1^{-1}\vec{u}_{\alpha\beta} - \mathbf{R}\mathbf{T}_1^{-1}\vec{i}_{\alpha\beta} - \omega \frac{d\mathbf{L}_s}{d\theta}\mathbf{T}_1^{-1}\vec{i}_{\alpha\beta} - \mathbf{L}_s\mathbf{T}_1^{-1}\frac{d\vec{i}_{\alpha\beta}}{dt} - \mathbf{T}_1^{-1}\frac{d\vec{\psi}_{\alpha\beta}^r}{dt} = 0$$

Multiplying left side for  $\mathbf{T}_1$  we obtain

$$\vec{u}_{\alpha\beta} - \mathbf{T}_1\mathbf{R}\mathbf{T}_1^{-1}\vec{i}_{\alpha\beta} - \omega \frac{d}{d\theta}[\mathbf{T}_1\mathbf{L}_s\mathbf{T}_1^{-1}]\vec{i}_{\alpha\beta} - \mathbf{T}_1\mathbf{L}_s\mathbf{T}_1^{-1}\frac{d\vec{i}_{\alpha\beta}}{dt} - \frac{d\vec{\psi}_{\alpha\beta}^r}{dt} = 0$$

which results as follows

$$\vec{u}_{\alpha\beta} - \mathbf{R}^{\alpha\beta}\vec{i}_{\alpha\beta} - \omega \frac{d\mathbf{L}_s^{\alpha\beta}}{d\theta}\vec{i}_{\alpha\beta} - \mathbf{L}_s^{\alpha\beta}\frac{d\vec{i}_{\alpha\beta}}{dt} - \frac{d\vec{\psi}_{\alpha\beta}^r}{dt} = 0 \quad (8.2.13)$$

where

$$\frac{d\vec{\psi}_{\alpha\beta}^r}{dt} = \omega \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r$$

in fact from Eq. 8.2.8 we can write as follows

$$\begin{cases} \psi_{\alpha}^r(t) = \Psi^M \cos \theta \\ \psi_{\beta}^r(t) = \Psi^M \sin \theta \end{cases} \quad (8.2.14)$$

which results in the following set of equations

$$\left\{ \begin{array}{l} \vec{u}_{\alpha\beta} - \mathbf{R}^{\alpha\beta} \vec{i}_{\alpha\beta} - \omega \frac{d\mathbf{L}_s^{\alpha\beta}}{d\theta} \vec{i}_{\alpha\beta} - \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}}{dt} - \omega \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r = 0 \\ \frac{d\vec{\psi}_{\alpha\beta}^r}{dt} = \omega \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r \\ J\dot{\omega} = \tau_m - \tau_l \\ \dot{\theta} = \omega \end{array} \right. \quad (8.2.15)$$

where

$$\mathbf{L}_s^{\alpha\beta} = \begin{bmatrix} L_\mu + \frac{3}{2}(L_a - L_b \cos 2\theta) & -\frac{3}{2}L_b \sin 2\theta \\ -\frac{3}{2}L_b \sin 2\theta & L_\mu + \frac{3}{2}(L_a + L_b \cos 2\theta) \end{bmatrix} \quad (8.2.16)$$

and

$$\frac{d\mathbf{L}_s^{\alpha\beta}}{d\theta} = 3L_b \begin{bmatrix} \sin 2\theta & -\cos 2\theta \\ -\cos 2\theta & -\sin 2\theta \end{bmatrix} \quad (8.2.17)$$

The isotropic ( $L_b = 0 \rightarrow d\mathbf{L}_s^{\alpha\beta}/d\theta = \mathbf{0}$ ) PMSM model respect to the stationary reference frame results as follows

$$\left\{ \begin{array}{l} \vec{u}_{\alpha\beta}(t) - \mathbf{R}^{\alpha\beta} \vec{i}_{\alpha\beta}(t) - \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}(t)}{dt} - \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r = 0 \\ \frac{d\vec{\psi}_{\alpha\beta}^r(t)}{dt} = \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r(t) \\ \vec{\psi}_{\alpha\beta}^r(0) = [\Psi^M \ 0]^T \\ J\dot{\omega}_m(t) = \tau_m(t) - \tau_l(t) \\ \dot{\theta}(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (8.2.18)$$

where  $\vec{\psi}_{\alpha\beta}^r = [\Psi^M \cos(\theta) \ \Psi^M \sin(\theta)]^T$  and

$$\begin{aligned} \mathbf{R}^{\alpha\beta} &= \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix} \\ \mathbf{L}^{\alpha\beta} &= \begin{bmatrix} L_\alpha & 0 \\ 0 & L_\beta \end{bmatrix} = \frac{3}{2} \begin{bmatrix} L_\mu + L_a & 0 \\ 0 & L_\mu + L_a \end{bmatrix} \end{aligned}$$

$\Psi^M$  is the amplitude of the permanent magnet flux as viewed from the stator phase winding

and we assume constant.

### 8.2.2 PMSM model respect to a rotating reference frame

Consider now a new change of the reference frame, in particular, we are going to consider the a rotating reference frame. Let  $\theta_0$  the phase of a not stationary reference frame here called  $(dq)$ , the set of equations reported in Eq. (8.2.15) can now be written respect the rotating reference frame as follows

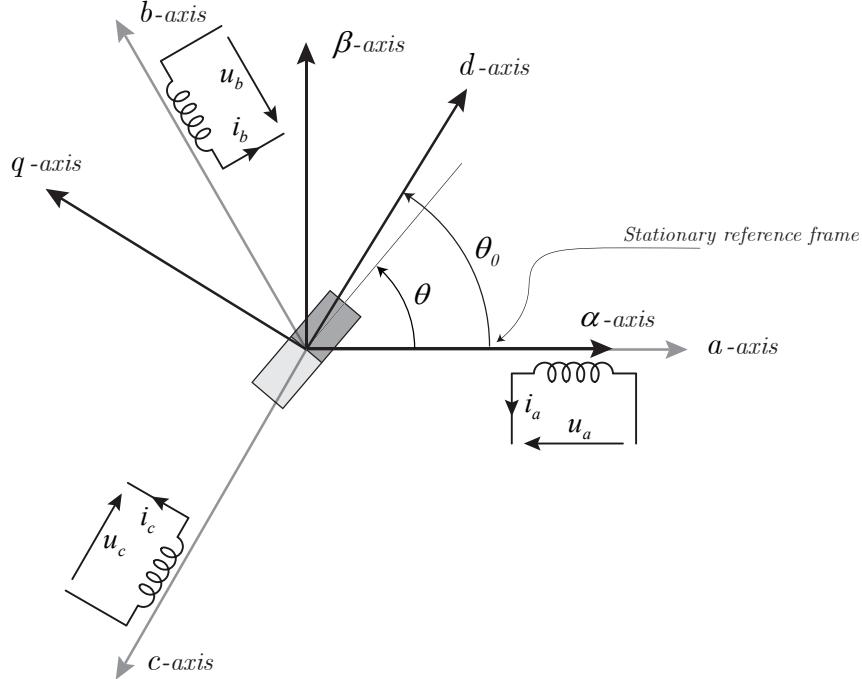


Figure 8.7: Rotating  $dq$  reference frame

$$\vec{x}_{dq} = \mathbf{T}_2 \vec{x}_{\alpha\beta} \quad (8.2.19)$$

$$\mathbf{T}_2^{-1} \vec{x}_{dq} = \vec{x}_{\alpha\beta} \quad (8.2.20)$$

where

$$\begin{aligned} \mathbf{T}_2(\theta_0) &= \begin{bmatrix} \cos(\theta_0) & \sin(\theta_0) \\ -\sin(\theta_0) & \cos(\theta_0) \end{bmatrix} \\ \mathbf{T}_2^{-1}(\theta_0) &= \begin{bmatrix} \cos(\theta_0) & -\sin(\theta_0) \\ \sin(\theta_0) & \cos(\theta_0) \end{bmatrix} \end{aligned} \quad (8.2.21)$$

$$\mathbf{T}_2(\theta_0) \mathbf{T}_2^{-1}(\theta_0) = \mathbf{I}_2$$

$$\dot{\theta}_0 = \omega_0$$

Note that  $d\mathbf{T}_2(\theta_0)/dt \neq 0$

Hence applying the transformation  $\mathbf{T}_2(\vartheta_0)$  to the system represented by Eq. (8.2.15) results as follows

$$\mathbf{T}_2^{-1}\vec{u}_{dq} - \mathbf{R}^{\alpha\beta} \mathbf{T}_2^{-1}\vec{i}_{dq} - \omega \frac{d\mathbf{L}_s^{\alpha\beta}}{d\theta} \mathbf{T}_2^{-1}\vec{i}_{dq} - \mathbf{L}^{\alpha\beta} \frac{d(\mathbf{T}_2^{-1}\vec{i}_{dq})}{dt} - \frac{d(\mathbf{T}_2^{-1}\vec{\psi}_{dq}^r)}{dt} = 0$$

where

$$\frac{d\mathbf{T}_2^{-1}(\vartheta_0)}{dt} = -\omega_0 \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \mathbf{T}_2^{-1}(\vartheta_0) \quad (8.2.22)$$

left multiplying by  $\mathbf{T}_2$  we obtain the following terms

$$\begin{aligned} & -\mathbf{T}_2 \mathbf{L}^{\alpha\beta} \frac{d(\mathbf{T}_2^{-1}\vec{i}_{dq})}{dt} = \\ & = \omega_0 \mathbf{T}_2 \mathbf{L}^{\alpha\beta} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \mathbf{T}_2^{-1}\vec{i}_{dq} + \mathbf{T}_2 \mathbf{L}^{\alpha\beta} \mathbf{T}_2^{-1} \frac{d\vec{i}_{dq}}{dt} \\ & \xrightarrow{\theta_0 \rightarrow \theta} \omega_0 \begin{pmatrix} 0 & L_d \\ -L_q & 0 \end{pmatrix} \vec{i}_{dq} + \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} \end{aligned} \quad (8.2.23)$$

where

$$\mathbf{L}^{dq} \xrightarrow{\theta_0 \rightarrow \theta} \begin{pmatrix} L_d & 0 \\ 0 & L_q \end{pmatrix} = \begin{pmatrix} L_\mu + \frac{3}{2}(L_a - L_b) & 0 \\ 0 & L_\mu + \frac{3}{2}(L_a + L_b) \end{pmatrix} \quad (8.2.24)$$

and

$$-\omega \mathbf{T}_2 \frac{d\mathbf{L}_s^{\alpha\beta}}{d\theta} \mathbf{T}_2^{-1}\vec{i}_{dq} = -\omega \mathbf{L}_m^{dq} \vec{i}_{dq} \xrightarrow{\theta_0 \rightarrow \theta} \omega \begin{pmatrix} 0 & 3L_b \\ 3L_b & 0 \end{pmatrix} \vec{i}_{dq} \quad (8.2.25)$$

Which results in the following set of equations

$$\left\{ \begin{array}{l} \vec{u}_{dq} - \mathbf{R}^{dq} \vec{i}_{dq} - \omega \mathbf{L}_m^{dq} \vec{i}_{dq} + \omega_0 \begin{pmatrix} 0 & L_d \\ -L_q & 0 \end{pmatrix} \vec{i}_{dq} + \\ - \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} + \omega_0 \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \vec{\psi}_{dq} + (\omega - \omega_0) \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \vec{\psi}_{dq}^r = 0 \\ \dot{\psi}_d^r = -(\omega - \omega_0) \psi_q^r \\ \dot{\psi}_q^r = (\omega - \omega_0) \psi_d^r \end{array} \right.$$

rearranging and approximating the term  $\omega \mathbf{L}_m^{dq} \vec{i}_{dq}$  to

$$-\omega \mathbf{L}_m^{dq} \vec{i}_{dq} \xrightarrow{\theta_0 \rightarrow \theta} \omega \begin{pmatrix} 0 & 3L_b \\ 3L_b & 0 \end{pmatrix} \vec{i}_{dq} = \omega \begin{pmatrix} 0 & L_q - L_d \\ L_q - L_d & 0 \end{pmatrix} \vec{i}_{dq}$$

we obtain the following set of equations

$$\left\{ \begin{array}{l} \vec{u}_{dq} - \mathbf{R}^{dq} \vec{i}_{dq} + \omega \begin{pmatrix} 0 & L_q - L_d \\ L_q - L_d & 0 \end{pmatrix} \vec{i}_{dq} + \omega_0 \begin{pmatrix} 0 & L_d \\ -L_q & 0 \end{pmatrix} \vec{i}_{dq} + \\ - \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} + \omega \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \vec{\psi}_{dq}^r = 0 \\ \dot{\psi}_d^r = -(\omega - \omega_0) \psi_q^r \\ \dot{\psi}_q^r = (\omega - \omega_0) \psi_d^r \end{array} \right. \quad (8.2.26)$$

recall that  $3L_b = (L_q - L_d)$ .

The flux equations in Eq. 8.2.26 are derived as follows

$$\begin{aligned} \frac{d\vec{\psi}_{\alpha\beta}^r(t)}{dt} &= \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r(t) \Rightarrow \\ \frac{d\mathbf{T}_2^{-1}\vec{\psi}_{dq}^r(t)}{dt} &= \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{T}_2^{-1}\vec{\psi}_{dq}^r(t) \Rightarrow \\ -\omega_0 \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \mathbf{T}_2^{-1}\vec{\psi}_{dq}^r + \mathbf{T}_2^{-1} \frac{d\vec{\psi}_{dq}^r(t)}{dt} &= \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{T}_2^{-1}\vec{\psi}_{dq}^r(t) \end{aligned} \quad (8.2.27)$$

Multiplying both terms per  $\mathbf{T}_2$  we obtain

$$\frac{d\vec{\psi}_{dq}^r(t)}{dt} = (\omega - \omega_0) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{dq}^r(t) \quad (8.2.28)$$

where  $\vec{\psi}_{dq}^r(t)$  is given as follows

$$\begin{aligned} \vec{\psi}_{dq}^r(t) &= \mathbf{T}_2 \vec{\psi}_{\alpha\beta}^r = \begin{bmatrix} \cos(\theta_0) & \sin(\theta_0) \\ -\sin(\theta_0) & \cos(\theta_0) \end{bmatrix} \begin{bmatrix} \Psi^M \cos(\theta_0) \\ \Psi^M \sin(\theta_0) \end{bmatrix} = \\ &= \Psi^M \begin{bmatrix} \cos \theta_0 \cos \theta + \sin \theta_0 \sin \theta \\ -\sin \theta_0 \cos \theta + \cos \theta_0 \sin \theta \end{bmatrix} = \Psi^M \begin{bmatrix} \cos(\theta - \theta_0) \\ \sin(\theta - \theta_0) \end{bmatrix} \end{aligned} \quad (8.2.29)$$

Considering the *DQ*-classical control, we can observe that, when the rotating reference frame (which is the frame where the controller is implemented) and the rotor reference frame are aligned ( $\theta_0 \rightarrow \theta$  and  $\omega_0 \rightarrow \omega$ ), all the inductance matrices become time-invariant and the flux components become  $\psi_q \rightarrow 0$ ,  $\psi_d \rightarrow \Psi^M$ .

The DQ-PMSM model for a generic rotating reference frame is here reported

$$\left\{ \begin{array}{l} \vec{u}_{dq} - \mathbf{R}^{dq} \vec{i}_{dq} + \omega \begin{pmatrix} 0 & L_q - L_d \\ L_q - L_d & 0 \end{pmatrix} \vec{i}_{dq} + \\ \omega_0 \begin{pmatrix} 0 & L_d \\ -L_q & 0 \end{pmatrix} \vec{i}_{dq} - \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} + \omega \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \vec{\psi}_{dq}^r = 0 \\ \dot{\psi}_d^r = -(\omega - \omega_0) \psi_q^r \\ \dot{\psi}_q^r = (\omega - \omega_0) \psi_d^r \\ \vec{\psi}_{dq}^r(0) = [\Psi^M \quad 0]^T \\ J\dot{\omega}_m(t) = \tau_m(t) - \tau_l(t) \\ \dot{\vartheta}(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (8.2.30)$$

where  $\vec{\psi}_{dq}^r = [\Psi^M \cos(\theta - \theta_0) \quad \Psi^M \sin(\theta - \theta_0)]^T$  and

$$\mathbf{R}^{dq} = \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix}$$

$$\mathbf{L}^{dq} = \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix} = \begin{bmatrix} L_\mu + \frac{3}{2}L_s - \frac{L_b}{2} & 0 \\ 0 & L_\mu + \frac{3}{2}L_s + \frac{L_b}{2} \end{bmatrix}$$

$\Psi^M$  is the amplitude of the permanent magnet flux as viewed from the stator phase winding and we assume constant.

The DQ-PMSM model, when rotating reference frame coincides with the rotor reference frame  $\omega_0 \rightarrow \omega$ , can be written as follows

$$\left\{ \begin{array}{l} \vec{u}_{dq} - \mathbf{R}^{dq} \vec{i}_{dq} + \omega \begin{pmatrix} 0 & L_q \\ -L_d & 0 \end{pmatrix} \vec{i}_{dq} - \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} + \omega \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \vec{\psi}_{dq}^r = 0 \\ \dot{\psi}_d^r = 0 \\ \dot{\psi}_q^r = 0 \\ \vec{\psi}_{dq}^r(0) = [\Psi^M \quad 0]^T \\ J\dot{\omega}_m(t) = \tau_m(t) - \tau_l(t) \\ \dot{\vartheta}(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (8.2.31)$$

where

$$\mathbf{R}^{dq} = \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix}$$

$$\mathbf{L}^{dq} = \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix} = \begin{bmatrix} L_\mu + \frac{3}{2}(L_a - L_b) & 0 \\ 0 & L_\mu + \frac{3}{2}(L_a + L_b) \end{bmatrix}$$

$\Psi^M$  is the amplitude of the permanent magnet flux as viewed from the stator phase winding and we assume constant.

### 8.2.3 PMSM Torque Derivation

#### Torque equation of the isotropic PMSM - stationary reference frame

Let  $E$  be the stored energy inside of the PMSM, it can be described as follows

$$E = \frac{3}{2} \left( \frac{1}{2} \vec{i}_{\alpha\beta}^T \mathbf{L}^{\alpha\beta} \vec{i}_{\alpha\beta} \right) + \frac{1}{2} J \omega_m^2$$

Applying the principle of energy conservation we can write

$$\frac{dE}{dt} = 0$$

$$\frac{dE}{dt} = \frac{3}{2} \left( \vec{i}_{\alpha\beta}^T \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}}{dt} \right) + J \omega_m \dot{\omega}_m = P_{in} - P_{out} - P_{loss} = 0$$

We can write

$$\begin{aligned} & \frac{3}{2} \left( \vec{i}_{\alpha\beta}^T \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}}{dt} \right) + J \omega_m \dot{\omega}_m = \\ &= \frac{3}{2} \left( \underbrace{u_\alpha i_\alpha + u_\beta i_\beta}_{=P_{in}} - \underbrace{R i_\alpha^2 - R i_\beta^2}_{=P_{loss}} - \underbrace{\omega \begin{bmatrix} i_\alpha & i_\beta \end{bmatrix} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \psi_\alpha^r & \psi_\beta^r \end{bmatrix}^T}_{=\omega_m \tau_m} \right) + \omega_m \tau_m - \omega_m \tau_l \end{aligned}$$

But

$$\frac{3}{2} \left( \vec{i}_{\alpha\beta}^T \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}}{dt} \right) + J \omega_m \dot{\omega}_m = P_{in} - P_{out} - P_{loss}$$

Hence

$$P_{in} = \frac{3}{2} (u_\alpha i_\alpha + u_\beta i_\beta)$$

$$P_{out} = \omega_m \tau_l$$

$$P_{loss} = \frac{3}{2} (R i_\alpha^2 + R i_\beta^2)$$

The expression of the torque  $\tau_m$  becomes as follows

$$\tau_m = \frac{3}{2} p (\psi_\alpha^r i_\beta - \psi_\beta^r i_\alpha) \quad (8.2.32)$$

**PMSM Mathematical Model in Stationary Reference Frame**

$$\left\{ \begin{array}{l} \vec{u}_{\alpha\beta}(t) - \mathbf{R}^{\alpha\beta} \vec{i}_{\alpha\beta}(t) - \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}(t)}{dt} - \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r = 0 \\ \frac{d\vec{\psi}_{\alpha\beta}^r(t)}{dt} = \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r(t) \quad \vec{\psi}_{\alpha\beta}^r(0) = [\Psi^M \quad 0]^T \\ J\dot{\omega}_m(t) = \frac{3}{2}p[\psi_{\alpha}^r(t)i_{\beta}(t) - \psi_{\beta}^r(t)i_{\alpha}(t)] - \tau_l(t) \\ \dot{\theta}(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (8.2.33)$$

or in canonical representation

$$\left\{ \begin{array}{l} \frac{d}{dt}i_{\alpha}(t) = -\frac{R_s}{L_{\alpha}}i_{\alpha}(t) + \frac{\omega(t)}{L_{\alpha}}\psi_{\beta}(t) + \frac{1}{L_{\alpha}}u_{\alpha}(t) \\ \frac{d}{dt}i_{\beta}(t) = -\frac{R_s}{L_{\beta}}i_{\beta}(t) - \frac{\omega(t)}{L_{\beta}}\psi_{\alpha}(t) + \frac{1}{L_{\beta}}u_{\beta}(t) \\ \frac{d}{dt}\psi_{\alpha}^r(t) = -\omega(t)\psi_{\beta}^r(t) \\ \frac{d}{dt}\psi_{\beta}^r(t) = \omega(t)\psi_{\alpha}^r(t) \\ J\dot{\omega}_m(t) = \frac{3}{2}p[\psi_{\alpha}^r(t)i_{\beta}(t) - \psi_{\beta}^r(t)i_{\alpha}(t)] - \tau_l(t) \\ \dot{\theta}(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (8.2.34)$$

An interesting model, useful for control and observer design considers the following approximation

$$\left\{ \begin{array}{l} \frac{d}{dt}i_{\alpha}(t) = -\frac{R_s}{L_{\alpha}}i_{\alpha}(t) + \frac{\omega(t)}{L_{\alpha}}\psi_{\beta}(t) + \frac{1}{L_{\alpha}}u_{\alpha}(t) \\ \frac{d}{dt}i_{\beta}(t) = -\frac{R_s}{L_{\beta}}i_{\beta}(t) - \frac{\omega(t)}{L_{\beta}}\psi_{\alpha}(t) + \frac{1}{L_{\beta}}u_{\beta}(t) \\ \frac{d}{dt}\psi_{\alpha}^r(t) = -\omega(t)\psi_{\beta}^r(t) \\ \frac{d}{dt}\psi_{\beta}^r(t) = \omega(t)\psi_{\alpha}^r(t) \\ \dot{\theta}(t) = \omega(t) \\ \dot{\omega}_m(t) = 0 \end{array} \right. \quad (8.2.35)$$

which results, in discrete-time domain, as follows

$$\begin{cases} i_\alpha(k+1) = \left(1 - t_s \frac{R_s}{L_\alpha}\right) i_\alpha(k) + t_s \frac{\omega(k)}{L_\alpha} \psi_\beta(k) + \frac{t_s}{L_\alpha} u_\alpha(k) \\ i_\beta(k+1) = \left(1 - t_s \frac{R_s}{L_\beta}\right) i_\beta(k) - t_s \frac{\omega(k)}{L_\beta} \psi_\alpha(k) + \frac{t_s}{L_\beta} u_\beta(k) \\ \psi_\alpha^r(k+1) = \psi_\alpha^r(k) - t_s \omega(k) \psi_\beta^r(k) \\ \psi_\beta^r(k+1) = \psi_\beta^r(k) + t_s \omega(k) \psi_\alpha^r(k) \\ \omega(k+1) = \omega(k) \\ \theta(k+1) = \theta(k) + t_s \omega(k) \end{cases} \quad (8.2.36)$$

### Torque equation of the anisotropic PMSM - rotor reference frame

Let  $E$  be the stored energy inside of the PMSM, it can be described as follows

$$E = \frac{3}{2} \left( \frac{1}{2} \vec{i}_{dq}^T \mathbf{L}^{dq} \vec{i}_{dq} \right) + \frac{1}{2} J \omega_m^2$$

Applying the principle of energy conservation we can write

$$\frac{dE}{dt} = 0$$

$$\frac{dE}{dt} = \frac{3}{2} \left( \vec{i}_{dq}^T \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} \right) + J \omega_m \dot{\omega}_m = P_{in} - P_{out} - P_{loss} = 0$$

We can write as follows

$$\begin{aligned} & \frac{3}{2} \left( \vec{i}_{dq}^T \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} \right) + J \omega_m \dot{\omega}_m = \\ & \underbrace{\frac{3}{2} (u_d i_d + u_q i_q)}_{=P_{in}} - \underbrace{\frac{3}{2} (R i_d^2 - R i_q^2)}_{=P_{loss}} + \\ & - \underbrace{\frac{3}{2} \left( \omega \begin{bmatrix} i_d & i_q \end{bmatrix} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \psi_d^r & \psi_q^r \end{bmatrix}^T + \omega \begin{bmatrix} i_d & i_q \end{bmatrix} \begin{bmatrix} 0 & -L_q \\ L_d & 0 \end{bmatrix} \begin{bmatrix} i_d & i_q \end{bmatrix}^T \right)}_{=\omega_m \tau_m} + \\ & + \omega_m \tau_m - \omega_m \tau_l \end{aligned}$$

But

$$\frac{3}{2} \left( \vec{i}_{dq}^T \mathbf{L}^{dq} \frac{d\vec{i}_{dq}}{dt} \right) + J \omega \dot{\omega} = P_{in} - P_{out} - P_{loss}$$

Hence

$$P_{in} = \frac{3}{2} (u_d i_d + u_q i_q)$$

$$P_{out} = \omega_m \tau_l$$

$$P_{loss} = \frac{3}{2} (R i_d^2 + R i_q^2)$$

The expression of the torque  $\tau_m$  becomes as follows

$$\tau_m = \frac{3}{2} p \left[ \psi_d^r i_q - \psi_q^r i_d + (L_d - L_q) i_d i_q \right] \xrightarrow{\theta_0 \rightarrow \theta} \frac{3}{2} p \left[ \Psi^M i_q + (L_d - L_q) i_d i_q \right] \quad (8.2.37)$$

**The full PMSM model respect to the rotor reference frame is as follows**

$$\begin{cases} \vec{u}_{dq}(t) - \mathbf{R}^{dq} \vec{i}_{dq}(t) + \omega \begin{bmatrix} 0 & L_q \\ -L_d & 0 \end{bmatrix} \vec{i}_{dq} - \mathbf{L}^{dq} \frac{d\vec{i}_{dq}(t)}{dt} - \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{dq}^r = 0 \\ \frac{d\vec{\psi}_{dq}^r(t)}{dt} = 0 \quad \vec{\psi}_{dq}^r(0) = [\Psi^M \quad 0]^T \\ J \dot{\omega}_m(t) = \frac{3}{2} p \left[ \Psi^M i_q + (L_d - L_q) i_d i_q \right] - \tau_l(t) \\ \dot{\theta}(t) = \omega(t) = p \omega_m(t) \end{cases} \quad (8.2.38)$$

#### 8.2.4 Equivalent Circuit of a PMSM

The DQ equivalent circuits of the PMSM is shown for both transient and steady state conditions.

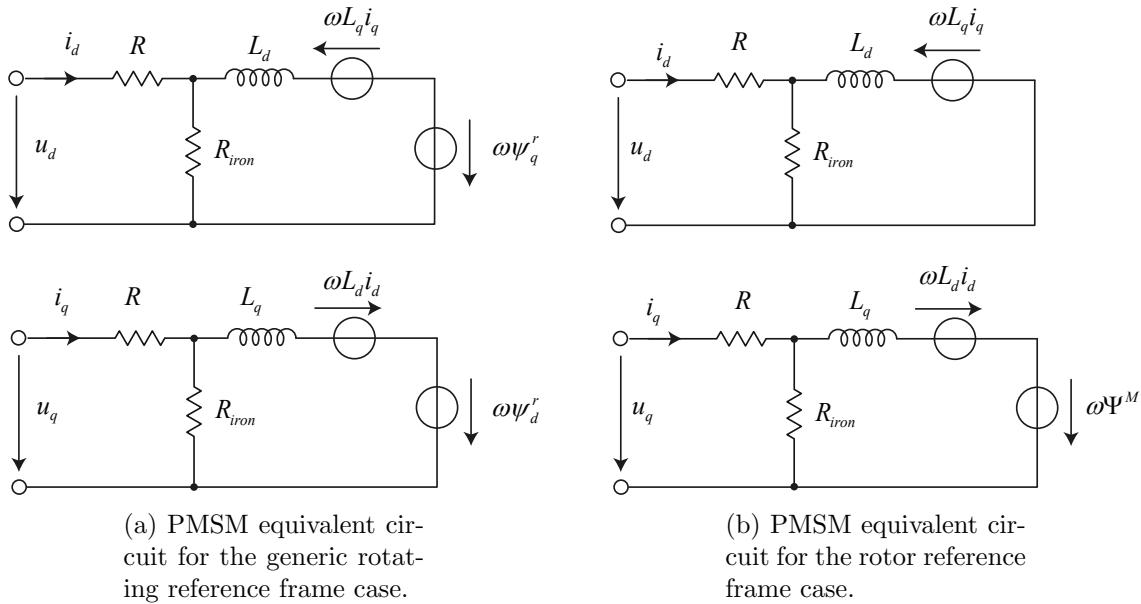


Figure 8.8: Rotating and rotor reference frame representation of the pmsm equivalent circuit.

### 8.3 Induction Motor Model Derivation

In this section we derive the dynamical model of the induction motor for control purpose. For the derivation of the dynamical model, the rotor windings are assumed to be identical to the stator windings with resistance  $R_r$  ( $R_s$  for the stator winding) and equivalent turns  $N_r = N_s$ .

Let

$$\vec{\psi}_{abc}^s = [\psi_{sa}, \psi_{sb}, \psi_{sc}]^T \quad \text{stator flux} \quad (8.3.1)$$

$$\vec{\psi}_{abc}^r = [\psi_{ra}, \psi_{rb}, \psi_{rc}]^T \quad \text{rotor flux} \quad (8.3.2)$$

be the vectors whose components are the stator and rotor flux linkages, respectively, with  $a, b, c$  denoting the three phases components. Similarly, let

$$\begin{aligned} \vec{i}_{abc}^s &= [i_{sa}, i_{sb}, i_{sc}]^T \quad \text{stator current} \\ \vec{i}_{abc}^r &= [i_{ra}, i_{rb}, i_{rc}]^T \quad \text{rotor current} \end{aligned} \quad (8.3.3)$$

be the vectors whose components are the stator and rotor currents. Then, for an induction motor with one pole pair we can write

$$\left. \begin{aligned} \frac{d\psi_{sa}}{dt} + R_s i_{sa} &= u_{sa} \\ \frac{d\psi_{sb}}{dt} + R_s i_{sb} &= u_{sb} \\ \frac{d\psi_{sc}}{dt} + R_s i_{sc} &= u_{sc} \\ \frac{d\psi_{ra}}{dt} + R_r i_{ra} &= 0 \\ \frac{d\psi_{rb}}{dt} + R_r i_{rb} &= 0 \\ \frac{d\psi_{rc}}{dt} + R_r i_{rc} &= 0 \end{aligned} \right\} \begin{array}{l} \text{respect to the stationary reference frame} \\ \text{respect to the rotor reference frame} \end{array} \quad (8.3.4)$$

where the stator and rotor fluxes, under the assumption of linear magnetic circuits, satisfy the linear relation

$$\begin{bmatrix} \vec{\psi}_{abc}^s \\ \vec{\psi}_{abc}^r \end{bmatrix} = \begin{bmatrix} \mathbf{L}_s & \mathbf{L}_m \\ \mathbf{L}_m^T & \mathbf{L}_r \end{bmatrix} \begin{bmatrix} \vec{i}_{abc}^s \\ \vec{i}_{abc}^r \end{bmatrix} \quad (8.3.5)$$

where

$$\mathbf{L}_s = \begin{bmatrix} L_\mu + L_s & -\frac{1}{2}L_s & -\frac{1}{2}L_s \\ -\frac{1}{2}L_s & L_\mu + L_s & -\frac{1}{2}L_s \\ -\frac{1}{2}L_s & -\frac{1}{2}L_s & L_\mu + L_s \end{bmatrix} \quad (8.3.6)$$

$$\mathbf{L}_r = \begin{bmatrix} L_\mu + L_r & -\frac{1}{2}L_r & -\frac{1}{2}L_r \\ -\frac{1}{2}L_r & L_\mu + L_r & -\frac{1}{2}L_r \\ -\frac{1}{2}L_r & -\frac{1}{2}L_r & L_\mu + L_r \end{bmatrix} \quad (8.3.7)$$

and

$$\mathbf{L}_m = L_m \begin{bmatrix} \cos(\vartheta_m) & \cos(\vartheta_m + \frac{2\pi}{3}) & \cos(\vartheta_m - \frac{2\pi}{3}) \\ \cos(\vartheta_m - \frac{2\pi}{3}) & \cos(\vartheta_m) & \cos(\vartheta_m + \frac{2\pi}{3}) \\ \cos(\vartheta_m + \frac{2\pi}{3}) & \cos(\vartheta_m - \frac{2\pi}{3}) & \cos(\vartheta_m) \end{bmatrix} \quad (8.3.8)$$

Considering the transformation

$$\mathbf{T}_1 = \frac{2}{3} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \quad (8.3.9)$$

such that

$$\begin{bmatrix} i_{s0} \\ i_{s\alpha} \\ i_{s\beta} \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_{sa} \\ i_{sb} \\ i_{sc} \end{bmatrix} \quad (8.3.10)$$

in which  $\mathbf{T}_1^{-1} = \mathbf{T}_1^T$ .

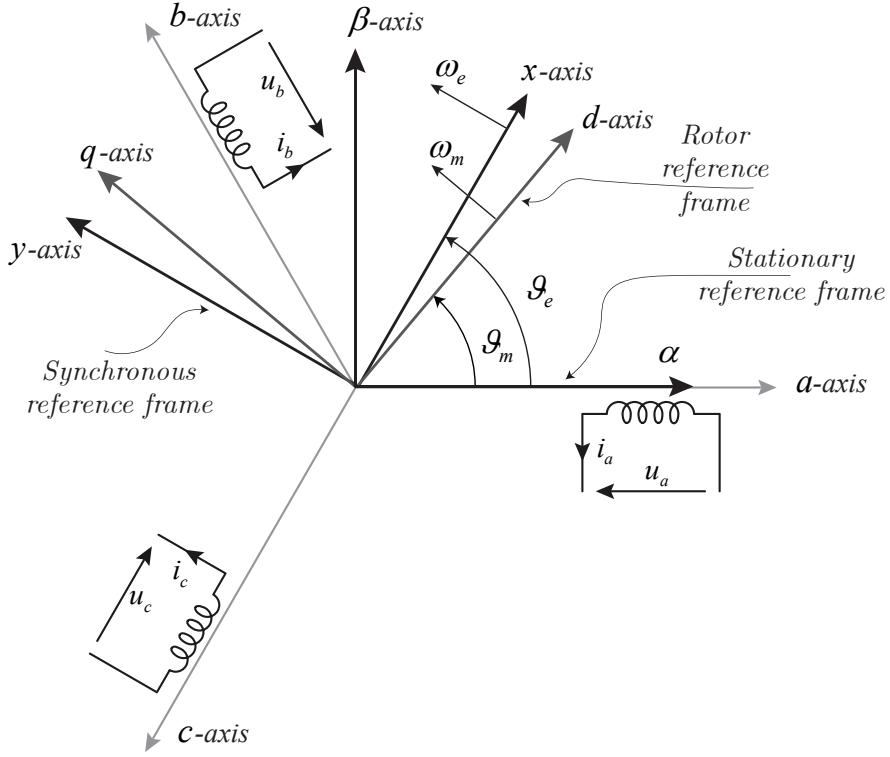


Figure 8.9: Machine reference frames description: stationary ( $\alpha, \beta$ ), rotor ( $d, q$ ) and synchronous speed ( $x, y$ ) reference frames.

The Eq. (8.3.4) of the induction motor become (considering balanced operating conditions,  $i_{s0} = 0, i_{r0} = 0$ )

$$\left. \begin{aligned} \frac{d\psi_{s\alpha}}{dt} + R_s i_{s\alpha} &= u_{s\alpha} \\ \frac{d\psi_{s\beta}}{dt} + R_s i_{s\beta} &= u_{s\beta} \end{aligned} \right\} \text{stationary reference frame}$$

$$\left. \begin{aligned} \frac{d\psi_{rd}}{dt} + R_r i_{rd} &= 0 \\ \frac{d\psi_{rq}}{dt} + R_r i_{rq} &= 0 \end{aligned} \right\} \text{rotor reference frame} \quad (8.3.11)$$

$$\left. \begin{aligned} \dot{\vartheta} &= \omega_m \\ J\dot{\omega}_m &= \tau_e - \tau_l \end{aligned} \right\} \text{mechanical equations}$$

where  $(\psi_{rd}, \psi_{rq})$  and  $(i_{rd}, i_{rq})$  denote the  $(d, q)$  components of the rotor flux and rotor current in the  $(d, q)$  reference frame attached to the rotor, rotating at rotor speed  $\omega_m = \dot{\vartheta}_m$ , while  $(\psi_{s\alpha}, \psi_{s\beta})$  and  $(u_{s\alpha}, u_{s\beta})$  denote the components of the stator flux and the stator voltage vectors in the fixed reference frame.

The linear operator for changing of coordinates between stationary reference frame and rotor

reference frame is

$$\mathbf{T}_2(\vartheta_m) = \begin{bmatrix} \cos \vartheta_m & \sin \vartheta_m \\ -\sin \vartheta_m & \cos \vartheta_m \end{bmatrix} \quad \mathbf{T}_2^{-1}(\vartheta_m) = \begin{bmatrix} \cos \vartheta_m & -\sin \vartheta_m \\ \sin \vartheta_m & \cos \vartheta_m \end{bmatrix} \quad (8.3.12)$$

such that

$$\begin{bmatrix} i_{rd} \\ i_{rq} \end{bmatrix} = \begin{bmatrix} \cos \vartheta_m & \sin \vartheta_m \\ -\sin \vartheta_m & \cos \vartheta_m \end{bmatrix} \begin{bmatrix} i_{r\alpha} \\ i_{r\beta} \end{bmatrix} \quad (8.3.13)$$

$$\begin{bmatrix} \psi_{rd} \\ \psi_{rq} \end{bmatrix} = \begin{bmatrix} \cos \vartheta_m & \sin \vartheta_m \\ -\sin \vartheta_m & \cos \vartheta_m \end{bmatrix} \begin{bmatrix} \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix} \quad (8.3.14)$$

with  $(\psi_{r\alpha}, \psi_{r\beta})$  and  $(i_{r\alpha}, i_{r\beta})$  denoting the  $(\alpha, \beta)$ -components of the rotor flux and current in the fixed frame.

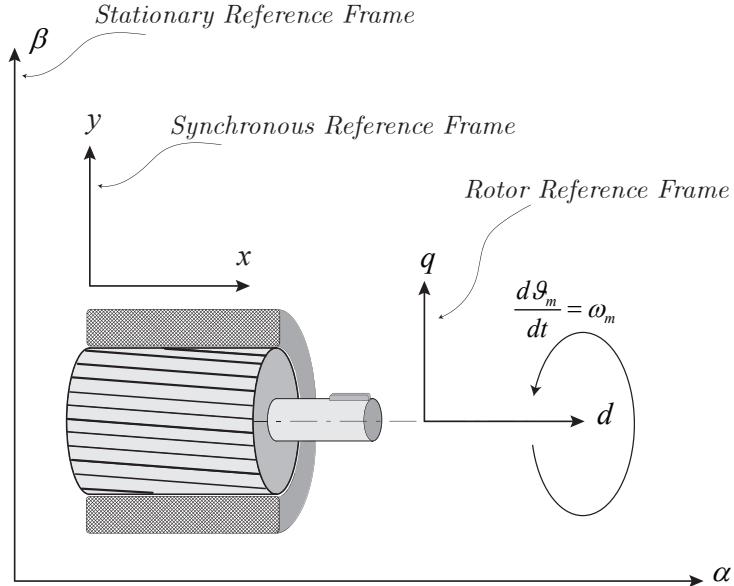


Figure 8.10: Different point of views regarding different reference frames: stationary  $(\alpha, \beta)$  can be considered in still condition; rotor  $(d, q)$  can be thought rotating at the rotor speed; synchronous reference frame  $(x, y)$  can be thought rotating at electrical stator pulsation.

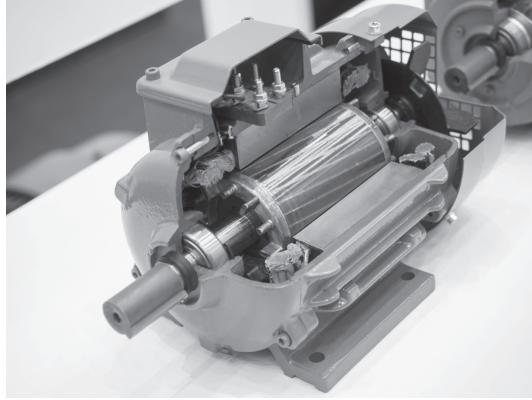


Figure 8.11: Internal view of a typical induction motor with squirrel cage rotor.

As we have seen, stator flux equations are represented respect the stationary reference frame while the rotor flux equations are represented respect to the rotor reference frame. Now we want to transform the rotor flux equations from the rotor to the stationary reference frame.

Considering that

$$\frac{d\mathbf{T}_2(\vartheta_m)}{dt} = -\omega_m \mathbf{T}_2(\vartheta_m) \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} = -\omega_m \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \mathbf{T}_2(\vartheta_m) \quad (8.3.15)$$

$$\frac{d\mathbf{T}_2^{-1}(\vartheta_m)}{dt} = -\omega_m \mathbf{T}_2^{-1}(\vartheta_m) \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = -\omega_m \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \mathbf{T}_2^{-1}(\vartheta_m) \quad (8.3.16)$$

The rotor flux equations respect to the stationary reference frame become as follows

$$\begin{aligned} \dot{\vec{\psi}}_{dq}^r &= \frac{d}{dt} \left( \mathbf{T}_2(\vartheta_m) \vec{\psi}_{\alpha\beta}^r \right) = -\omega_m \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \mathbf{T}_2(\vartheta_m) \vec{\psi}_{\alpha\beta}^r + \mathbf{T}_2(\vartheta_m) \dot{\vec{\psi}}_{\alpha\beta}^r \\ &= -\mathbf{R}_r \mathbf{T}_2(\vartheta_m) \vec{i}_{\alpha\beta}^r \end{aligned} \quad (8.3.17)$$

multiplying left side by  $\mathbf{T}_2^{-1}(\vartheta_m)$  we obtain

$$\dot{\vec{\psi}}_{\alpha\beta}^r = \omega_m \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \vec{\psi}_{\alpha\beta}^r - \mathbf{R}_r \vec{i}_{\alpha\beta}^r \quad (8.3.18)$$

or

$$\begin{bmatrix} \dot{\psi}_{r\alpha} \\ \dot{\psi}_{r\beta} \end{bmatrix} = - \begin{bmatrix} R_r & 0 \\ 0 & R_r \end{bmatrix} \begin{bmatrix} i_{r\alpha} \\ i_{r\beta} \end{bmatrix} - \begin{bmatrix} 0 & \omega_m \\ -\omega_m & 0 \end{bmatrix} \begin{bmatrix} \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix} \quad (8.3.19)$$

Now we can write all terms of the Eq. (8.3.11) in  $(\alpha, \beta)$  coordinates as follows

$$\begin{aligned}\begin{bmatrix} \dot{\psi}_{s\alpha} \\ \dot{\psi}_{s\beta} \end{bmatrix} &= - \begin{bmatrix} R_s & 0 \\ 0 & R_s \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix} + \begin{bmatrix} u_{s\alpha} \\ u_{s\beta} \end{bmatrix} \\ \begin{bmatrix} \dot{\psi}_{r\alpha} \\ \dot{\psi}_{r\beta} \end{bmatrix} &= - \begin{bmatrix} R_r & 0 \\ 0 & R_r \end{bmatrix} \begin{bmatrix} i_{r\alpha} \\ i_{r\beta} \end{bmatrix} - \begin{bmatrix} 0 & \omega_m \\ -\omega_m & 0 \end{bmatrix} \begin{bmatrix} \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix}\end{aligned}\quad (8.3.20)$$

The electromagnetic equations are

$$\begin{bmatrix} \psi_{s\alpha} \\ \psi_{s\beta} \\ \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix} = \begin{bmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \\ i_{r\alpha} \\ i_{r\beta} \end{bmatrix} = \mathbf{L} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \\ i_{r\alpha} \\ i_{r\beta} \end{bmatrix} \quad (8.3.21)$$

i.e. the quadratic form associated to  $\mathbf{L}$

$$\frac{3}{2} \left( \frac{1}{2} \vec{i}_{\alpha\beta}^T \mathbf{L} \vec{i}_{\alpha\beta} \right) \quad (8.3.22)$$

is positive for any non-zero value of the current vector  $\vec{i}_{\alpha\beta} = [i_{s\alpha}, i_{s\beta}, i_{r\alpha}, i_{r\beta}]$  and represent the magnetic energy.

Define

$$\mathbf{R} = \begin{bmatrix} R_s & 0 & 0 & 0 \\ 0 & R_s & 0 & 0 \\ 0 & 0 & R_r & 0 \\ 0 & 0 & 0 & R_r \end{bmatrix} \quad (8.3.23)$$

the state space model representation of the system Eq. (8.3.20) is

$$\begin{aligned}
 \mathbf{L} \begin{bmatrix} \frac{d}{dt} i_{s\alpha} \\ \frac{d}{dt} i_{s\beta} \\ \frac{d}{dt} i_{r\alpha} \\ \frac{d}{dt} i_{r\beta} \end{bmatrix} &= -\mathbf{R} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \\ i_{r\alpha} \\ i_{r\beta} \end{bmatrix} + \\
 &+ \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \omega_m L_m & 0 & \omega_m L_r \\ -\omega_m L_m & 0 & -\omega_m L_r & 0 \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \\ i_{r\alpha} \\ i_{r\beta} \end{bmatrix} + \begin{bmatrix} u_{s\alpha} \\ u_{s\beta} \\ 0 \\ 0 \end{bmatrix} \\
 J \frac{d}{dt} \omega_m &= \tau_e - \tau_l
 \end{aligned} \tag{8.3.24}$$

in which the state variables are  $(i_{s\alpha}, i_{s\beta}, i_{r\alpha}, i_{r\beta}, \omega_m)$  and the electromagnetic torque  $\tau_e$  is still to be determined as a function of the state variables. This choice of state variables is naturally linked to the energy stored in the energy stored in the motor given by

$$E = \frac{3}{2} \left( \frac{1}{2} \vec{i}_{\alpha\beta}^T \mathbf{L} \vec{i}_{\alpha\beta} \right) + \frac{1}{2} J \omega_m^2 \tag{8.3.25}$$

which is the sum of the magnetic energy and of the kinetic energy

$$\frac{1}{2} J \omega_m^2 \tag{8.3.26}$$

The expression of  $\tau_e$  can be obtained from the energy balance

$$\begin{aligned}
 \frac{dE}{dt} &= \frac{3}{2} \begin{bmatrix} i_{s\alpha} & i_{s\beta} \end{bmatrix} \begin{bmatrix} u_{s\alpha} & u_{s\beta} \end{bmatrix}^T - \tau_l \omega_m - \frac{3}{2} \vec{i}_{\alpha\beta}^T \mathbf{R} \vec{i}_{\alpha\beta} + \\
 &+ \frac{3}{2} \omega_m \begin{bmatrix} i_{r\alpha} & i_{r\beta} \end{bmatrix} \begin{bmatrix} 0 & -L_m \\ L_m & 0 \end{bmatrix} \begin{bmatrix} i_{s\alpha} & i_{s\beta} \end{bmatrix}^T + \tau_e \omega_m
 \end{aligned} \tag{8.3.27}$$

Let  $P_{in}$ ,  $P_{out}$  and  $P_{loss}$  denote the input power, the output power, and the power losses, respectively; since

$$\begin{aligned}
 \frac{dE}{dt} &= P_{in} - P_{out} - P_{loss} \\
 &= \frac{3}{2} \begin{bmatrix} i_{s\alpha} & i_{s\beta} \end{bmatrix} \begin{bmatrix} u_{s\alpha} & u_{s\beta} \end{bmatrix}^T - \tau_l \omega_m - \frac{3}{2} \vec{i}_{\alpha\beta}^T \mathbf{R} \vec{i}_{\alpha\beta}
 \end{aligned} \tag{8.3.28}$$

which means

$$\tau_e = \frac{3}{2} \begin{bmatrix} i_{r\alpha} & i_{r\beta} \end{bmatrix} \begin{bmatrix} 0 & -L_m \\ L_m & 0 \end{bmatrix} \begin{bmatrix} i_{s\alpha} & i_{s\beta} \end{bmatrix}^T = \frac{3}{2} L_m (i_{r\alpha} i_{s\beta} - i_{r\beta} i_{s\alpha}) \quad (8.3.29)$$

Note that the electromagnetic torque  $\tau_e$  produced by the motor is a non-linear function of the state variables and constitutes the main non-linear term in the induction motor model. Since

$$\begin{aligned} i_{r\alpha} &= -\frac{L_m}{L_r} i_{s\alpha} + \frac{1}{L_r} \psi_{r\alpha} \\ i_{r\beta} &= -\frac{L_m}{L_r} i_{s\beta} + \frac{1}{L_r} \psi_{r\beta} \end{aligned} \quad (8.3.30)$$

the electromagnetic torque can be expressed as

$$\tau_e = \frac{3}{2} \frac{L_m}{L_r} (\psi_{r\alpha} i_{s\beta} - \psi_{r\beta} i_{s\alpha}). \quad (8.3.31)$$

Eliminating  $(\psi_{s\alpha}, \psi_{s\beta}, i_{r\alpha}, i_{r\beta})$  from equation 8.3.20 using equations 8.3.20 and 8.3.30, namely

$$\begin{bmatrix} i_{r\alpha} \\ i_{r\beta} \end{bmatrix} = \begin{bmatrix} -\frac{L_m}{L_r} & 0 \\ 0 & -\frac{L_m}{L_r} \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix} + \begin{bmatrix} \frac{1}{L_r} & 0 \\ 0 & \frac{1}{L_r} \end{bmatrix} \begin{bmatrix} \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix} \quad (8.3.32)$$

$$\begin{bmatrix} \psi_{s\alpha} \\ \psi_{s\beta} \end{bmatrix} = \begin{bmatrix} L_s - \frac{L_m^2}{L_r} & 0 \\ 0 & L_s - \frac{L_m^2}{L_r} \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix} + \begin{bmatrix} \frac{L_m}{L_r} & 0 \\ 0 & \frac{L_m}{L_r} \end{bmatrix} \begin{bmatrix} \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix} \quad (8.3.33)$$

we obtain a state model in terms of the state variables  $(\omega_m, \psi_{r\alpha}, \psi_{r\beta}, i_{s\alpha}, i_{s\beta})^T$ , the input variables  $(u_{s\alpha}, u_{s\beta})^T$  and the disturbance  $\tau_l$

#### IM Dynamical Model Respect to a Stationary Reference Frame

$$\begin{aligned} \frac{di_{s\alpha}}{dt} &= -\gamma i_{s\alpha} + \beta \alpha \psi_{r\alpha} + \beta \omega_m \psi_{r\beta} + \frac{u_{s\alpha}}{\sigma} \\ \frac{di_{s\beta}}{dt} &= -\gamma i_{s\beta} + \beta \alpha \psi_{r\beta} - \beta \omega_m \psi_{r\alpha} + \frac{u_{s\beta}}{\sigma} \\ \frac{d\psi_{r\alpha}}{dt} &= -\alpha \psi_{r\alpha} - \omega_m \psi_{r\beta} + \alpha L_m i_{s\alpha} \\ \frac{d\psi_{r\beta}}{dt} &= -\alpha \psi_{r\beta} + \omega_m \psi_{r\alpha} + \alpha L_m i_{s\beta} \\ \frac{d\omega_m}{dt} &= \mu (\psi_{r\alpha} i_{s\beta} - \psi_{r\beta} i_{s\alpha}) - \frac{\tau_l}{J} \end{aligned} \quad (8.3.34)$$

where

$$\begin{aligned}\mu &= \frac{3}{2} \frac{L_m}{JL_r}, \quad \alpha = \frac{R_r}{L_r}, \quad \sigma = L_s \left(1 - \frac{L_m^2}{L_s L_r}\right) \\ \beta &= \frac{L_m}{\sigma L_r}, \quad \gamma = \frac{R_s}{\sigma} + \beta \alpha L_m\end{aligned}\tag{8.3.35}$$

Consider now the condition of constant speed

$$\frac{d\omega_m}{dt} = 0 \Rightarrow \omega_m = \omega_m^0 \tag{8.3.36}$$

which brings the above system from a nonlinear time-variant into a linear time-invariant as shown in the following

**IM Dynamical Model Respect to a Stationary Reference Frame and Constant Rotor Speed**

$$\begin{aligned}\dot{\vec{x}} &= \tilde{\mathbf{A}} \vec{x} + \tilde{\mathbf{B}} \vec{u} \\ \vec{y} &= \mathbf{C} \vec{x}\end{aligned}\tag{8.3.37}$$

where

$$\begin{aligned}\vec{x} &= [i_{s\alpha} \quad i_{s\beta} \quad \psi_{r\alpha} \quad \psi_{r\beta}]^T \\ \vec{u} &= [u_{s\alpha} \quad u_{s\beta}]^T \\ \vec{y} &= [i_{s\alpha} \quad i_{s\beta}]^T \\ \tilde{\mathbf{A}} &= \begin{bmatrix} -\gamma & 0 & \beta\alpha & \beta\omega_m^0 \\ 0 & -\gamma & -\beta\omega_m^0 & \beta\alpha \\ \alpha L_m & 0 & -\alpha & -\omega_m^0 \\ 0 & \alpha L_m & \omega_m^0 & -\alpha \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} \frac{1}{\sigma} & 0 \\ 0 & \frac{1}{\sigma} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \mathbf{C} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\end{aligned}\tag{8.3.38}$$

The expression of the electromagnetic torque is as follows

$$\tau_e = \frac{3}{2} \frac{L_m}{L_r} (\psi_{r\alpha} i_{s\beta} - \psi_{r\beta} i_{s\alpha}) \tag{8.3.39}$$

The state space model 8.3.34, which will be referred to as fixed frame model, has some advantages from the control view point since it clarifies that the control inputs ( $u_{s\alpha}, u_{s\beta}$ ) directly affect the dynamics of the stator currents ( $i_{s\alpha}, i_{s\beta}$ ) which can be viewed as immediate control variables since they control the rotor speed  $\omega_m$  and the rotor flux modulus  $\sqrt{\psi_{r\alpha}^2 + \psi_{r\beta}^2}$  in the first three equations.

### 8.3. Induction Motor Model Derivation

---

Let us now introduce a time-varying ( $d, q$ ) frame which rotates at an arbitrary speed  $\omega_0(t)$  and is identified at each time  $t$  by the angle  $\vartheta_0(t)$  so that

$$\frac{d\vartheta_0}{dt} = \omega_0 \quad (8.3.40)$$

Rotor fluxes ( $\psi_{rd}, \psi_{r\beta}$ ), stator currents ( $i_{s\alpha}, i_{s\beta}$ ), and stator voltages ( $u_{s\alpha}, u_{s\beta}$ ) are expressed with respect to the time-varying rotating ( $d, q$ ) frame as

$$\begin{aligned} \begin{bmatrix} \psi_{rd} \\ \psi_{rq} \end{bmatrix} &= \begin{bmatrix} \cos \vartheta_0 & \sin \vartheta_0 \\ -\sin \vartheta_0 & \cos \vartheta_0 \end{bmatrix} \begin{bmatrix} \psi_{r\alpha} \\ \psi_{r\beta} \end{bmatrix} \\ \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} &= \begin{bmatrix} \cos \vartheta_0 & \sin \vartheta_0 \\ -\sin \vartheta_0 & \cos \vartheta_0 \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix} \\ \begin{bmatrix} u_{sd} \\ u_{sq} \end{bmatrix} &= \begin{bmatrix} \cos \vartheta_0 & \sin \vartheta_0 \\ -\sin \vartheta_0 & \cos \vartheta_0 \end{bmatrix} \begin{bmatrix} u_{s\alpha} \\ u_{s\beta} \end{bmatrix} \end{aligned} \quad (8.3.41)$$

If the new **state variables**  $(\omega_m, \psi_{rd}, \psi_{rq}, i_{sd}, i_{sq})^T$  and input variables  $(u_{sd}, u_{sq})$  are used, in the new ( $d, q$ ) rotating coordinates the induction motor model becomes; consider the quantity  $\tau_l$  as disturbance.

#### IM Dynamical Model Respect to a Rotating Reference Frame

$$\begin{aligned} \frac{d\omega_m}{dt} &= \mu (\psi_{rd}i_{sq} - \psi_{rq}i_{sd}) - \frac{\tau_l}{J} \\ \frac{d\psi_{rd}}{dt} &= -\alpha\psi_{rd} + (\omega_0 - \omega_m)\psi_{rq} + \alpha L_m i_{sd} \\ \frac{d\psi_{rq}}{dt} &= -\alpha\psi_{rq} - (\omega_0 - \omega_m)\psi_{rd} + \alpha L_m i_{sq} \\ \frac{di_{sd}}{dt} &= -\gamma i_{sd} + \omega_0 i_{sq} + \beta\alpha\psi_{rd} + \beta\omega_m\psi_{rq} + \frac{u_{sd}}{\sigma} \\ \frac{di_{sq}}{dt} &= -\gamma i_{sq} - \omega_0 i_{sd} + \beta\alpha\psi_{rq} - \beta\omega_m\psi_{rd} + \frac{u_{sq}}{\sigma} \end{aligned} \quad (8.3.42)$$

where  $\omega_0$  is the speed of the rotating reference frame **where**

$$\begin{aligned} \mu &= \frac{3}{2} \frac{L_m}{JL_r}, \quad \alpha = \frac{R_r}{L_r}, \quad \sigma = L_s \left(1 - \frac{L_m^2}{L_s L_r}\right) \\ \beta &= \frac{L_m}{\sigma L_r}, \quad \gamma = \frac{R_s}{\sigma} + \beta\alpha L_m \end{aligned} \quad (8.3.43)$$

In the case the rotating reference is attached to the rotor the induction motor model becomes

**IM Dynamical Model Respect to the Rotor Reference Frame**

$$\begin{aligned}
 \frac{d\omega_m}{dt} &= \mu(\psi_{rd}i_{sq} - \psi_{rq}i_{sd}) - \frac{\tau_l}{J} \\
 \frac{d\psi_{rd}}{dt} &= -\alpha\psi_{rd} + \alpha L_m i_{sd} \\
 \frac{d\psi_{rq}}{dt} &= -\alpha\psi_{rq} + \alpha L_m i_{sq} \\
 \frac{di_{sd}}{dt} &= -\gamma i_{sd} + \omega_m i_{sq} + \beta\alpha\psi_{rd} + \beta\omega_m\psi_{rq} + \frac{u_{sd}}{\sigma} \\
 \frac{di_{sq}}{dt} &= -\gamma i_{sq} - \omega_m i_{sd} + \beta\alpha\psi_{rq} - \beta\omega_m\psi_{rd} + \frac{u_{sq}}{\sigma}
 \end{aligned} \tag{8.3.44}$$

The rotor frame state space model has some advantages from observability view point given by the linearity of the rotor flux equation where the cross terms are deleted from the selected rotating reference frame.

## 8.4 Lithium-ion Battery Model

### 8.4.1 Principle of Operation

In this section we are going to describe the lithium-ion cell operation using the simplified schematic of Figure 8.12. In the figure, the negative and positive electrodes are drawn as crystal structures comprising layers of electrode material. Lithium, drawn as small sphere, can be added to or removed from the space between layers. The lithium-ion cell used the method called **Intercalation** to store lithium charge-neutral atoms inside of the crystal structure.

Within the electrodes, lithium is stored as independent charge neutral atoms. Each lithium atom's valence electron is very loosely shared with neighboring atoms in the crystal structure. As such, the lithium is not tightly bonded in one place and is actually quite free to move around. Lithium enters and exits the surface of the electrodes, but diffuses within the layered open crystal structure to equalize the concentration of lithium within the vacant spaces of the electrode.

During discharge, lithium atoms at the surface of the negative electrode releases electrons - which travel through the external circuit - and become positive lithium ions,  $\text{Li}^+$  - which exit the crystal structure of the electrode and dissolve into the electrolyte. We can write  $\text{Li} \rightarrow \text{Li}^+ + e^-$ . Conversely, lithium ions proximate to the surface of the positive electrode receive electrons from the external circuit, and the resulting charge-neutral lithium atoms enter the crystal structure of the electrode. We can write  $\text{Li}^+ + e^- \rightarrow \text{Li}$ .

The process is completely reversible. Thus the lithium ions pass back and forth between the electrodes during charging and discharging. The intercalation mechanism is much gentler than an electromechanical reaction, so lithium-ion cells have much longer lives than other secondary

cells.

The majority of commercial lithium-ion cells use some form of graphite ( $C_6$ ) for the negative-electrode material. Graphite comprises multiple graphene layers, in which hexagonal  $C_6$  structures are tightly bonded together. The graphene layers are stacked loosely on top of each other, held together only by weak van der Waals forces; lithium intercalates between these layers. The maximum amount of lithium that can be stored in graphite is one atom of lithium per six atoms of carbon; the minimum amount is zero. Therefore, when talking about the degree of lithiation of a graphite electrode, we use notation  $Li_xC_6$ , where  $0 \leq x \leq 1$ . Clearly, when viewed at the atomic level, there is either a single lithium atom or no lithium atom at all for any given  $C_6$  site.

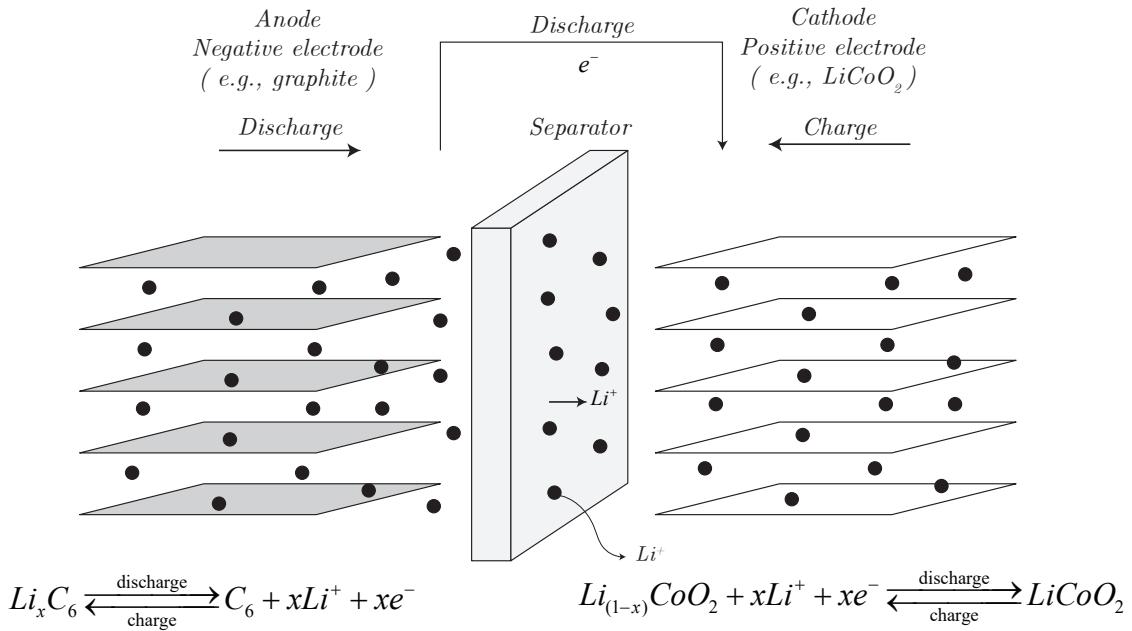


Figure 8.12: Schematic energy diagram of a lithium cell at open circuit.

But, when the entire electrode is considered, some fraction of the total number of  $C_6$  site is occupied and that fraction is the value of  $x$ . When the cell is charged the negative electrode is highly lithiated, and  $x$  is close to 1. When the cell is discharged, the negative electrode is largely depleted of lithium, and  $x$  is close to zero.

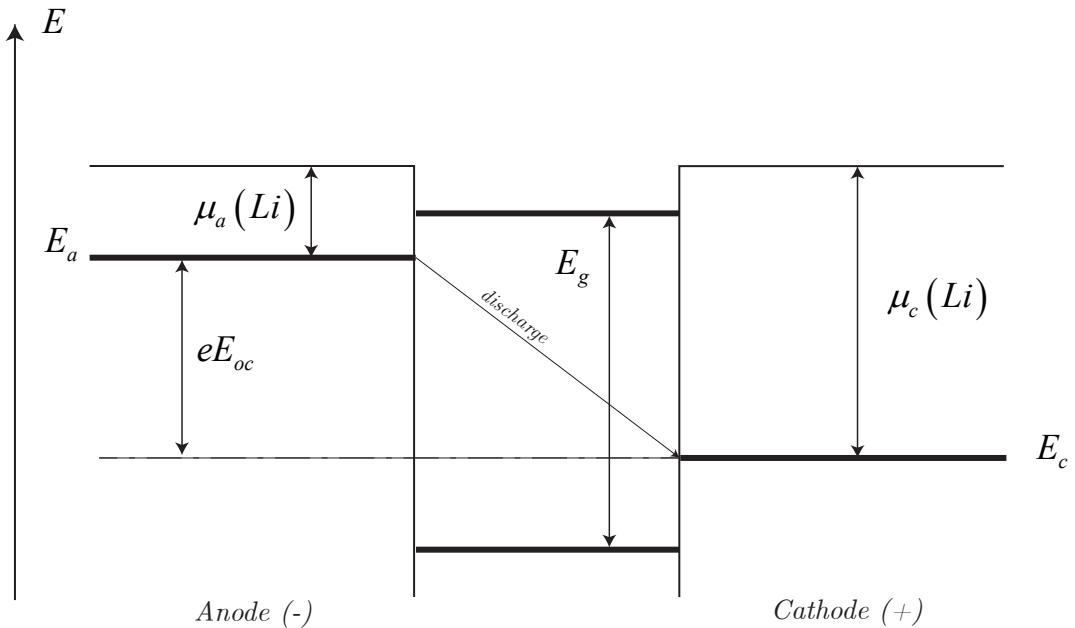


Figure 8.13: Schematic energy diagram of a lithium cell at open circuit.

#### 8.4.2 Dynamical Model of Lithium-ion Battery

In this section we investigate a simpler approach to modeling cell operation, which uses an electrical-circuit analogy to define a *behavioural* or *phenomenological* approximation to how a cell's voltage responds to different input-current stimuli. Such models are called *equivalent-circuit models*.

From simple observation of the lithium-ion cell we can see that the open circuit voltage (**OCV**) is function of the state of charge (**SOC**) of the cell. the state of charge shall be considered a unit-less quantity and is denote here as  $z(t)$ . To quantify the state of charge, we need to know how much charge a cell holds when is fully charged versus how much charge remains when it is fully discharged.<sup>†</sup> So, we define the *total charge capacity* - or more simply the *total capacity* - of a cell to be the total amount of charge removed when discharging a cell from  $z = 100\%$  to  $z = 0\%$ . Total (charge) capacity is usually measured in ampere-hours ( $\text{A h}^{-1}$ ) and is denoted by the symbol  $Q$ . The value of total capacity is a parameter of the cell model; that is, it is a constant that may differ from cell to cell. Total capacity is not a function of temperature or current, although the total capacity of a cell does tend to decrease very gradually as the cell ages dues to undesired parasitic chemical side reactions and structural breakdown of the cell's electrode materials<sup>‡</sup>

<sup>†</sup>A *fully discharged* cell still has charge in it, but in general is never discharged beyond a certain point because that can cause damage and possibly a safety concern.

<sup>‡</sup>The *total capacity* is different from a cell's *discharge capacity*. The latter is defined as the total amount of charge removed when discharging a cell at a constant rate from  $z = 100\%$  until the cell terminal voltage reaches some minimum cut-off voltage. This will occur before  $z = 0\%$  because real cells have internal resistance (and hence a voltage drop across the internal resistance). So, a cell's discharge capacities are always lower than its total capacity, unless discharge occurs at an infinitesimally slow rate.

We can model the state of charge using an ODE as follows

$$\frac{dz(t)}{dt} = -\frac{1}{Q}\eta(t)i(t) \quad (8.4.1)$$

where the sign of  $i(t)$  is positive when discharging. The current  $i(t)$  is measured in amperes,  $Q$  in coulombs (or A s). Both  $z(t)$  and  $\eta(t)$  are unitless.

The term  $\eta(t)$  is the *coulomb efficiency* or *charge efficiency* of the cell. We model  $\eta(t) = 1$  at time instants when the sign of current is positive (discharging), but  $\eta(t) \leq 1$  at time instants when the sign of current is negative (charging). When charging, most of the charge that passes through the cell participates in the desired chemical reactions, which do not increase the cell's state of charge (and often cause irreversible degradation to the cell as well).

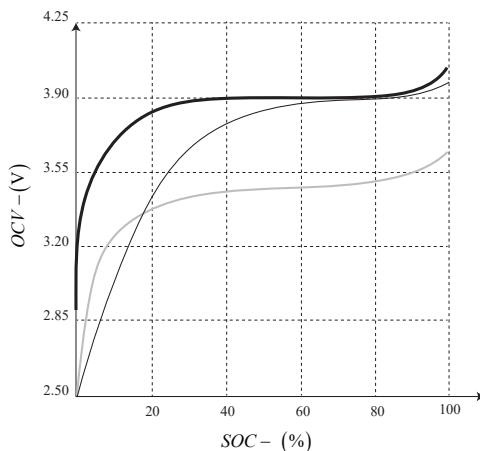


Figure 8.14: Open circuit voltage as a function of state of charge for different lithium-ion cell chemistries.

The **OCV** value for a cell are determined empirically at different **SOC** points. The values can be stored in a look of table.

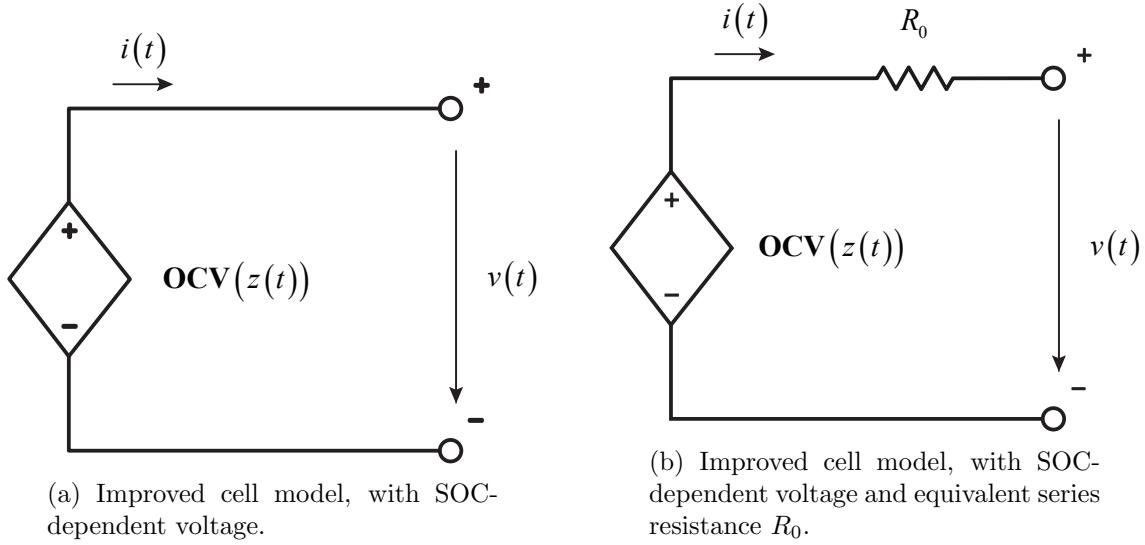


Figure 8.15: Equivalent circuit of the cell.

Experimentally we can observe that when the cell is subjected to a load (discharging) the terminal voltage drops below the open-circuit voltage and the terminal voltage rises above the open-circuit voltage when the cell is being charged. This phenomenon can be explained, in part, by placing a resistance in series with the controlled voltage source. The revised model is drawn in Figure 8.15b. The added circuit component represent the so-called *equivalent series resistance* (ESR) of the cell.

In the revised model, the state of the charge equation remains unchanged. However, we add a second equation to the model to describe how to compute the terminal voltage. In continuous time, we have

$$\begin{cases} \frac{dz(t)}{dt} = -\frac{1}{Q}\eta(t)i(t) \\ v(t) = \mathbf{OCV}(z(t)) - i(t)R_0 \end{cases} \quad (8.4.2)$$

with this model, we see that  $v(t) > \mathbf{OCV}(z(t))$  when  $i(t) < 0$  (i.e. when charging) and  $v(t) < \mathbf{OCV}(z(t))$  when  $i(t) > 0$  (i.e. when discharging).

Finally, we note that the cell's resistance is often a function of the cell's state of charge and is always a function of the cell's internal temperature. The fidelity of the model's predictions will be enhanced if these dependencies are taken into account in  $R_0$ .

**Polarization** refers to any deviation of the cell's terminal voltage away from open-circuit voltage due to a passage of current through the cell. In the equivalent circuit model we have modelled instantaneous polarization via the  $i(t)R_0$  term. Real cells have more complex behaviour, where the voltage polarization slowly develops over time as current is demanded from the cell and then slowly decay over time when the cell is allowed to rest.

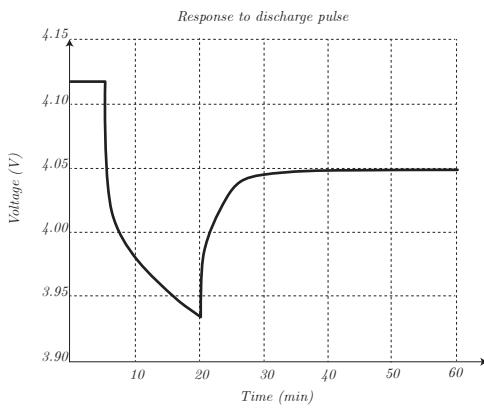


Figure 8.16: Behavior of the cell polarization, when a cell is subjected to a discharge pulse followed by a rest.

The phenomenon shown in Figure 8.16 and which we witness every time we use a battery-powered device is mainly due to the slow diffusion process of lithium in lithium-ion cell, so we will refer to this slowly changing in a circuit using one or more parallel resistor-capacitor sub-circuits. The equivalent circuit model become as follows

$$\begin{cases} \frac{dz(t)}{dt} = -\frac{1}{Q}\eta(t)i(t) \\ \frac{di_{R_1}(t)}{dt} = -\frac{1}{R_1C_1}(i_{R_1}(t) - i(t)) \\ v(t) = \mathbf{OCV}(z(t)) - i_{R_1}(t)R_1 - i(t)R_0 \end{cases} \quad (8.4.3)$$

where its representation in discrete time domain becomes as follows

$$\begin{cases} z[k+1] = z[k] - \frac{t_s}{Q}\eta[k]i[k] \\ i_{R_1}[k+1] = \exp\left(-\frac{t_s}{R_1C_1}\right)i_{R_1}[k] + \left[1 - \exp\left(-\frac{t_s}{R_1C_1}\right)\right]i[k] \\ v[k] = \mathbf{OCV}(z[k]) - R_1i_{R_1}[k] - R_0i[k] \end{cases} \quad (8.4.4)$$

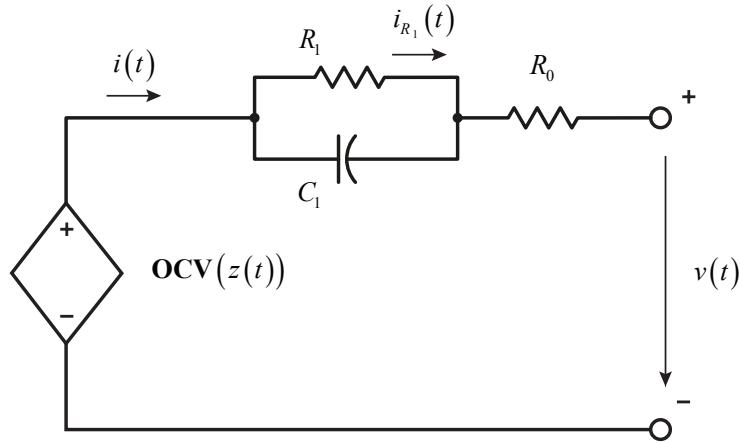


Figure 8.17: Circuit that models the diffusion voltage process.

Finally, we note that a cell's diffusion-voltage response is generally a function of the cell's state of charge and its internal temperature. If  $R_1$  and  $C_1$  are modeled as function of  $z(t)$  and  $T(t)$ , the model prediction can be improved.

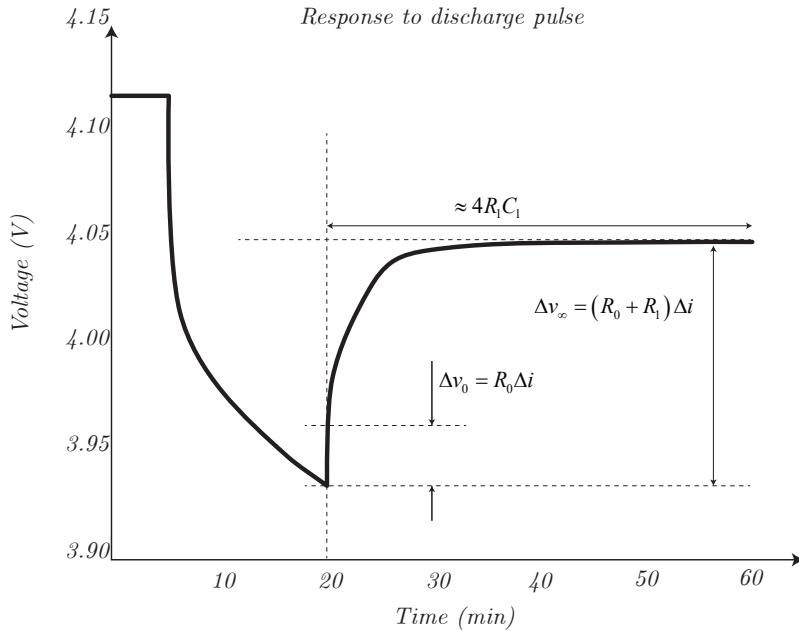


Figure 8.18: Measuring parameter values from a pulse response.

Now we are going to discuss about hysteresis behaviour if the cell. Considering a given **OCV** as function of the **SOC**, it can be observed that if we discharge a cell to 50% **SOC** and allow the cell to rest, the equilibrium voltage is lower than **OCV**. If we charge a cell to 50% **SOC** and allow the cell to rest, the equilibrium voltage is higher than **OCV**. These observations indicate that there is hysteresis in the cell terminal voltage.

Let  $h(z, t)$  be the dynamic hysteresis voltage as a function of **SOC** and time. We model the

change in hysteresis voltage as a function of change in **SOC** as follows

$$\frac{dh(z, t)}{dt} = \gamma \text{sign}(\dot{z}) [M(z, \dot{z}) - h(z, t)] \quad (8.4.5)$$

where  $M(z, \dot{z})$  is a function that gives the maximum polarization due to hysteresis as a function of **SOC** and the rate of charge of **SOC**. Specifically, we require that  $M(z, \dot{z})$  be positive for charge ( $\dot{z} > 0$ ) and negative for discharge ( $\dot{z} < 0$ ). The  $M(z, \dot{z}) - h(z, t)$  term is the differential equation states that the rate of change of hysteresis voltage is proportional to the distance of the present hysteresis value away from the major hysteresis loop, leading to a kind of exponential decay of voltage to the major loop. The term in front of this has a positive constant  $\gamma$  which tunes the rate of decay and  $\text{sign}(\dot{z})$ , which forces the equation to be stable both for charge and discharge.

To fit the differential equation for  $h(z, t)$  into our model, we must manipulate it to be a differential equation with respect to time, not with respect to **SOC**. We accomplish this by multiplying both sides of the equation by  $dz/dt$

$$\frac{dh(z, t)}{dz} \frac{dz}{dt} = \gamma \text{sign}(\dot{z}) [M(z, \dot{z}) - h(z, t)] \frac{dz}{dt} \quad (8.4.6)$$

we use the chain rule to write the left-hand side of the equation as  $dh(z, t)/dt$  and we substitute  $dz/dt = -\eta(t)i(t)/Q$  into the right hand side, noting that  $\dot{z}\text{sign}(\dot{z}) = |\dot{z}|$ . Thus,

$$\frac{dh(t)}{dt} = -\left| \frac{\eta(t)i(t)\gamma}{Q} \right| h(t) + \left| \frac{\eta(t)i(t)\gamma}{Q} \right| M(z, \dot{z}). \quad (8.4.7)$$

This may be converted into a difference equation for our discrete-time application assuming  $i(t)$  and  $M(z, \dot{z})$  are constant over the sample period.

$$h[k+1] = \exp\left(-\left|\frac{\eta[k]i[k]\gamma t_s}{Q}\right|\right)h[k] + \left[1 - \exp\left(-\left|\frac{\eta[k]i[k]\gamma t_s}{Q}\right|\right)\right]M(z, \dot{z}). \quad (8.4.8)$$

Note that this is a nonlinear time-varying system as the factors multiplying the state and input change with  $i[k]$ . The simplest representation is when  $M(z, \dot{z}) = -M\text{sign}(i[k])$ , when

$$h[k+1] = \exp\left(-\left|\frac{\eta[k]i[k]\gamma t_s}{Q}\right|\right)h[k] - \left[1 - \exp\left(-\left|\frac{\eta[k]i[k]\gamma t_s}{Q}\right|\right)\right]M\text{sign}(i[k]). \quad (8.4.9)$$

With this representation  $-M \leq h[k] \leq M$  at all times, and  $h[k]$  has units of volts. When attempting to find the model parameters, we will find it valuable to rewrite this in an equivalent but slightly different representation, which has unitless hysteresis state  $-1 \leq h[k] \leq 1$ ,

$$h[k+1] = \exp\left(-\left|\frac{\eta[k]i[k]\gamma t_s}{Q}\right|\right)h[k] - \left[1 - \exp\left(-\left|\frac{\eta[k]i[k]\gamma t_s}{Q}\right|\right)\right]\text{sign}(i[k]). \quad (8.4.10)$$

$$\text{Hysteresis voltage} = Mh[k]. \quad (8.4.11)$$

This makes the output equation linear in  $M$ .

In addition to the type of dynamic hysteresis that changes as **SOC** changes, we also can modeling an instantaneous change in hysteresis voltage when the sign of the current changes.

$$s[k] = \begin{cases} \text{sign}(i[k]) & |i[k]| > 0 \\ s[k-1], & \text{otherwise} \end{cases} \quad (8.4.12)$$

Then the instantaneous hysteresis is modelled as

$$\text{Instantaneous hysteresis voltage} = M_0 s[k]. \quad (8.4.13)$$

and overall hysteresis is

$$\text{Hysteresis voltage} = M_0 s[k] + Mh[k]. \quad (8.4.14)$$

### Enhanced self-correcting (ESC) model state equations

$$\left\{ \begin{array}{l} z[k+1] = z[k] - \frac{t_s}{Q} \eta[k] i[k] \\ i_{R_1}[k+1] = \exp\left(-\frac{t_s}{R_1 C_1}\right) i_{R_1}[k] + \left[1 - \exp\left(-\frac{t_s}{R_1 C_1}\right)\right] i[k] \\ h[k+1] = \exp\left(-\left|\frac{\eta[k] i[k] \gamma t_s}{Q}\right|\right) h[k] - \left[1 - \exp\left(-\left|\frac{\eta[k] i[k] \gamma t_s}{Q}\right|\right)\right] \text{sign}(i[k]) \\ s[k] = \begin{cases} \text{sign}(i[k]) & |i[k]| > 0 \\ s[k-1], & \text{otherwise} \end{cases} \\ v[k] = \mathbf{OCV}\left(z[k], T[k]\right) + M_0 s[k] + Mh[k] - R_1 i_{R_1}[k] - R_0 i[k] \end{array} \right. \quad (8.4.15)$$

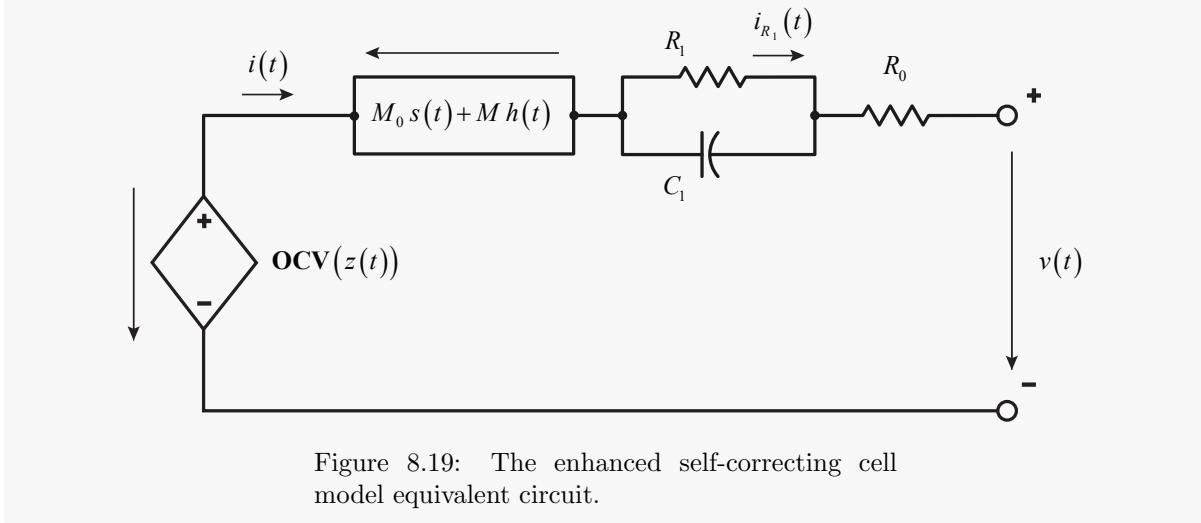


Figure 8.19: The enhanced self-correcting cell model equivalent circuit.

### 8.4.3 Open Circuit Voltage Model

To find an analytical model of the open circuit voltage of the lithium ion cell plays a fundamental role in the development of a battery management system. The open circuit voltage of a lithium ion cell is a quantity which can be measured with very high accuracy and the knowledge of the model (and its matching) permits to determine the state of charge of the battery.

The model of the  $\text{OCV}(z)$  as function of the  $z(t) = \text{SOC}(t)$  can be represented as follows

$$\text{OCV}(z) = E_{-1} \exp(-z\alpha) + E_0 + E_1 z + E_2 z^2 + E_3 z^3 + E_{\log} \log(1 - z) \quad (8.4.16)$$

where  $z$  belong to an open set as  $z \in (0, 1)$ .

For the sake of clarity we can consider the following case:

- $E_{-1} = -1.031 \text{ V}$
- $\alpha = 35$
- $E_0 = 3.685 \text{ V}$
- $E_1 = 0.015 \text{ V}$
- $E_2 = 0 \text{ V}$
- $E_3 = 0 \text{ V}$
- $E_{\log} = -0.05 \text{ V}$

which results in a  $\text{OCV}(z)$  curve as shown in Figure 8.20

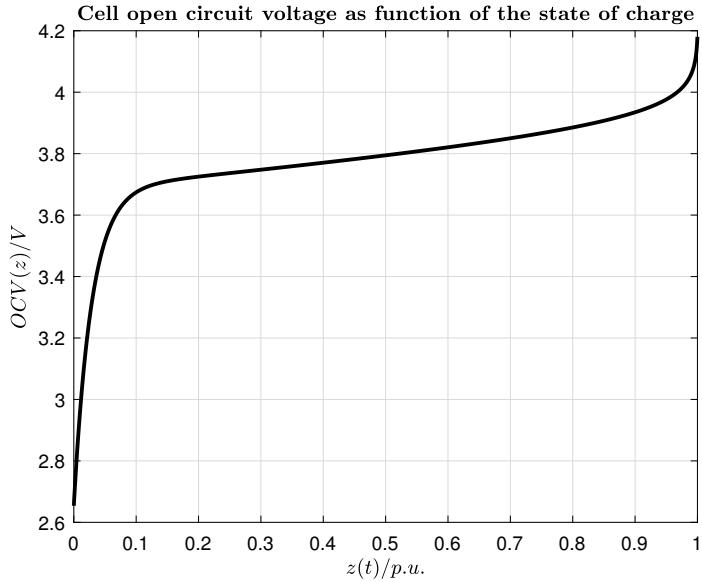


Figure 8.20: Lithium-ion battery open circuit cell as function of the state of charge.

## 8.5 LRCL Circuit Modelization

In this section a very wide used electrical network will be taken into account in order to write a state space model. The circuit under investigation is reported in Figure 8.21. This electrical network implements a second order filter with damping in series with the  $C$ . The location of the damping permits to damp the natural frequency of the  $LC$  circuit, and, in the same time permits to avoid excessive power losses which can be generated in the case the damping is located in series with the  $L$  inductance.

The damping resistor is the  $R_c$  term in Figure 8.21.

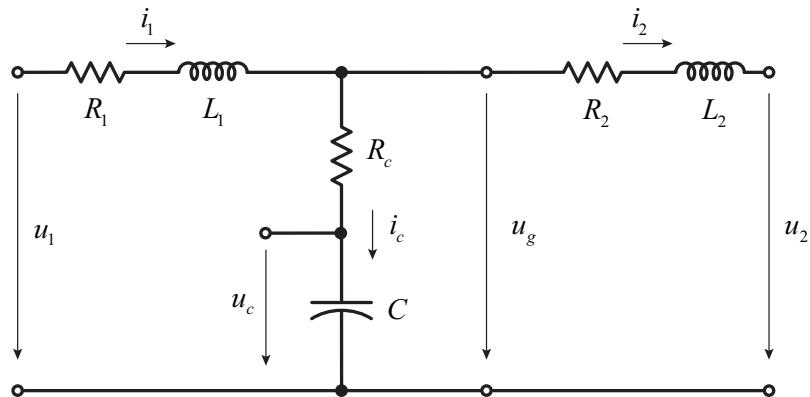


Figure 8.21: LRCL electrical schematic.

**Kirchhoff's equations<sup>†</sup>:**

$$u_1(t) - R_1 i_1(t) - L_1 \frac{di_1(t)}{dt} - u_g(t) = 0 \quad (8.5.1)$$

$$u_g(t) - R_2 i_2(t) - L_2 \frac{di_2(t)}{dt} - u_2(t) = 0 \quad (8.5.2)$$

$$i_1(t) - i_2(t) - i_c(t) = 0 \quad (8.5.3)$$

$$u_g(t) - R_c i_c(t) - u_c(t) = 0 \quad (8.5.4)$$

$$i_c(t) = C \frac{du_c(t)}{dt} \quad (8.5.5)$$

Eq. (8.5.4) and Eq. (8.5.3) can be merged and Eq. (8.5.4) can be written as follows

$$u_g(t) = R_c [i_1(t) - i_2(t)] + u_c(t) \quad (8.5.6)$$

Eq. (8.5.3) and Eq. (8.5.4)-Eq. (8.5.5) can be merged and can be written as follows

$$i_c(t) = i_1(t) - i_2(t) = C \frac{du_c(t)}{dt} \quad (8.5.7)$$

Combining Eq. (8.5.6)-Eq. (8.5.7) with Eq. (8.5.1)-Eq. (8.5.2) we obtain the following system

$$u_1(t) - R_1 i_1(t) - L_1 \frac{di_1(t)}{dt} - \{R_c [i_1(t) - i_2(t)] + u_c(t)\} = 0 \quad (8.5.8)$$

$$R_c [i_1(t) - i_2(t)] + u_c(t) - R_2 i_2(t) - L_2 \frac{di_2(t)}{dt} - u_2(t) = 0 \quad (8.5.9)$$

$$C \frac{du_c(t)}{dt} = i_1(t) - i_2(t) \quad (8.5.10)$$

or

$$L_1 \frac{di_1(t)}{dt} = - (R_1 + R_c) i_1(t) + R_c i_2(t) - u_c(t) + u_1(t) \quad (8.5.11)$$

$$L_2 \frac{di_2(t)}{dt} = R_c i_1(t) - (R_2 + R_c) i_2(t) + u_c(t) - u_2(t) \quad (8.5.12)$$

$$C \frac{du_c(t)}{dt} = i_1(t) - i_2(t) \quad (8.5.13)$$

---

<sup>†</sup>Physical relation between voltage variation and relative charge variation of an electrical capacitor:

$$CdV(t) = dQ(t) \Rightarrow C \frac{dV(t)}{dt} = i_c(t)$$

and

$$\frac{di_1(t)}{dt} = -\frac{R_1 + R_c}{L_1}i_1(t) + \frac{R_c}{L_1}i_2(t) - \frac{1}{L_1}u_c(t) + \frac{1}{L_1}u_1(t) \quad (8.5.14)$$

$$\frac{di_2(t)}{dt} = \frac{R_c}{L_2}i_1(t) - \frac{R_2 + R_c}{L_2}i_2(t) + \frac{1}{L_2}u_c(t) - \frac{1}{L_2}u_2(t) \quad (8.5.15)$$

$$\frac{du_c(t)}{dt} = \frac{1}{C}[i_1(t) - i_2(t)] \quad (8.5.16)$$

To write the system of Eq. (8.5.14)-Eq. (8.5.16) in a state space form it is necessary to define the input and the output of the system as follows

$$u(t) = u_1(t) \quad \text{input (scalar) of the system} \quad (8.5.17)$$

$$\vec{y}(t) = \begin{bmatrix} i_1(t) \\ i_2(t) \\ u_g(t) \end{bmatrix} \quad \text{output (vector) of the system} \quad (8.5.18)$$

the system contains also another term  $u_2(t)$  which can be treated as a unknown (or partially unknown) disturbance.

Obviously the state vector is represented by the vector

$$\vec{x} = \begin{bmatrix} i_1(t) \\ i_2(t) \\ u_c(t) \end{bmatrix} \quad (8.5.19)$$

The system Eq. (8.5.14)-Eq. (8.5.16) can be now written in a matrix form as follows

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u_1(t) + \tilde{\mathbf{E}}u_2(t) \quad (8.5.20)$$

$$\vec{y}(t) = \mathbf{C}\vec{x}(t) \quad (8.5.21)$$

where

$$\tilde{\mathbf{A}} = \begin{bmatrix} -\frac{R_1 + R_c}{L_1} & \frac{R_c}{L_1} & -\frac{1}{L_1} \\ \frac{R_c}{L_2} & -\frac{R_2 + R_c}{L_2} & \frac{1}{L_2} \\ \frac{1}{C} & -\frac{1}{C} & 0 \end{bmatrix} \quad (8.5.22)$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \end{bmatrix} \quad \tilde{\mathbf{E}} = \begin{bmatrix} 0 \\ -\frac{1}{L_2} \\ 0 \end{bmatrix} \quad (8.5.23)$$

and

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ R_c & -R_c & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \\ \mathbf{C}_3 \end{bmatrix} \quad (8.5.24)$$

The transfer function between the input  $u_1(t)$  and the output  $u_g(t)$  can be represented as follows

$$H_g(s) = \frac{U_g(s)}{U_1(s)} = \mathbf{C}_3 \left[ s\mathbf{I}_3 - \tilde{\mathbf{A}} \right]^{-1} \tilde{\mathbf{B}} \quad (8.5.25)$$

and result is in the form

$$H_g(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} \quad (8.5.26)$$

The transfer function between the input  $u_1(t)$  and the output  $i_2(t)$  can be represented as follows

$$H_2(s) = \frac{I_2(s)}{U_1(s)} = \mathbf{C}_2 \left[ s\mathbf{I}_3 - \tilde{\mathbf{A}} \right]^{-1} \tilde{\mathbf{B}} \quad (8.5.27)$$

and result is in the form

$$H_2(s) = \frac{b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} \quad (8.5.28)$$

The transfer function between the input  $u_1(t)$  and the output  $i_1(t)$  can be represented as follows

$$H_1(s) = \frac{I_1(s)}{U_1(s)} = \mathbf{C}_1 \left[ s\mathbf{I}_3 - \tilde{\mathbf{A}} \right]^{-1} \tilde{\mathbf{B}} \quad (8.5.29)$$

and result is in the form

$$H_1(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} \quad (8.5.30)$$

In the following, the  $H_2(s)$  transfer function of Eq. (8.5.28) will be taken into account. This transfer function is derived from the state space system

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}} \vec{x}(t) + \tilde{\mathbf{B}} u_1(t) \quad (8.5.31)$$

$$i_2(t) = \mathbf{C}_2 \vec{x}(t) \quad (8.5.32)$$

where the disturbance term  $u_2(t)$  has been omitted.

The coefficients relative to Eq. (8.5.28) are as follows

$$a_2 = \frac{L_1 R_c + L_2 R_c + L_1 R_2 + L_2 R_1}{L_1 L_2} \quad (8.5.33)$$

$$a_1 = \frac{L_1 + L_2 + R_2 R_c C + R_1 R_2 C + R_1 R_2 C}{L_1 L_2 C} \quad (8.5.34)$$

$$a_0 = \frac{R_1 + R_2}{L_1 L_2 C} \quad (8.5.35)$$

$$b_1 = \frac{R_c}{L_1 L_2} \quad (8.5.36)$$

$$b_0 = \frac{1}{L_1 L_2 C} \quad (8.5.37)$$

The transfer function of Eq. (8.5.28) can be rewritten in state space form using the controllable canonical form as follows

$$\dot{\vec{z}}(t) = \tilde{\mathbf{M}}\vec{z}(t) + \tilde{\mathbf{N}}u_1(t) \quad (8.5.38)$$

$$y(t) = i_2(t) = \mathbf{G}\vec{z}(t) \quad (8.5.39)$$

where

$$\tilde{\mathbf{M}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix}, \quad \tilde{\mathbf{N}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (8.5.40)$$

and

$$\mathbf{G} = \begin{bmatrix} b_0 & b_1 & 0 \end{bmatrix} \quad (8.5.41)$$

$$i_2^{(0)}(t) = \mathbf{G}\vec{z}(t) \quad (8.5.42)$$

$$i_2^{(1)}(t) = \mathbf{G}\tilde{\mathbf{M}}\vec{z}(t) + \mathbf{G}\tilde{\mathbf{N}}u_1(t) \quad (8.5.43)$$

$$i_2^{(2)}(t) = \mathbf{G}\tilde{\mathbf{M}}^2\vec{z}(t) + \mathbf{G}\tilde{\mathbf{M}}\tilde{\mathbf{N}}u_1(t) \quad (8.5.44)$$

## 8.6 Generic Thermal Model

### 8.6.1 Preliminary background

Let's try to consider the case where the equipment which is the source of heat power is placed in contact with an heat-sink which in turn is in contact with a fluid coolant with a given mass flow  $\dot{m}$ .

Because the flow in the heat-sink is completely enclosed, an energy balance may be applied to determine how the mean temperature  $T_m(x)$  varies with position  $x$  and how the total convection heat transfer  $P_{conv}$  is related to the difference in temperatures at the inlet and outlet. Consider the case study of Figure 8.22. Fluid moves at constant mass flow  $\dot{m}$ , and convection heat transfer occurs at the surface.

Applying conservation of energy<sup>§</sup> to the differential control volume of Figure 8.22, we obtain

$$dP_{conv} + \dot{m}(cT_m + pv) - \left[ \dot{m}(cT_m + pv) + \dot{m} \frac{d(cT_m + pv)}{dx} dx \right] = 0 \quad (8.6.2)$$

where  $v = 1/\rho$  is the specific volume, or

$$dP_{conv} = \dot{m}d(cT_m + pv) \quad (8.6.3)$$

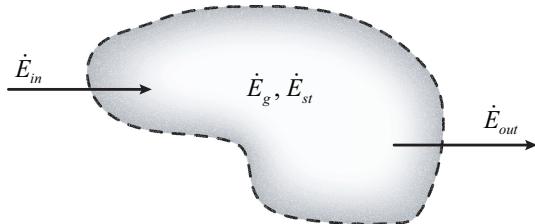
That is, the rate of convection heat transfer to the fluid must equal the rate at which the fluid thermal energy increases plus the net rate at which work is done in moving the fluid through the control volume. For *incompressible liquids*, where  $c_v = c_p = c$  and  $v$  is very small,  $d(pv)$  is generally much less than  $d(cT_m)$ . A special form of Eq. (8.6.3) relates to conditions for the *entire* length  $L$ . In particular integrating from the inlet  $i$  to the outlet  $o$ , it follows that

$$P_{conv} = \dot{m}c(T_{m,o} - T_{m,i}) \quad (8.6.4)$$

<sup>§</sup>

$$\dot{E}_{in} + \dot{E}_g - \dot{E}_{out} = \dot{E}_{st} \quad (8.6.1)$$

where  $E_g$  is the generated energy and  $E_{st}$  is the stored energy.



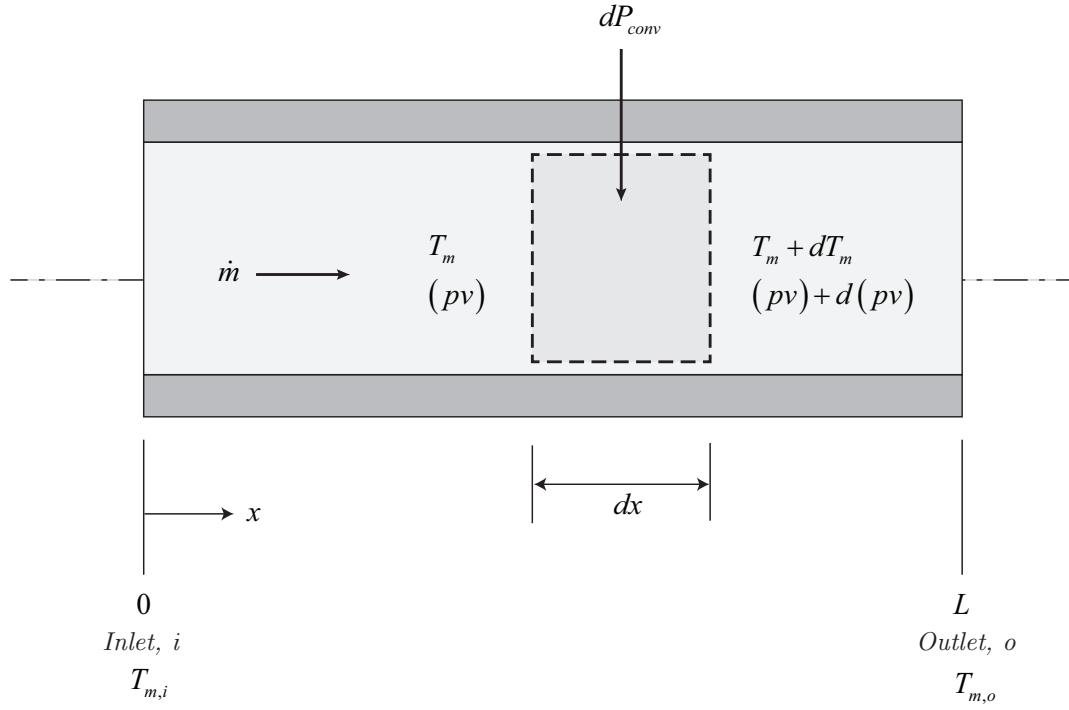


Figure 8.22: Control volume for internal flow in a heat-sink.

From the result of Eq. (8.6.4) and from the fundamental thermal model of Figure 8.23 we can now write the fundamental equation which represents the thermal equilibrium of the system, as follows

$$\sum_i m_i c_i \dot{T}_2 = P_{loss} - \alpha A(T_2 - T_a) - \dot{m}c(T_2 - T_1) \quad (8.6.5)$$

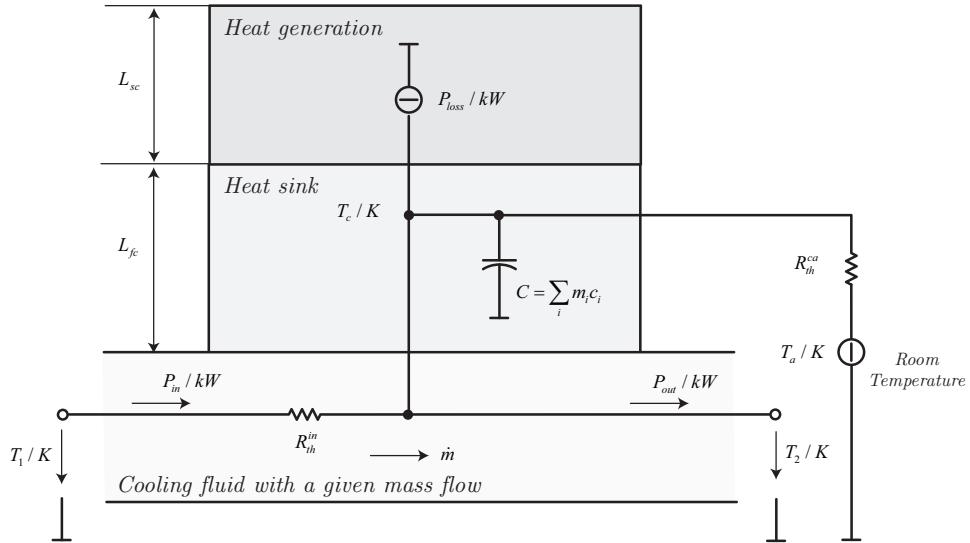


Figure 8.23: Fundamental thermal circuit used as benchmark.

where

$$R_{th}^{ca} = \frac{1}{\alpha A} \quad (8.6.6)$$

and

$$R_{th}^{in} = \frac{1}{\dot{m}c} \quad (8.6.7)$$

Figure 8.23 can also be represent as depicted in Figure 8.24.

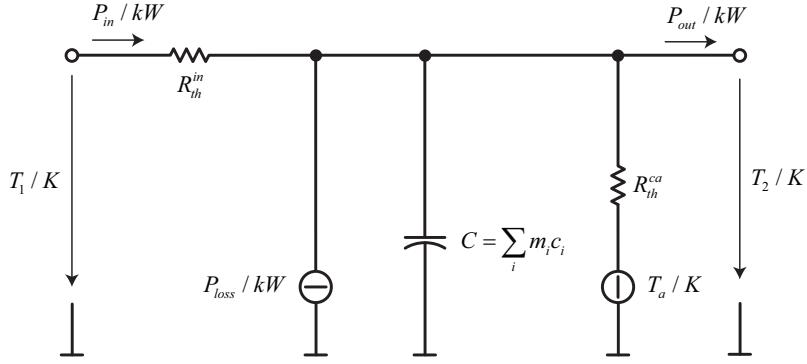


Figure 8.24: Schematic representation of the fundamental thermal circuit.

### 8.6.2 Equivalent mathematical model

To represent a generic thermal equipment we consider the following stack of components

- An heat generator. Any non ideal device can be represented as source of heat.
- An heat sink. The component which generates the heat power is placed in touch with an heat sink. The aim of the heat sink is to transfer the by a convective process the total amount of the generated heat. The transfer of the heat power from the generator to the heat sink is based on a conductive process. The heat sink is made by a material which, in general, has an high value of specific heat capacity, that means the heat sink create a temperature transient when it is involved into the transferring of the heat power.
- Fluid cooler. The heat sink is in general immersed into a force fluid cooler. The fluid can be air, water-glycol or oil. The coolant create a convective process transporting the heat from the heat-sink toward an external heat-exchanger.

Figure 8.25 shows a simple thermal structure of a generic cooling system, where

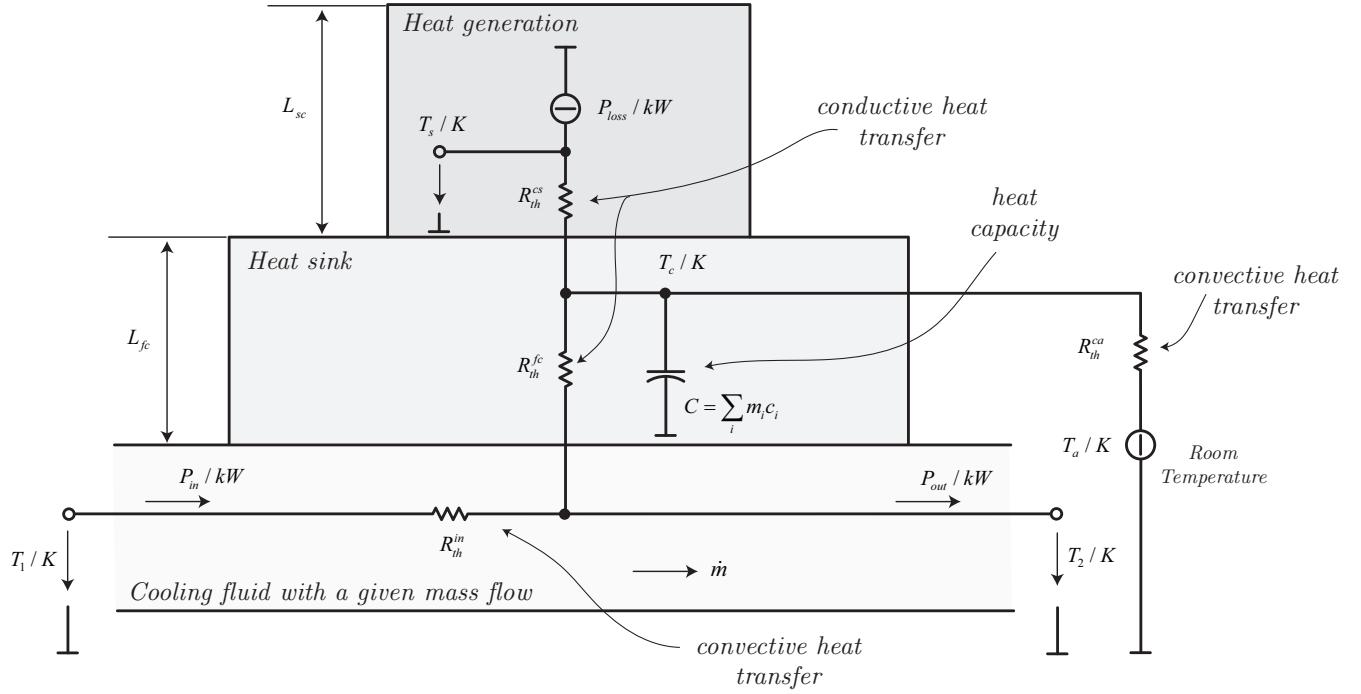


Figure 8.25: Generic thermal structure for a generic electromechanical system.

- $P_{\text{loss}}$  [ $\text{kW}$ ] it represents the source of the heat flow.
- $T_s$  [ $\text{K}$ ] it represents the temperature of the source of the heat flow.
- $T_c$  [ $\text{K}$ ] it represents the temperature of the heat-sink.
- $T_1$  [ $\text{K}$ ] it represents the temperature of the fluid coolant at the input of the system.
- $T_2$  [ $\text{K}$ ] it represents the temperature of the fluid coolant at the output of the system.
- $P_{\text{in}}$  [ $\text{kW}$ ] it represents the heat flow (transported by the fluid coolant) at the input of the system.
- $P_{\text{out}}$  [ $\text{kW}$ ] it represents the total heat flow which the fluid coolant is charged before to be exhaust from the system.
- $R_{\text{th}}^{\text{sc}}$  [ $\text{K W}^{-1}$ ] it represents the conductive thermal resistance between the source of the heat flow and the heat-sink.
- $R_{\text{th}}^{\text{ca}}$  [ $\text{K W}^{-1}$ ] it represents the convective thermal resistance between the source of the room.
- $R_{\text{th}}^{\text{fc}}$  [ $\text{K W}^{-1}$ ] it represents the conductive thermal resistance between the heat-sink and the fluid coolant.

- $R_{th}^{in} \left[ \text{K W}^{-1} \right]$  it represents the convective thermal resistance between the heat-sink and the fluid coolant.
- $C = \sum_i m_i c_i \left[ \text{J K}^{-1} \text{kg}^{-1} \right]$  it represents the total heat capacity of the heat-sink.

Figure 8.25 can be represented also as in Figure 8.26.

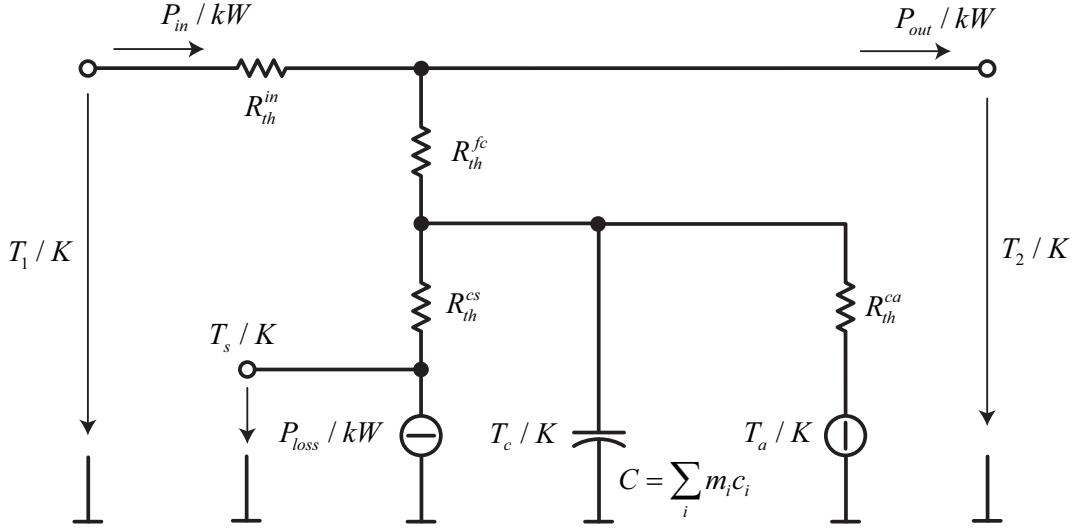


Figure 8.26: Generic thermal equivalent circuit for a generic electromechanical system.

where

$$R_{th}^{in} = \frac{1}{\dot{m}c} \quad (8.6.8)$$

$$R_{th}^{ca} = \frac{1}{\alpha_{ca} A_{ca}} \quad (8.6.9)$$

$$R_{th}^{fc} = \frac{L_{fc}}{\lambda_{fc} A_{fc}} \quad (8.6.10)$$

$$R_{th}^{cs} = \frac{L_{cs}}{\lambda_{cs} A_{cs}} \quad (8.6.11)$$

where

- $\lambda_{fc} \left[ \text{W K}^{-1} \text{m}^{-1} \right]$  it represents the thermal conductivity of the heat-sink.
- $\lambda_{cs} \left[ \text{W K}^{-1} \text{m}^{-1} \right]$  it represents the thermal conductivity of the heat generator.

The mathematical model of the system depicted in Figure 8.26 can be written as follows

$$\begin{cases} \sum_i m_i c_i \dot{T}_c = P_{loss} - \frac{1}{R_{th}^{ca}} (T_c - T_a) - \frac{1}{R_{th}^{fc}} (T_c - T_2) \\ T_2 = T_1 + (P_{out} - P_{in}) \end{cases} \quad (8.6.12)$$

The system depicted in Figure 8.26 concerns the equivalent thermal circuit of a given electromechanical system. From a wider point of view we can consider a system made by a cascade (series and parallel) of the fundamental thermal block of Figure 8.26 resulting in a system similar to Figure 8.27.

As shown in Figure 8.27, the common fluid coolant is driven by an external pump. The coolant is cooled down from an additional heat exchanger depicted by the thermal resistance  $R_{th}^{fe}$  and  $R_{th}^{ea}$ .

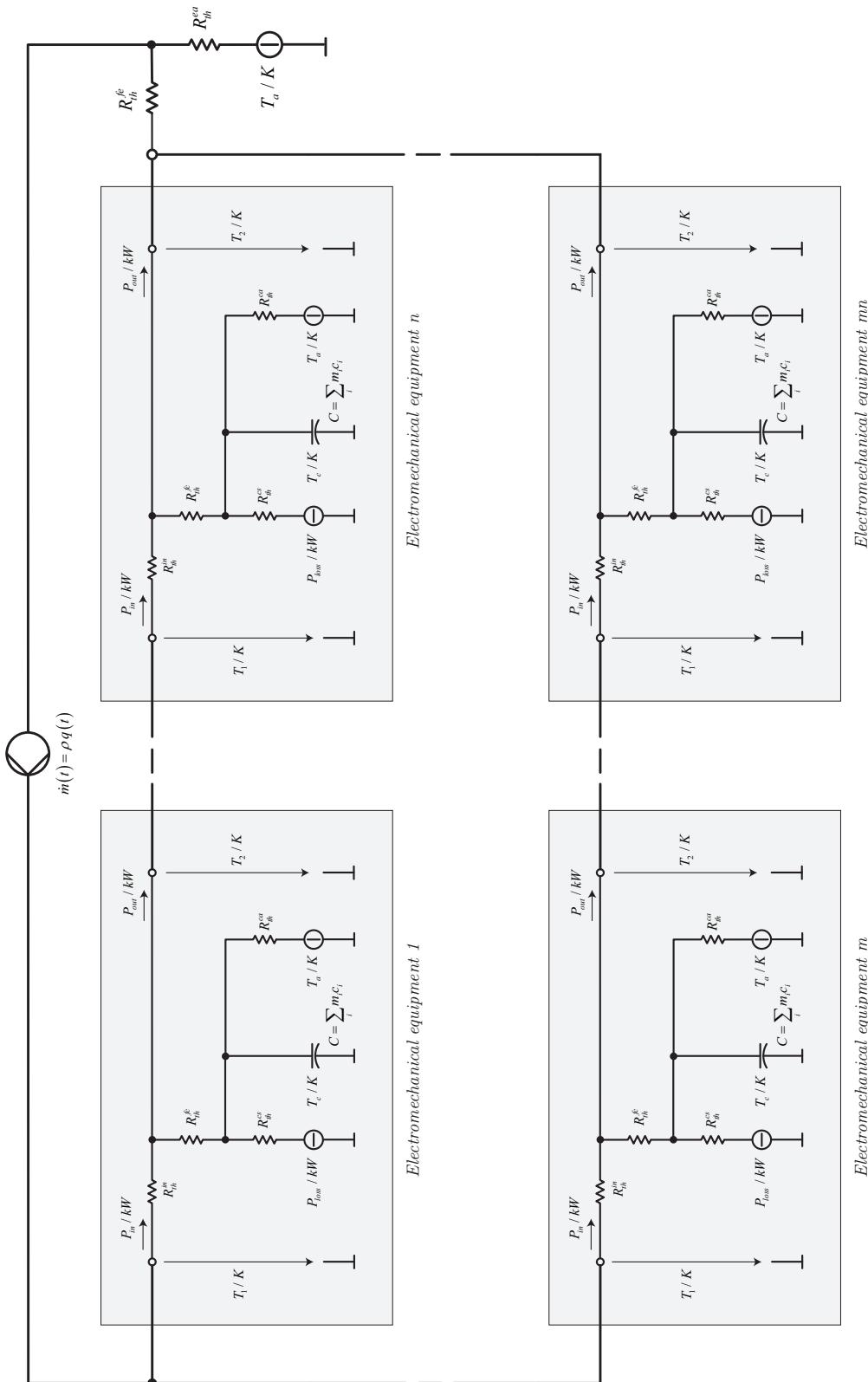


Figure 8.27: Generic thermal structure for a cascade of electromechanical systems - the flow of the thermal fluid is created by a hydraulic pump.

# Chapter 9

## Case Studies

### 9.1 Control of a Wheel with C-Code Implementation

#### 9.1.1 Rotor Speed and Load Torque Observer

Consider to control the rotational speed of a flywheel as reported in Figure 9.1 where the only available measured is the rotor position  $\theta(t)$  taken from an absolute encoder. The aim of the control problem, here proposed, is to design a state observer, which is able to estimate the rotational speed and the torque load (the external load).

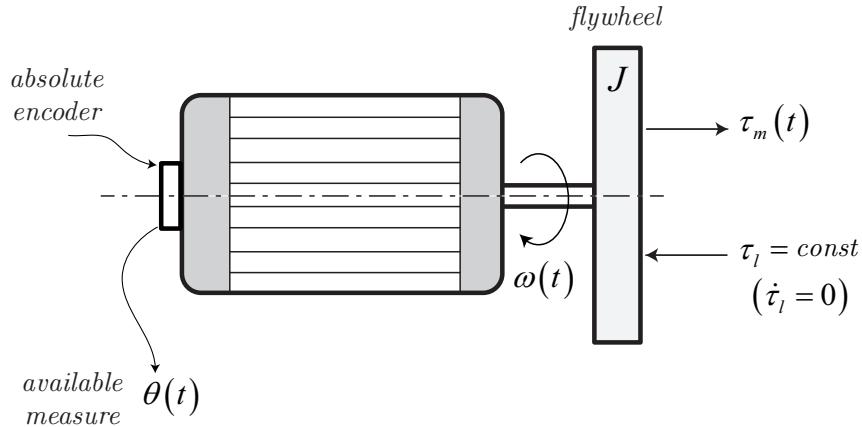


Figure 9.1: Description of the system.

The dynamical equation of the rotating flywheel driven by an input torque  $\tau_m$  and subjected to an external load  $\tau_l$  is given as follows

$$\ddot{\theta} = \frac{\tau_m - \tau_l}{J} \quad (9.1.1)$$

or

$$\begin{cases} \begin{bmatrix} \dot{\theta}(t) \\ \dot{\omega}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix} + \frac{1}{J} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tau_m(t) + \frac{1}{J} \begin{bmatrix} 0 \\ -1 \end{bmatrix} \tau_l(t) \\ y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix} \end{cases} \quad (9.1.2)$$

or in matrix form

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}\tau_m(t) + \tilde{\mathbf{E}}\tau_l \\ y(t) = \mathbf{C}\vec{x}(t) \end{cases} \quad (9.1.3)$$

The idea of the control problem is to design an observer which is able to estimate the external load. For this purpose we have to find a dynamical representation of the external load. The preliminary hypothesis concerning the nature of the torque load are:  $\tau_l = \text{const}$ . That means the dynamical equation which describe the nature of the external load can be written as follows

$$\dot{\tau}_l = 0 \quad (9.1.4)$$

we, now, create a new state vector  $\vec{x}_e(t) = [\theta(t) \ \omega(t) \ \tau_l(t)]^T$  and we describe the augmented system as follows

$$\begin{cases} \dot{\vec{x}}_e(t) = \tilde{\mathbf{A}}_e\vec{x}_e(t) + \tilde{\mathbf{B}}_e\tau_m(t) \\ y(t) = \mathbf{C}_e\vec{x}_e(t) \end{cases} \quad (9.1.5)$$

$$\begin{bmatrix} \dot{\theta}(t) \\ \dot{\omega}(t) \\ \dot{\tau}_l(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{J} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta(t) \\ \omega(t) \\ \tau_l(t) \end{bmatrix} + \frac{1}{J} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tau_m(t) \quad (9.1.6)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta(t) \\ \omega(t) \\ \tau_l(t) \end{bmatrix} \quad (9.1.7)$$

Where

$$\tilde{\mathbf{A}}_e = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{J} \\ 0 & 0 & 0 \end{bmatrix} \quad (9.1.8)$$

$$\tilde{\mathbf{B}}_e = \frac{1}{J} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (9.1.9)$$

$$\mathbf{C}_e = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \quad (9.1.10)$$

Where the transfer function of the plant is:

$$|C_e(sI - \tilde{A}_e)^{-1}\tilde{B}_e| = \frac{\begin{bmatrix} s & 1 & -\frac{1}{Js} \\ 0 & s & -\frac{1}{J} \\ 0 & 0 & s \end{bmatrix} \tilde{B}_e}{s^2} = \frac{1}{Js^2} \quad (9.1.11)$$

Let's evaluate the rank of the observability matrix:

$$\text{rank } \mathcal{O} = \text{rank} \begin{bmatrix} C_e \\ C_e \tilde{A}_e \\ C_e \tilde{A}_e^2 \end{bmatrix} = \text{rank} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{J} \end{bmatrix} = 3. \quad (9.1.12)$$

**Solution in the continuous time domain.** The observability matrix has full rank, hence the system is full state observable.

The state observer model become

$$\begin{bmatrix} \dot{\hat{\theta}}(t) \\ \dot{\hat{\omega}}(t) \\ \dot{\hat{\tau}}_l(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{J} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\theta}(t) \\ \hat{\omega}(t) \\ \hat{\tau}_l(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J} \\ 0 \end{bmatrix} \tau_m(t) + \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} [\theta(t) - \hat{\theta}(t)] \quad (9.1.13)$$

Let's calculate the state observer matrix gain, equaling the polynomial characteristic of the state observer with the desired one, as follows

$$|sI - (\tilde{A}_e - \tilde{L}_e C_e)| = (s - \mu_1)(s - \mu_2)(s - \mu_3) \quad (9.1.14)$$

which results in the following algebraic system equations

$$\begin{aligned} |sI - (\tilde{A}_e - \tilde{L}_e C_e)| &= \\ &= \begin{vmatrix} s + l_1 & -1 & 0 \\ l_2 & s & \frac{1}{J} \\ l_3 & 0 & s \end{vmatrix} = \\ &= s^3 + l_1 s^2 + l_2 s - \frac{l_3}{J} = \\ &= (s - \mu_1)(s - \mu_2)(s - \mu_3) = \\ &= s^3 - (\mu_1 + \mu_2 + \mu_3)s^2 + \\ &\quad + (\mu_1\mu_3 + \mu_1\mu_2 + \mu_2\mu_3)s + \mu_1\mu_2\mu_3 \end{aligned} \quad (9.1.15)$$

Hence

$$l_1 = -(\mu_1 + \mu_2 + \mu_3) \quad (9.1.16)$$

$$l_2 = \mu_1\mu_3 + \mu_1\mu_2 + \mu_2\mu_3 \quad (9.1.17)$$

$$l_3 = J\mu_1\mu_2\mu_3 \quad (9.1.18)$$

The choice of a set of  $\mu_1, \mu_2, \dots, \mu_n$  is, in many instances, not unique. As a general rule, however, the observer poles must be two to five times faster than the controller poles to make sure the observation error (estimation error) converges to zero quickly. This means that the observer estimation error decays two to five times faster than does the state vector  $\vec{x}$ . Such faster decay of the observer error compared with the desired dynamics make the controller poles dominate the system response. It is important to note that if sensor noise is considerable, we may choose the observer poles to be slower than two times the controller poles, so that the bandwidth of the system will become lower and smooth the noise. In this case the system response will be strongly influenced by the observer poles. If the observer poles are located to the right of the controller poles in the left half s-plane, the system response will be dominated by the observer poles rather than by control poles.

The final implementation can done applying a PI-control to the speed reference using the estimated speed  $\hat{\omega}(t)$  as feedback. The estimation of the torque load  $\hat{\tau}_l(t)$  can be used as feed-forward to compensate the external load, as shown in Figure 9.2.

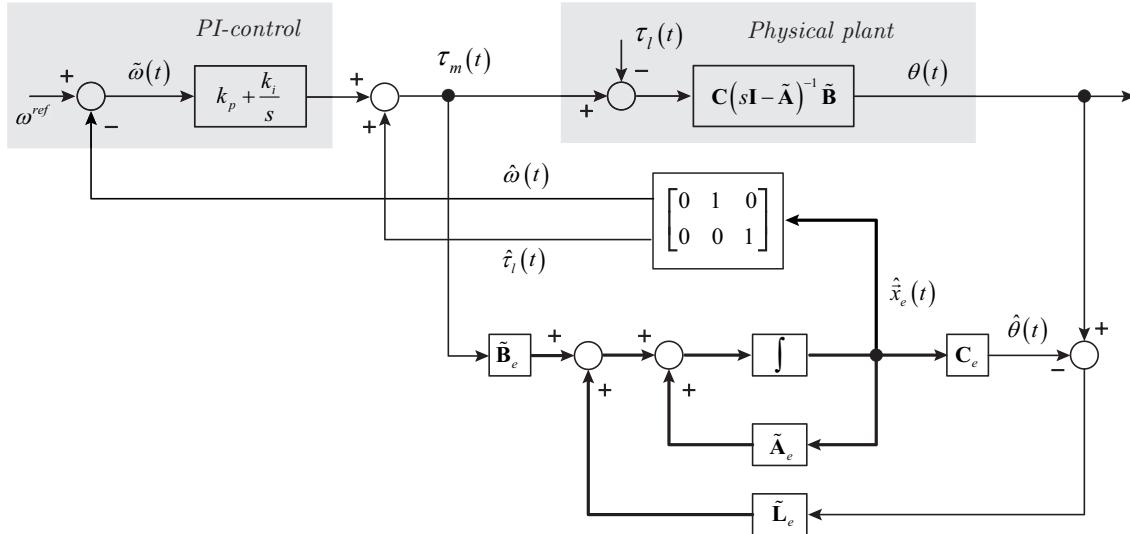


Figure 9.2: Control system implementation with state observer in continuous time domain.

**Solution in the discrete time domain.** Now we proceed implementing the control system in the discrete time domain. Representing of the control system in discrete time domain permits us to implement it into a microcontroller using e.g. C-language.

**Discretization.** Two control blocks has to be discretized: the PI-control and the state observer. Supposing a sampling time  $t_s = 1 \text{ ms}$  the PI-control can be written as follows

$$\begin{aligned} \text{PI-control } (z) &= k_p + \frac{k_i}{s} \Big|_{s=\frac{1-z^{-1}}{t_s}} \\ &= k_p + t_s \frac{k_i}{1 - z^{-1}} \end{aligned} \quad (9.1.19)$$

Let  $\tilde{\omega}(k) = \omega_{\text{ref}}(k) - \omega(k)$ , the PI-control, in discrete time domain, can be written as follows

$$\begin{cases} u_i(k) = u_i(k-1) + t_s k_i \tilde{\omega}(k) \\ u_{\text{pi}}(k) = k_p \tilde{\omega}(k) + u_i(k) \end{cases} \quad (9.1.20)$$

where  $u_{\text{pi}}(k)$  is the output of the PI-control at instant  $k$ .

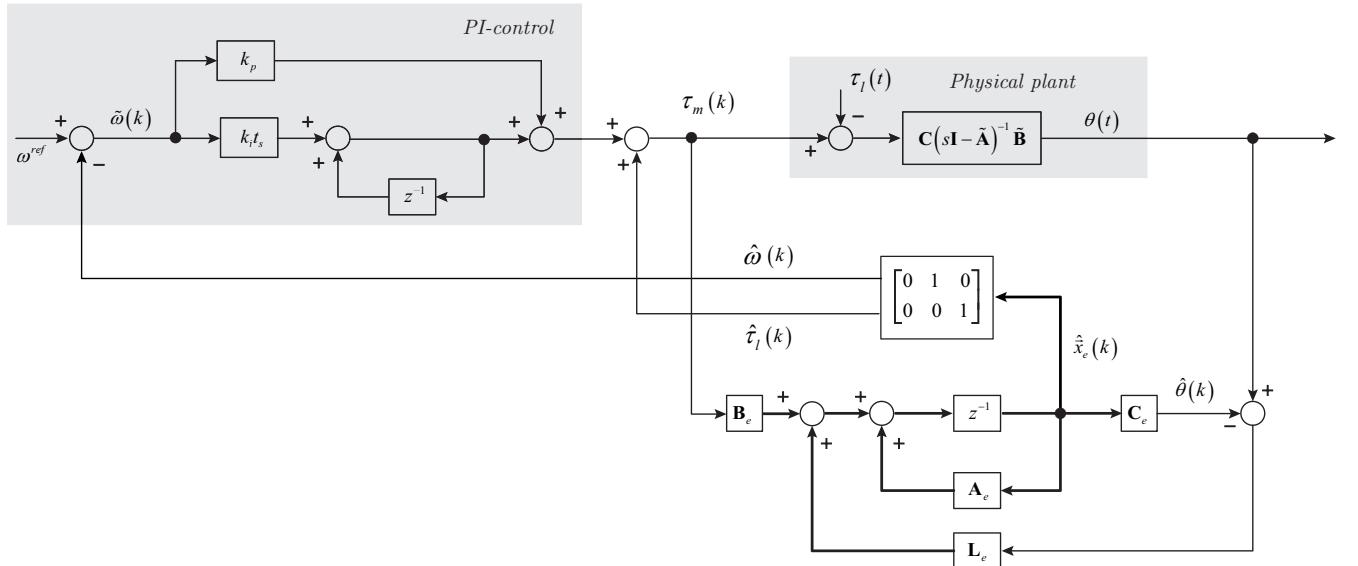


Figure 9.3: Control system implementation with state observer in discrete time domain.

To state observer can be discretized as follows

$$\begin{cases} \mathbf{A}_e \approx \mathbf{I} + \tilde{\mathbf{A}}_e t_s \\ \mathbf{B}_e \approx \tilde{\mathbf{B}}_e t_s \end{cases} \quad (9.1.21)$$

and the matrix  $\mathbf{L}_e$  can be calculated using pole-placement, which results in following system

$$\hat{x}_e(k+1) = \mathbf{A}_e \hat{x}_e(k) + \mathbf{B}_e \tau_m(k) + \mathbf{L}_e [\theta(k) - \hat{\theta}(k)] \quad (9.1.22)$$

### 9.1.2 C-Code Implementation Using C-caller

The control architectures we have seen up to now **MUST** be implemented into a microcontroller. The two parts of the control structures, PI-control and state observer shall be implemented into two different functions and they will lay in a master function which will "call" them properly. A proposed control structure will here shown, see also Figure 9.4.

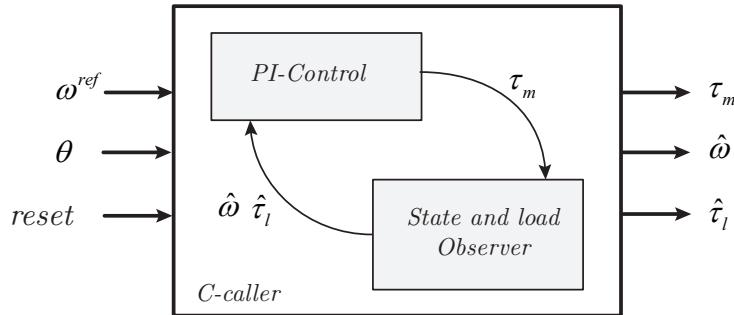


Figure 9.4: C-caller control architecture.

We start implementing the header file which contains the following parts

- Controller and state observer parameters defined as `#define`.
- Structure of the observer state variable here defined as type `SOBVCTRL`.
- Structure of the PI-controller here defined as type `PICTRL`.
- The state observer structure is made by the state variable  $\hat{x}_1$ ,  $\hat{x}_2$  and  $\hat{x}_3$  and by the output estimations  $\hat{\omega}$  and  $\hat{\tau}_l$ :

#### Variables for the State Observer

```

typedef struct sobvctrl_s {
    float x1_hat;
    float x2_hat;
    float x3_hat;
    float omega_hat;
    float tau_load_hat;
} sobvctrl_t;
#define SOBVCTRL sobvctrl_t

```

#### Variables for the PI control

```

typedef struct pictrl_s {
    float tau_m_i;
    float tau_m;
} pictrl_t;
#define PICTRL pictrl_t

```

- The list of main functions which are used in the model are as follows
  - State observer reset (or initialization). The state observer incorporate a system of differential equations hence its solutions concern in a integration process. The state of the integrator shall be initialized and properly reset.
  - PI-control reset, as above.
  - The state observer process.
  - The PI-control process.
  - The function `ctrlSimulink()` concerns the only simulink C-caller block. The variables `tau_m`, `omega_hat` and `tau_load_hat` have been defined as pointers to permit to be used as output variables in the C-caller block function.

`sfbctrl.h`

```
#ifndef _CTRL_
#define _CTRL_

#define ts 0.001

#define ae11 1.0
#define ae12 0.001
#define ae13 0.0

#define ae21 0.0
#define ae22 1.0
#define ae23 -0.001

#define ae31 0.0
#define ae32 0.0
#define ae33 1.0

#define be1 0.0
#define be2 0.001
#define be3 0.0

#define ce1 1.0
#define ce2 0.0
#define ce3 0.0

#define le1 2.052734441579463
#define le2 1265.733547807589
#define le3 -211635.1210315072

#define kp 31.41592653589793
#define ki 314.1592653589793
```

```

typedef struct sobvctrl_s {
    float x1_hat;
    float x2_hat;
    float x3_hat;
    float omega_hat;
    float tau_load_hat;
} sobvctrl_t;
#define SOBVCTRL sobvctrl_t

typedef struct pictrl_s {
    float tau_m_i;
    float tau_m;
} pictrl_t;
#define PICTRL pictrl_t

void sobvReset(SOBVCTRL *obv);

void piReset(PICTRL *pi);

void sobvProcess(SOBVCTRL *obv, float tau_m, float theta);

void piProcess(PICTRL *pi, float omega_ref, float omega_hat, float tau_load_hat);

void ctrlSimulink(float omega_ref, float theta, float reset,
                  float *tau_m, float *omega_hat, float *tau_load_hat);

#endif

```

Now we are going to describe the implemented source code. The function `piProcess()` implements the PI-control, applying the following equations

$$\begin{cases} \tilde{\omega}(k) = \omega_{\text{ref}}(k) - \omega(k) \\ \xi(k) = \xi(k-1) + t_s k_i \tilde{\omega}(k) \\ \tau_m(k) = k_p \tilde{\omega}(k) + \xi(k) + \hat{\tau}_l(k) \end{cases} \quad (9.1.23)$$

where the state variables  $\hat{\omega}(k)$  and  $\hat{\tau}_l(k)$  are estimated by the state observer by the function `sobvProcess()`.

The PI-control of Eq. (9.1.23) is implemented as follows

```

pi->tau_m = kp * (omega_ref - omega_hat) + pi->tau_m_i + tau_load_hat;
pi->tau_m_i = ki*ts*(omega_ref - omega_hat) + pi->tau_m_i;

```

The function `sobvProcess()` implements the whole state observer, applying the following

equations

$$\begin{cases} \hat{\omega}(k) = \hat{x}_2(k) \\ \hat{\tau}_l(k) = \hat{x}_3(k) \\ \hat{x}_1(k+1) = b_1\tau_m(k) + a_{11}\hat{x}_1(k) + a_{12}\hat{x}_2(k) + a_{13}\hat{x}_3(k) + l_1[\theta(k) - \hat{x}_1(k)] \\ \hat{x}_2(k+1) = b_2\tau_m(k) + a_{21}\hat{x}_1(k) + a_{22}\hat{x}_2(k) + a_{23}\hat{x}_3(k) + l_2[\theta(k) - \hat{x}_1(k)] \\ \hat{x}_3(k+1) = b_3\tau_m(k) + a_{31}\hat{x}_1(k) + a_{32}\hat{x}_2(k) + a_{33}\hat{x}_3(k) + l_3[\theta(k) - \hat{x}_1(k)] \end{cases} \quad (9.1.24)$$

The state observer of Eq. 9.1.24 is implemented as follows

```
sobv->omega_hat = sobv->x2_hat;
sobv->tau_load_hat = sobv->x3_hat;
error_obs = theta - sobv->x1_hat;
sobv->x1_hat = be1*tau_m + ae11*sobv->x1_hat + ae12*sobv->x2_hat
               + ae13*sobv->x3_hat + le1*error_obs;
sobv->x2_hat = be2*tau_m + ae21*sobv->x1_hat + ae22*sobv->x2_hat
               + ae23*sobv->x3_hat + le2*error_obs;
sobv->x3_hat = be3*tau_m + ae31*sobv->x1_hat + ae32*sobv->x2_hat
               + ae33*sobv->x3_hat + le3*error_obs;
```

The final function, `ctrlSimulink()` is deputed to manage the whole control. In this function we define the input and output of the C-caller block by the definition:

```
void ctrlSimulink(float omega_ref, float theta, float reset,
float *tau_m, float *omega_hat, float *tau_load_hat)
```

The quantity defined as pointers: `float *tau_m`, `float *omega_hat` and `float *tau_load_hat`, can be set as output quantities.

```
sfbctrl.c

#include "../include/speedctrl_with_sobv.h"
// -----
void sobvReset(SOBVCTRL *sobv)
{
    sobv->x1_hat = 0.0;
    sobv->x2_hat = 0.0;
    sobv->x3_hat = 0.0;
    sobv->omega_hat = 0.0;
    sobv->tau_load_hat = 0.0;
}
```

```

void piReset(PICTRL *pi)
{
    pi->tau_m_i = 0.0;
    pi->tau_m = 0.0;
}

void sobvProcess(SOBVCTRL *sobv, float tau_m, float theta)
{
    float error_obs = 0;

    sobv->omega_hat = sobv->x2_hat;

    sobv->tau_load_hat = sobv->x3_hat;

    error_obs = theta - sobv->x1_hat;

    sobv->x1_hat = be1*tau_m + ae11*sobv->x1_hat + ae12*sobv->x2_hat
                    + ae13*sobv->x3_hat + le1*error_obs;

    sobv->x2_hat = be2*tau_m + ae21*sobv->x1_hat + ae22*sobv->x2_hat
                    + ae23*sobv->x3_hat + le2*error_obs;

    sobv->x3_hat = be3*tau_m + ae31*sobv->x1_hat + ae32*sobv->x2_hat
                    + ae33*sobv->x3_hat + le3*error_obs;
}

void piProcess(PICTRL *pi, float omega_ref, float omega_hat,
               float tau_load_hat)
{
    pi->tau_m = kp * (omega_ref - omega_hat) + pi->tau_m_i + tau_load_hat;
    pi->tau_m_i = ki*ts*(omega_ref - omega_hat) + pi->tau_m_i;
}

void ctrlSimulink(float omega_ref, float theta, float reset,
                  float *tau_m, float *omega_hat, float *tau_load_hat)
{
    static SOBVCTRL sobv_instance;
    static PICTRL pi_instance;

    if (reset == 1)
    {
        sobvReset(&sobv_instance);
        piReset(&pi_instance);
    }

    *tau_m = pi_instance.tau_m;
    *omega_hat = sobv_instance.omega_hat;
    *tau_load_hat = sobv_instance.tau_load_hat;
}

```

```

piProcess(&pi_instance, omega_ref, *omega_hat, *tau_load_hat);
sobvProcess(&sobv_instance, *tau_m, theta);

}

```

In the `ctrlSimulink()` function (reported below) we defines the variables `sobv_instance` and `pi_instance` as `static`. The proprieties of a `static` variable are extremely important in the implementation of control systems. A `static` variable inside a function maintains its value among invocations and permits to enable the integration process.

```

ctrlSimulink()

void ctrlSimulink(float omega_ref, float theta, float reset,
                  float *tau_m, float *omega_hat, float *tau_load_hat)
{
    static SOBVCTRL sobv_instance;
    static PICTRL pi_instance;

    if (reset == 1)
    {
        sobvReset(&sobv_instance);
        piReset(&pi_instance);
    }

    *tau_m = pi_instance.tau_m;
    *omega_hat = sobv_instance.omega_hat;
    *tau_load_hat = sobv_instance.tau_load_hat;

    piProcess(&pi_instance, omega_ref, *omega_hat, *tau_load_hat);
    sobvProcess(&sobv_instance, *tau_m, theta);
}

```

### 9.1.3 Simulation results

tbd.  $\triangleleft$

## 9.2 Mass-Spring-Mass Control Problem

The mass-spring-mass model is a valid approximation of the flexible shaft. The aim of the control, we are going to develop, is to modify the natural frequency and damping of the original system in order to keep both rotor speeds ( $\omega_1$  and  $\omega_2$ ) as close as possible. Moreover a zero steady-state error from a given reference speed must be achieved despite a load disturbance applied to the system, see also Figure 9.5 where  $\tau_1$  is the torque actuated by the control and  $\tau_2$  a disturbance (or load torque).

### 9.2.1 Model Derivation

The model presents two masses ( $J_1, J_2$ ) connected by a torsional spring ( $k_\theta$ ) which include a friction term ( $b_\theta$ ). Both two masses are connected to the chassis by two virtual bearings modeled with a simple viscosity coefficient ( $b$ ).

Mathematical model can be derived using Euler-Lagrange equations as follows

$$\frac{d}{dt} \left( \frac{\partial E_{kin}}{\partial \dot{q}_1} \right) - \frac{\partial E_{kin}}{\partial q_1} + \frac{\partial E_{pot}}{\partial q_1} = \underbrace{Q_{diss}^{(12)} + Q_1}_{\text{not conservative force}} \quad (9.2.1)$$

$$\frac{d}{dt} \left( \frac{\partial E_{kin}}{\partial \dot{q}_2} \right) - \frac{\partial E_{kin}}{\partial q_2} + \frac{\partial E_{pot}}{\partial q_2} = \underbrace{Q_{diss}^{(21)} + Q_2}_{\text{not conservative force}} \quad (9.2.2)$$

Where  $Q_1$ ,  $Q_2$ ,  $Q_{diss}^{(12)}$  and  $Q_{diss}^{(21)}$  are the generalized forces, like magnetic force, unknown load, disturbance and friction or in general not conservative force\*.

Applying Eqs. (9.2.1) and (9.2.2) for the case

$$\begin{aligned} q_1 &= \theta_1 \\ q_2 &= \theta_2 \end{aligned} \quad (9.2.6)$$

The kinematic and potential energy can be written as follows

$$E_{kin} = \frac{1}{2} J_1 \dot{\theta}_1^2 + \frac{1}{2} J_2 \dot{\theta}_2^2 \quad (9.2.7)$$

$$E_{pot} = \frac{1}{2} k_\theta (\theta_1 - \theta_2)^2 \quad (9.2.8)$$

---

\*A conservative force is a force which is derived by a potential and satisfies to the following conditions:

$$\vec{\nabla} \times \vec{f} = \vec{0} \quad (9.2.3)$$

$$\oint_C \vec{f} \cdot d\vec{r} = 0 \quad (9.2.4)$$

$$\vec{f} = -\vec{\nabla}\phi \quad (9.2.5)$$

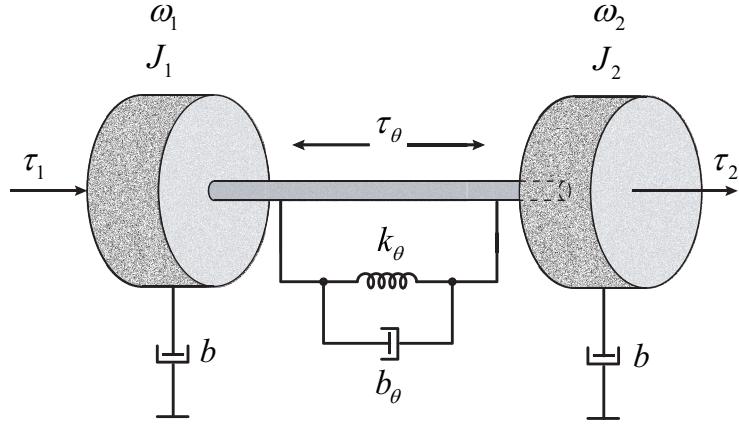


Figure 9.5: Simplified crankshaft model.

Applying the Lagrange equations we obtain the following motion equations

$$J_1 \ddot{\theta}_1 + k_\theta (\theta_1 - \theta_2) = -b\dot{\theta}_1 - b_\theta (\dot{\theta}_1 - \dot{\theta}_2) + \tau_1 \quad (9.2.9)$$

$$J_2 \ddot{\theta}_2 + k_\theta (\theta_1 - \theta_2) = -b\dot{\theta}_2 - b_\theta (\dot{\theta}_2 - \dot{\theta}_1) + \tau_2 \quad (9.2.10)$$

or

$$\begin{cases} \dot{\theta}_1 = \omega_1 \\ \dot{\omega}_1 = -\frac{k_\theta}{J_1}\theta_1 - \frac{b+b_\theta}{J_1}\omega_1 + \frac{k_\theta}{J_1}\theta_2 + \frac{b_\theta}{J_1}\omega_2 + \frac{1}{J_1}\tau_1 \\ \dot{\theta}_2 = \omega_2 \\ \dot{\omega}_2 = \frac{k_\theta}{J_2}\theta_1 + \frac{b_\theta}{J_2}\omega_1 - \frac{k_\theta}{J_2}\theta_2 - \frac{b+b_\theta}{J_2}\omega_2 + \frac{1}{J_2}\tau_2 \end{cases} \quad (9.2.11)$$

The state space representation become

$$\dot{\vec{z}} = \tilde{\mathbf{G}} \vec{z} + \tilde{\mathbf{H}} \tau_1 + \tilde{\mathbf{M}} \tau_2 \quad (9.2.12)$$

where  $\vec{z} = [\theta_1 \ \omega_1 \ \theta_2 \ \omega_2]^T$  and

$$\tilde{\mathbf{G}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_\theta}{J_1} & -\frac{b+b_\theta}{J_1} & \frac{k_\theta}{J_1} & \frac{b_\theta}{J_1} \\ 0 & 0 & 0 & 1 \\ \frac{k_\theta}{J_2} & +\frac{b_\theta}{J_2} & -\frac{k_\theta}{J_2} & -\frac{b+b_\theta}{J_2} \end{bmatrix} \quad (9.2.13)$$

and  $\tilde{\mathbf{H}} = [0 \ \frac{1}{J_1} \ 0 \ 0]^T$ ,  $\tilde{\mathbf{M}} = [0 \ 0 \ 0 \ -\frac{1}{J_2}]^T$ .

For our purpose, we want to change the state representation using a new state vector

$$\vec{x} = \mathbf{T}\vec{z} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \tau_\theta \end{bmatrix} \quad (9.2.14)$$

where<sup>†</sup>  $\mathbf{T}$  is given by

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ k_\theta & 0 & -k_\theta & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{T}^{-1} = \begin{bmatrix} 0 & 0 & \frac{1}{2k_\theta} \\ 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{2k_\theta} \\ 0 & 1 & 0 \end{bmatrix} \quad (9.2.17)$$

Basically, the linear transformation  $\mathbf{T}$  consists in a model reduction. The minimum representation of a dynamical system can be reached applying some matrix analysis results, but also considering the total number of energy storage parts of the system. In the case of mass-spring-mass model, we can assume that three components can store energy: the two inertial masses  $J_1$  and  $J_2$  as kinetic energy and the torsional spring and potential energy. That means the minimal representation of the mass-spring-mass model is made by three equations.

The new state space representation becomes

$$\dot{\vec{z}} = \tilde{\mathbf{G}}\vec{z} + \tilde{\mathbf{H}}\tau_1 + \tilde{\mathbf{M}}\tau_2 \quad (9.2.18)$$

$$\mathbf{T}^{-1}\dot{\vec{x}} = \tilde{\mathbf{G}}\mathbf{T}^{-1}\vec{x} + \tilde{\mathbf{H}}\tau_1 + \tilde{\mathbf{M}}\tau_2 \quad (9.2.19)$$

$$\dot{\vec{x}} = \mathbf{T}\tilde{\mathbf{G}}\mathbf{T}^{-1}\vec{x} + \mathbf{T}\tilde{\mathbf{H}}\tau_1 + \mathbf{T}\tilde{\mathbf{M}}\tau_2 \quad (9.2.20)$$

or

$$\dot{\vec{x}} = \tilde{\mathbf{A}}\vec{x} + \tilde{\mathbf{B}}\tau_1 + \tilde{\mathbf{E}}\tau_2 \quad (9.2.21)$$

---

<sup>†</sup>where

$$\tau_\theta = k_\theta(\theta_1 - \theta_2) \quad (9.2.15)$$

which can be written also as

$$\dot{\tau}_\theta = k_\theta(\omega_1 - \omega_2) \quad (9.2.16)$$

which describes the dynamic of the shaft torsional torque.

or

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\tau}_\theta \end{bmatrix} = \begin{bmatrix} -\frac{b+b_\theta}{J_1} & \frac{b_\theta}{J_1} & -\frac{1}{J_1} \\ \frac{b_\theta}{J_2} & -\frac{b+b_\theta}{J_2} & \frac{1}{J_2} \\ k_\theta & -k_\theta & 0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \tau_\theta \end{bmatrix} + \begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix} \tau_1 + \begin{bmatrix} 0 \\ \frac{1}{J_2} \\ 0 \end{bmatrix} \tau_2 \quad (9.2.22)$$

where

$$\vec{x} = [\omega_1 \ \omega_2 \ \tau_\theta]^T$$

$$\tilde{\mathbf{A}} = \mathbf{T} \tilde{\mathbf{G}} \mathbf{T}^{-1} = \begin{bmatrix} -\frac{b+b_\theta}{J_1} & \frac{b_\theta}{J_1} & -\frac{1}{J_1} \\ \frac{b_\theta}{J_2} & -\frac{b+b_\theta}{J_2} & \frac{1}{J_2} \\ k_\theta & -k_\theta & 0 \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \mathbf{T} \tilde{\mathbf{H}} = \begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix} \quad \tilde{\mathbf{E}} = \mathbf{T} \tilde{\mathbf{M}} = \begin{bmatrix} 0 \\ \frac{1}{J_2} \\ 0 \end{bmatrix}$$

**Consider the term  $\tau_2(t)$  as a disturbance (or load) which perturbs the system.**

For a more physical viewpoint the mass-spring-mass (or flexible shaft) dynamic equations can also be written as follows:

$$\begin{cases} J_1 \dot{\omega}_1 + b\omega_1 + b_\theta(\omega_1 - \omega_2) = \tau_1 - \tau_\theta \\ J_2 \dot{\omega}_2 + b\omega_2 + b_\theta(\omega_2 - \omega_1) = \tau_\theta + \tau_2 \\ \dot{\tau}_\theta = k_\theta(\omega_1 - \omega_2) \end{cases} \quad (9.2.23)$$

**Discretization.** Considering now, a discretization process at sampling time  $t_s$ , the equivalent discrete-time model can be approximated as follows:

$$\mathbf{A} \approx \mathbf{I} + \tilde{\mathbf{A}} t_s \quad (9.2.24)$$

$$\mathbf{B} \approx \tilde{\mathbf{B}} t_s \quad (9.2.25)$$

and

$$\mathbf{E} \approx \tilde{\mathbf{E}} t_s \quad (9.2.26)$$

Resulting in the following system:

$$\vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \tau_1(k) + \mathbf{E} \tau_2(k) \quad (9.2.27)$$

### 9.2.2 Control Design

In this section we explore some different control approaches based on *state-feedback with integrator*. The presence of the state-feedback control architecture imposes the use of a state observer. The state can not be measured. In particular we will compare the tracking and the estimation performance for two different state-observer scenario

- the estimator observer is based on the state vector

$$\hat{\vec{x}} = \begin{bmatrix} \hat{\omega}_1 & \hat{\omega}_2 & \hat{\tau}_\theta \end{bmatrix}^T \quad (9.2.28)$$

- the estimator observer include the unknown load torque  $\tau_2(t)$ , hence its state vector can be represented as follows

$$\hat{\vec{x}}_{le} = \begin{bmatrix} \hat{\omega}_1 & \hat{\omega}_2 & \hat{\tau}_\theta & \hat{\tau}_2 \end{bmatrix}^T \quad (9.2.29)$$

The extension of the state observer vector including the unknown disturbance  $\tau_2(t)$  imposes to write a differential equation which describes the dynamic of the disturbance (its nature). For this purpose we are going to consider the following dynamic representation of the load  $\tau_2(t)$

$$\dot{\tau}_2(t) = 0 \Rightarrow \ddot{\tau}_2(t) = 0 \quad (9.2.30)$$

or in discrete time domain

$$\tau_2(k+1) = \tau_2(k) \Rightarrow \hat{\tau}_2(k+1) = \hat{\tau}_2(k) \quad (9.2.31)$$

Eq. (9.2.30) says that the load disturbance  $\tau_2(t)$  change much slower than the natural dynamics of the mass-spring-mass system, that means we can consider the load as constant.

Both control system will be implemented in discrete time domain.

For model and control implementation data reported in Table 9.1 can be used.

$J_1 = 0.025 \text{ kg m}^2$	$J_2 = 0.025 \text{ kg m}^2$	$b_\theta = 0.02 \text{ N m s rad}^{-1}$	;
$k_\theta = 1 \text{ N m rad}^{-1}$	$t_s = 1 \text{ ms}$	$b = 0.02 \text{ N m s rad}^{-1}$	

Table 9.1: Flexible shaft model parameters.

Before to start with the development of the control, a short description of the properties of the system shall be considered. Taking a look into the transfer function or into the characteristic polynomial of the system we can see it contains an harmonic oscillator. This is due to the presence of the torsional spring and the two inertia masses. The natural frequency of the harmonic oscillator can be analytically determined as follows

setting  $b = 0$  and  $b_\theta = 0$  we obtain the following characteristic polynomial

$$|s\mathbf{I} - \tilde{\mathbf{A}}| = \begin{vmatrix} s & 0 & 1/J_1 \\ 0 & s & -1/J_2 \\ -k_\theta & k_\theta & s \end{vmatrix} = s \left[ s^2 + k_\theta \frac{J_1 + J_2}{J_1 J_2} \right] \quad (9.2.32)$$

where the characteristic roots are

$$\begin{cases} s_1 = 0 \\ s_2 = j \sqrt{k_\theta \frac{J_1 + J_2}{J_1 J_2}} \\ s_3 = -j \sqrt{k_\theta \frac{J_1 + J_2}{J_1 J_2}} \end{cases} \quad (9.2.33)$$

the roots  $s_{2,3}$  are the roots of the harmonic oscillator.

Figure 9.6 shows the effect of the inner harmonic oscillator when a impulse torque is applied to the system. The natural behaviour of the system is to oscillate until the internal friction dissipate the initial potential energy stored into the spring.

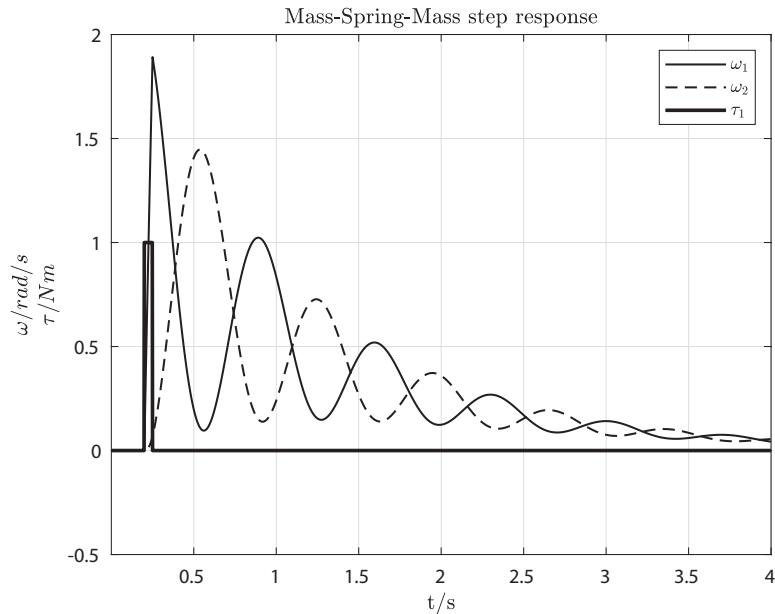


Figure 9.6: Open loop response of the system applying an impulse to the torque input  $\tau_1$ . The natural frequency of the system is  $f_0 \approx \frac{1}{2\pi} \sqrt{k_\theta \frac{J_1 + J_2}{J_1 J_2}} = 1.424\text{Hz}$

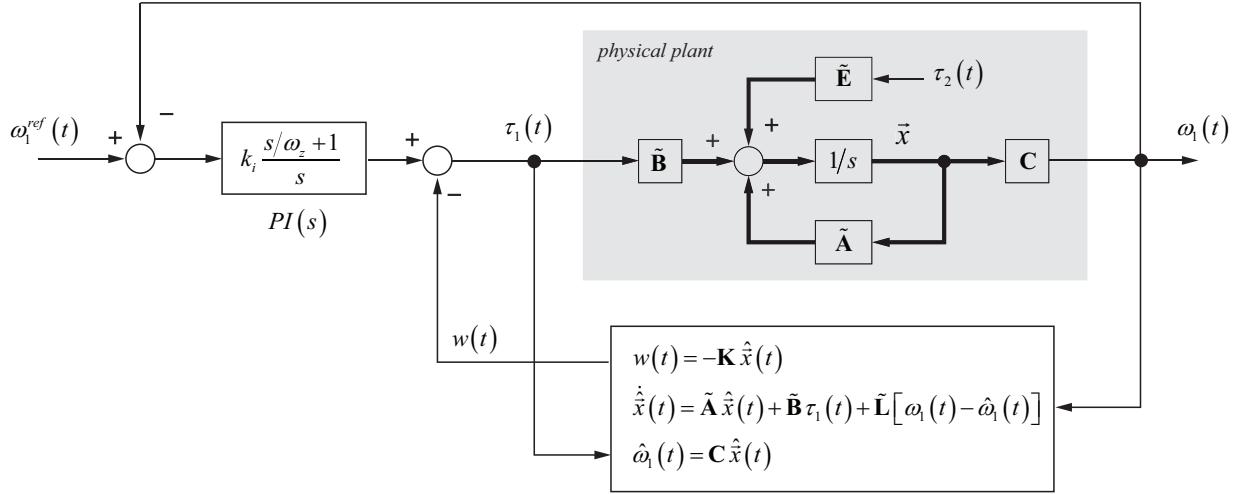


Figure 9.7: A generic representation of the Proportional Integrator + State Feedback control architecture. This structure can be used when inner properties of the plant need to be modified.

The control objective is to create a controller which is able to keep both rotor speeds  $\omega_1(t)$  and  $\omega_2(t)$  close to each other, and in parallel the controller should have the ability to track a constant rotor speed set-point with zero error steady state performance

$$\tilde{\omega}_1(t) = \omega_1^{ref}(t) - \omega_1(t) \rightarrow 0 \quad \text{as } t \rightarrow \infty$$

The use of the state feedback permits to modify this natural behaviour and force the system to move as without the internal oscillator. By the pole placement design the state feedback system will behave as imposed via pole placement.

The requirement of tracking with zero error with constant reference in steady state condition can be reached introducing an integrator part in the forward loop which take into consideration the tracking error  $\tilde{\omega}_1 = \omega_1^{ref}(t) - \omega_1(t)$ , the whole control structure is called *State-Feedback with Integrator* or *Servo*, see also Figure 9.8.

Another approach would also be to design, in almost independent way, a state feedback approach and a proportional integral control. The state feedback via pole placement approach modify the nature of the plant while the proportional integral controller performs the tracking problem control, , see also Figure 9.7.

In the following, we will take into account the control structure *State-Feedback with Integrator* and we will show a method for a global design of both state feedback part and the integral part as one single pole placement strategy.

### 9.2.3 Integral control without load estimation

In this section we design a controller based on architecture of Figure 9.8. The design of the control consist of

- computing of the discrete time domain matrices  $\mathbf{A}$  and  $\mathbf{B}$  from matrices  $\tilde{\mathbf{A}}$  and  $\tilde{\mathbf{B}}$ .

- computing of the state feedback matrix gain  $\mathbf{K}$  and of the integral term  $k_i$  via a special pole placement.
- computing of the state observer matrix gain  $\mathbf{L}$  via pole placement.

Considering the continuous time domain plant

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}} \vec{x}(t) + \tilde{\mathbf{B}} \tau_1(t) \quad (9.2.34)$$

$$\vec{y}(t) = \mathbf{C} \vec{x}(t) \quad (9.2.35)$$

by the computation<sup>‡</sup>

$$\begin{aligned} \tilde{\mathbf{A}} &\rightarrow \mathbf{A} \approx \mathbf{I} + t_s \tilde{\mathbf{A}} \\ \tilde{\mathbf{B}} &\rightarrow \mathbf{B} \approx t_s \tilde{\mathbf{B}} \end{aligned} \quad (9.2.38)$$

we obtain the discrete time domain model as in Eqs. (9.2.39) and. (9.2.40)

$$\vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} \tau_1(k) \quad (9.2.39)$$

$$\vec{y}(k) = \mathbf{C} \vec{x}(k) \quad (9.2.40)$$

where  $\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$ , that means,  $\omega_1(t)$  if the feedback measure of the state vector  $\vec{x} = \begin{bmatrix} \omega_1 & \omega_2 & \tau_\theta \end{bmatrix}^T$ .

Eqs. (9.2.39) and (9.2.40) have been derived in order to implement the state-observer - of Eqs. (9.2.44) and (9.2.45) - required to implement the state feedback control.

As already mentioned, Figure 9.8. shows the proposed control architecture.

---

<sup>‡</sup>The order of the Taylor approximation can be incremented properly according to the criticality of the discretization, as follows

$$\tilde{\mathbf{A}} \rightarrow \mathbf{A} = \mathbf{I} + t_s \tilde{\mathbf{A}} + \frac{t_s^2}{2} \tilde{\mathbf{A}}^2 + \frac{t_s^3}{6} \tilde{\mathbf{A}}^3 + \dots \quad (9.2.36)$$

$$\tilde{\mathbf{B}} \rightarrow \mathbf{B} = t_s \tilde{\mathbf{B}} + \frac{t_s^2}{2} \tilde{\mathbf{A}} \tilde{\mathbf{B}} + \frac{t_s^3}{6} \tilde{\mathbf{A}}^2 \tilde{\mathbf{B}} + \dots \quad (9.2.37)$$

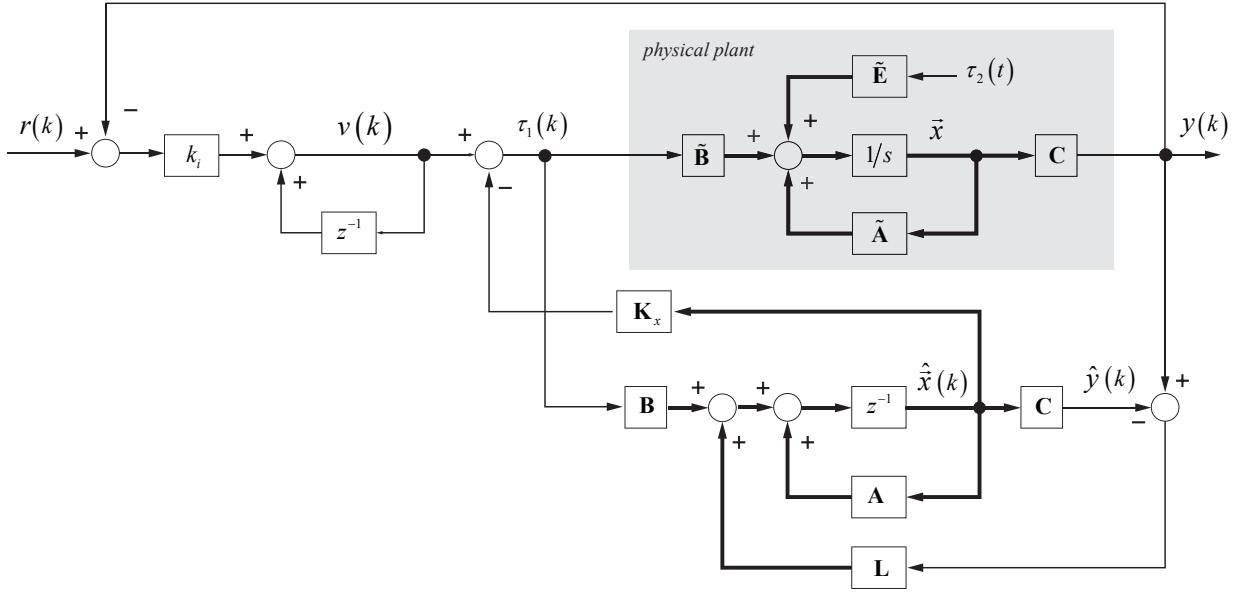


Figure 9.8: Description of the control strategy in discrete-time domain without  $\tau_2(t)$  estimator.

The computing of the matrix  $[\mathbf{K}_x \ k_i]$  is reduced as follows

$$[\mathbf{K}_x \ k_i] = \left\{ \mathbf{K}' + \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \right\} \begin{bmatrix} \mathbf{A} - \mathbf{I}_3 & \mathbf{B} \\ \mathbf{C}\mathbf{A} & \mathbf{C}\mathbf{B} \end{bmatrix}^{-1} \quad (9.2.41)$$

where  $\mathbf{K}'$  can be computed by Ackermann's formula using the system  $(\mathbf{A}_e, \mathbf{B}_e)$

$$\mathbf{K}' = \text{acker}(\mathbf{A}_e, \mathbf{B}_e, \text{poles\_sf\_z}) \quad (9.2.42)$$

where

$$\mathbf{A}_e = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \vec{0} & 0 \end{bmatrix} \quad \mathbf{B}_e = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T \quad (9.2.43)$$

Regarding the observer block we can consider the following system equation:

$$\hat{\vec{x}}(k+1) = \mathbf{A} \hat{\vec{x}}(k) + \mathbf{B} \tau_1(k) + \mathbf{L} (\vec{y}(k) - \hat{y}(k)) \quad (9.2.44)$$

$$\hat{y}(k) = \mathbf{C} \hat{\vec{x}}(k) \quad (9.2.45)$$

Where  $\hat{y}(k) = \hat{\omega}_1(k)$ .

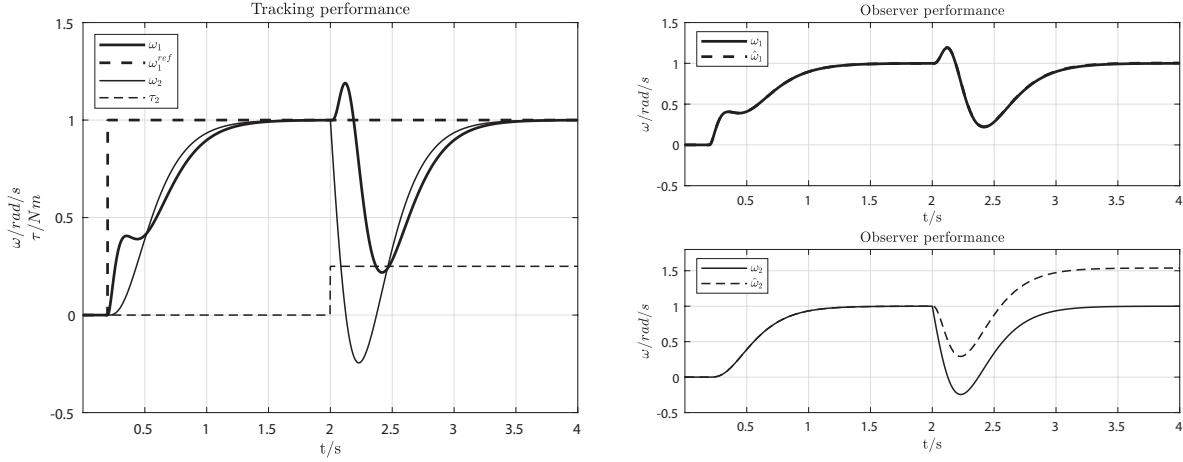
The matrix  $\mathbf{L}$  is calculated applying the pole placement method to the system  $(\mathbf{A}, \mathbf{C})$ .

$$\mathbf{L}_d = \text{acker}(\mathbf{A}_d, \mathbf{C}, \text{poles\_sobv\_z}) \quad (9.2.46)$$

### 9.2.3.1 Simulation results

The following simulation results consider at time  $t_{load} = 2$  s a load step of  $\tau_2 = 0.25 \text{ N m}$ .

As depicted in Figure 9.9b it can be seen the steady state error of the estimation of the state variable  $\omega_2$  due to the missed modelization of the load ( $\tau_2$ ) dynamic.



(a) Step response of the state feed-back control with observer.  $\tau_2$  is the external load considered in this case as a perturbation.

(b) Performance of the observer block control. The quantity  $\hat{\omega}_2$  is affected by a steady state error as effect of the load disturbance.

Figure 9.9: Simulation results.

### 9.2.4 Integral control with load estimation

As previously seen the estimation of the state variable  $\omega_2$  is affected by the load which perturbs the system. A possibility to compensate the load disturbance is to estimate it including its dynamic into the observer model. To this aim we consider a load subject to the following dynamic:  $\dot{\tau}_2 = 0$ . That means, in case of constant load the observer is able to estimate the amplitude of the disturbance with zero steady state error. The selected control architecture is shown in Figure 9.10. The extended observer model can be described as follows

$$\begin{bmatrix} \dot{\hat{\omega}}_1 \\ \dot{\hat{\omega}}_2 \\ \dot{\hat{\tau}}_\theta \\ \dot{\hat{\tau}}_2 \end{bmatrix} = \begin{bmatrix} -\frac{b+b_\theta}{J_1} & \frac{b_\theta}{J_1} & -\frac{1}{J_1} & 0 \\ \frac{b_\theta}{J_2} & -\frac{b+b_\theta}{J_2} & \frac{1}{J_2} & \frac{1}{J_2} \\ k_\theta & -k_\theta & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\omega}_1 \\ \hat{\omega}_2 \\ \hat{\tau}_\theta \\ \hat{\tau}_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tau_1 + \begin{bmatrix} l_1^d \\ l_2^d \\ l_3^d \\ l_4^d \end{bmatrix} (\omega_1 - \hat{\omega}_1) \quad (9.2.47)$$

or

$$\begin{aligned} \hat{x}_{le}(t) &= \tilde{\mathbf{A}}_{le} \hat{x}_{le}(t) + \tilde{\mathbf{B}}_{le} \tau_1(t) + \tilde{\mathbf{L}}_{le} (\vec{y}(t) - \hat{\vec{y}}(t)) \\ \hat{\vec{y}}(t) &= \mathbf{C}_{le} \hat{x}_{le}(t) \end{aligned} \quad (9.2.48)$$

where  $\vec{x}_{le} = [\omega_1 \ \omega_2 \ \tau_\theta \ \tau_2]^T$ ,  $\mathbf{C}_{le} = [1 \ 0 \ 0 \ 0]$

$$\tilde{\mathbf{A}}_{le} = \begin{bmatrix} -\frac{b+b_\theta}{J_1} & \frac{b_\theta}{J_1} & -\frac{1}{J_1} & 0 \\ \frac{b_\theta}{J_2} & -\frac{b+b_\theta}{J_2} & \frac{1}{J_2} & \frac{1}{J_2} \\ k_\theta & -k_\theta & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.2.49)$$

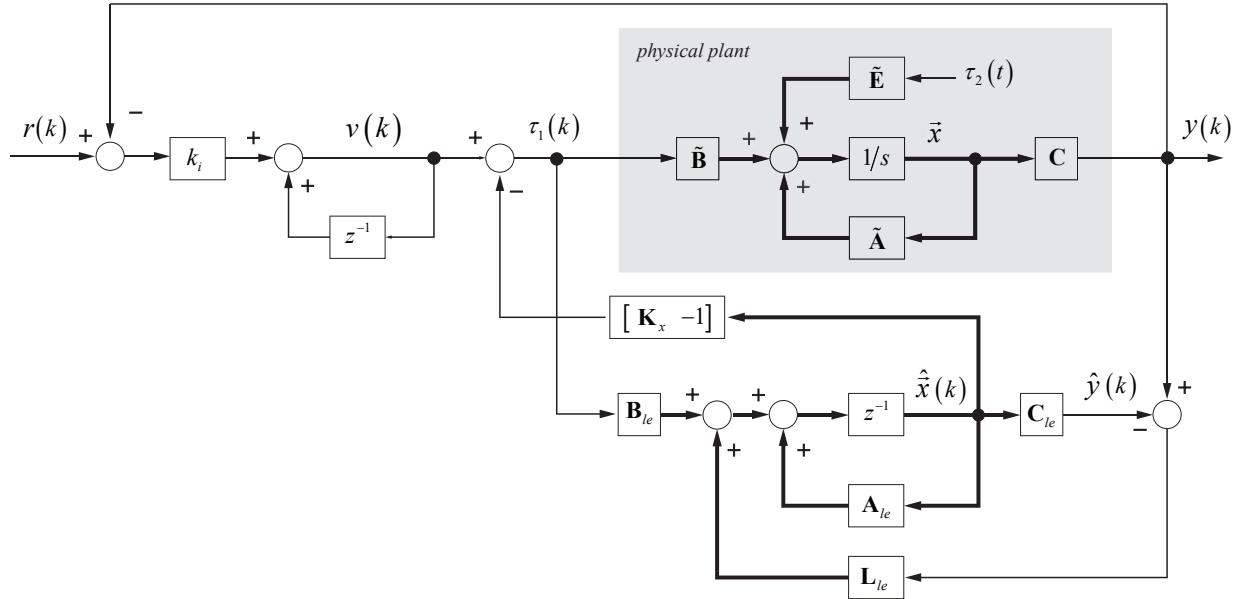


Figure 9.10: Description of the control strategy with integration and compensation of the load.

The system  $(\mathbf{A}_{le}, \mathbf{C}_{le})$  is fully observable.

To implement the state observer control algorithm the discrete time domain representation shall be derived<sup>§</sup>

$$\begin{aligned} \tilde{\mathbf{A}}_{le} &\rightarrow \mathbf{A}_{le} \approx \mathbf{I} + t_s \tilde{\mathbf{A}}_{le} \\ \tilde{\mathbf{B}}_{le} &\rightarrow \mathbf{B}_{le} \approx t_s \tilde{\mathbf{B}}_{le} \end{aligned} \quad (9.2.52)$$

---


$$\tilde{\mathbf{A}}_{le} \rightarrow \mathbf{A}_{le} = \mathbf{I} + t_s \tilde{\mathbf{A}}_{le} + \frac{t_s^2}{2} \tilde{\mathbf{A}}_{le}^2 + \frac{t_s^3}{6} \tilde{\mathbf{A}}_{le}^3 + \dots \quad (9.2.50)$$

$$\tilde{\mathbf{B}}_{le} \rightarrow \mathbf{B}_{le} = t_s \tilde{\mathbf{B}}_{le} + \frac{t_s^2}{2} \tilde{\mathbf{A}}_{le} \tilde{\mathbf{B}}_{le} + \frac{t_s^3}{6} \tilde{\mathbf{A}}_{le}^2 \tilde{\mathbf{B}}_{le} + \dots \quad (9.2.51)$$

we obtain the following state observer system

$$\begin{aligned}\hat{\vec{x}}_{le}(k+1) &= \mathbf{A}_{le} \hat{\vec{x}}_{le}(k) + \mathbf{B}_{le} \tau_1(t) + \mathbf{L}_{le} (\vec{y}(k) - \hat{\vec{y}}(k)) \\ \hat{\vec{y}}(k) &= \mathbf{C}_{le} \hat{\vec{x}}_{le}(k)\end{aligned}\quad (9.2.53)$$

Where matrix  $\mathbf{L}_{le}$  is calculated applying the pole placement to the system  $(\mathbf{A}_{le}, \mathbf{C}_{le})$ .

#### 9.2.4.1 Simulation results

The following simulation results consider at time  $t_{load} = 2$  s a load step of  $\tau_2 = 0.25$  N m.

The estimated load  $\hat{\tau}_2$  is feed-forwarded to the control output to compensate the load disturbance.

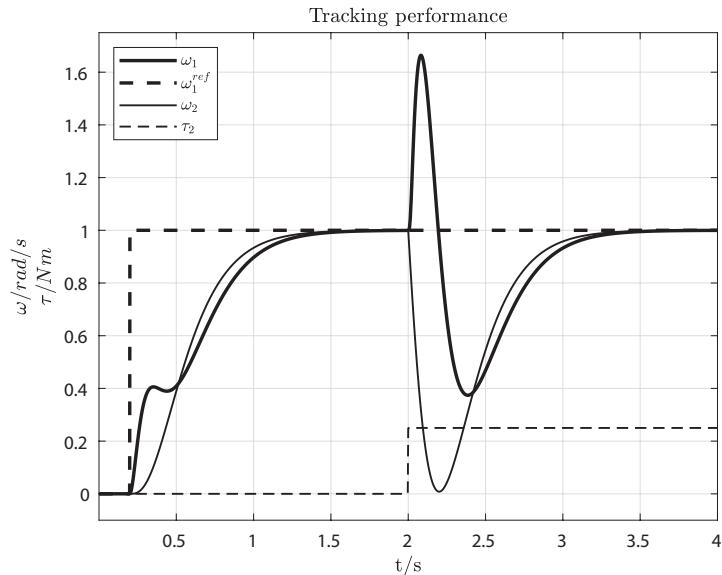


Figure 9.11: Step response of the state feedback control with observer and load compensation.  $\tau_2$  is the external load.

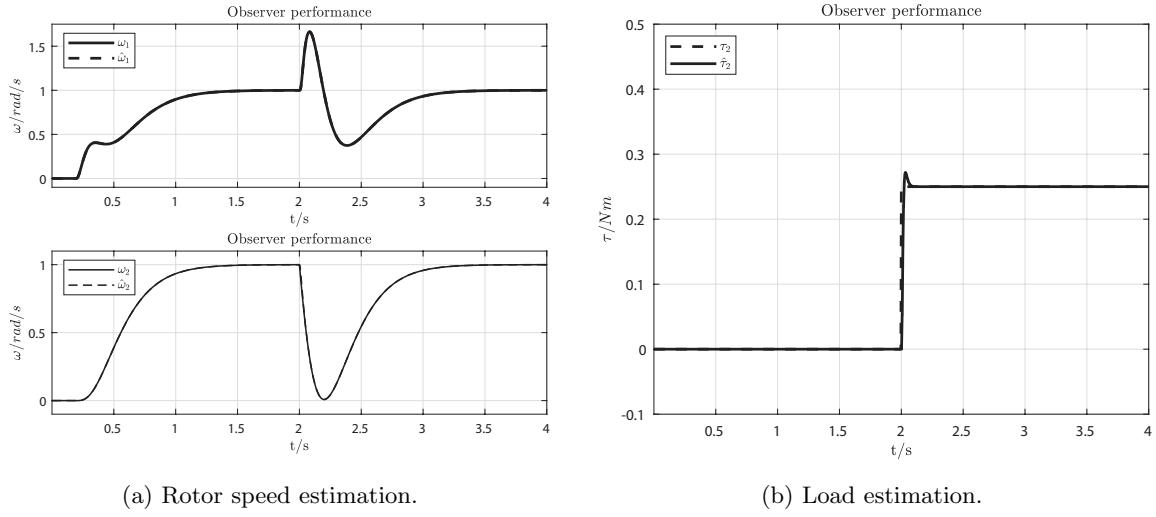


Figure 9.12: Observer performance.

## 9.3 Control of a Single Phase Inverter

In the following project we are going to consider a single phase inverter applied to an  $RL$  load with an output filter and a voltage disturbance  $d(t)$ . The aim of the control is the following:

- The quantity  $i_1(t)$  must track a sinusoidal reference  $r(t) = \sin(\omega_0 t)$  with constant frequency  $\omega_0$ .
- The sinusoidal disturbance  $d(t) = d \cos(\omega_n t)$  where  $\omega_n = n\omega_0$  must be estimated and compensated.

The following project could be classified as ‘‘harmonic compensator’’.

### 9.3.1 Model Derivation

In power electronics application the H-bridge cover a very wide application, both AC and DC power supply and active rectifiers. In the following a possible modelization of single phase inverter with output filter and disturbance is shown.

Figure 9.13 shows the the reference plant. Consider  $d(t) = d \cos(\omega_n t)$  as the voltage source disturbance.

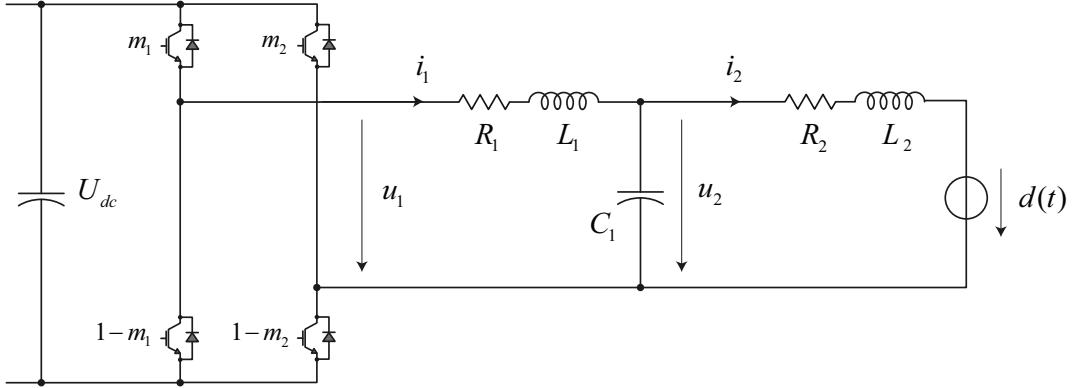


Figure 9.13: Single phase inverter circuit with load disturbance.

Consider the **Unipolar Voltage Switching** as PWM law: that means the **equivalent** voltage output of the inverter is given as follows

$$u_1 = (m_1 - m_2)U_{dc}$$

where  $m_1 \in [0 \ 1]$ ,  $m_2 \in [0 \ 1]$  and  $m_1 + m_2 = 1$ , that means

$$\begin{cases} u_1 = (2m_1 - 1) U_{dc} \\ m_1 + m_2 = 1 \end{cases}$$

A one-to-one relationship exists between  $u_1$  and  $m_1$ , therefore we consider the voltage input  $u_1$  a continuous time quantity and the equivalent circuit becomes as shown in Figure 9.14.

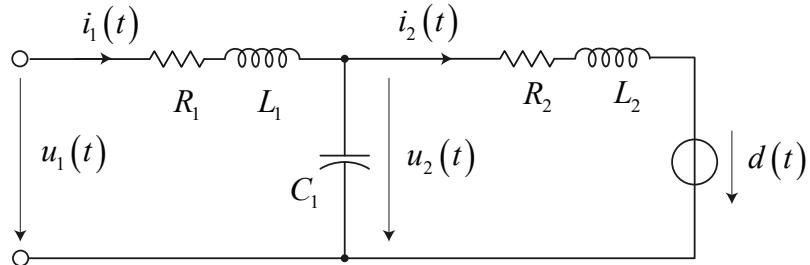


Figure 9.14: Equivalent model of the single phase inverter with load disturbance.

Applying Kirchhoff's rules we obtain

$$\begin{cases} u_1(t) - R_1 i_1(t) - L_1 \frac{di_1(t)}{dt} - u_2(t) = 0 \\ u_2(t) - R_2 i_2(t) - L_2 \frac{di_2(t)}{dt} - d(t) = 0 \\ C_1 \frac{du_2(t)}{dt} = i_1(t) - i_2(t) \end{cases}$$

Let  $\vec{x} = [i_1 \ i_2 \ u_2]^T$  be the state vector,  $u = u_1$  the control input and  $\vec{y} = [i_1 \ u_2]^T$  the measure vector.

**The state space representation becomes**

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) + \tilde{\mathbf{E}}d(t) \\ \vec{y}(t) = \tilde{\mathbf{C}}\vec{x}(t) \end{cases}$$

$$\begin{bmatrix} \frac{di_1}{dt}(t) \\ \frac{di_2}{dt}(t) \\ \frac{du_2}{dt}(t) \end{bmatrix} = \begin{bmatrix} -\frac{R_1}{L_1} & 0 & -\frac{1}{L_1} \\ 0 & -\frac{R_2}{L_2} & \frac{1}{L_2} \\ \frac{1}{C_1} & -\frac{1}{C_1} & 0 \end{bmatrix} \begin{bmatrix} i_1(t) \\ i_2(t) \\ u_2(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \end{bmatrix} u_1(t) + \begin{bmatrix} 0 \\ -\frac{1}{L_2} \\ 0 \end{bmatrix} d(t)$$

where

$$\begin{aligned} \tilde{\mathbf{A}} &= \begin{bmatrix} -\frac{R_1}{L_1} & 0 & -\frac{1}{L_1} \\ 0 & -\frac{R_2}{L_2} & \frac{1}{L_2} \\ \frac{1}{C_1} & -\frac{1}{C_1} & 0 \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \end{bmatrix} \\ \tilde{\mathbf{E}} &= \begin{bmatrix} 0 \\ -\frac{1}{L_2} \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

We know the nature of the disturbance which is defined as  $d(t) = \cos(\omega_n t)$ , where  $\ddot{d}(t) = -\omega_n^2 \cos(\omega_n t)$  ( $\omega_n$  is known) and that permits to extend the model as follows: let

$$\vec{x}_e = [i_1 \ i_2 \ u_2 \ d \ \dot{d}]^T$$

be the new state vector,  $u = u_1$  the control input and  $\vec{y} = [i_1 \ u_2]^T$  the measure vector.

**The new state space representation becomes**

$$\begin{cases} \dot{\vec{x}}_e(t) = \tilde{\mathbf{A}}_e\vec{x}_e(t) + \tilde{\mathbf{B}}_e u(t) \\ \vec{y}(t) = \tilde{\mathbf{C}}_e\vec{x}_e(t) \end{cases} \quad (9.3.1)$$

$$\begin{bmatrix} di_1(t)/dt \\ di_2(t)/dt \\ du_2(t)/dt \\ \dot{d}(t) \\ \ddot{d}(t) \end{bmatrix} = \begin{bmatrix} -\frac{R_1}{L_1} & 0 & -\frac{1}{L_1} & 0 & 0 \\ 0 & -\frac{R_2}{L_2} & \frac{1}{L_2} & -\frac{1}{L_2} & 0 \\ \frac{1}{C_1} & -\frac{1}{C_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\omega_n^2 & 0 \end{bmatrix} \begin{bmatrix} i_1(t) \\ i_2(t) \\ u_2(t) \\ d(t) \\ \dot{d}(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_1(t) \quad (9.3.2)$$

where

$$\tilde{\mathbf{A}}_e = \begin{bmatrix} -\frac{R_1}{L_1} & 0 & -\frac{1}{L_1} & 0 & 0 \\ 0 & -\frac{R_2}{L_2} & \frac{1}{L_2} & -\frac{1}{L_2} & 0 \\ \frac{1}{C_1} & -\frac{1}{C_2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\omega_n^2 & 0 \end{bmatrix}, \quad \tilde{\mathbf{B}}_e = \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{C}}_e = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

### 9.3.2 Control Design

The following control architecture has been chosen (see also Figure 9.16)

- Proportional resonant controller in the forward path.
- Estimation of disturbance  $d(t)$ .
- Feed-forward of the estimated disturbance signal<sup>¶</sup>

<sup>¶</sup>The intuitive concept of the feed-forward compensation lays on the concept of the feedback linearization. In its simplest form, feedback-linearization consist in the cancellation of the nonlinearities in a nonlinear system so that the closed loop dynamics is in linear form.

**Example 9.1.** Consider the control of the level  $h(t)$  of fluid in a tank to a specific level  $h_d$ . The control input is the flow  $u(t)$  into the tank and the initial level is  $h_0$ .

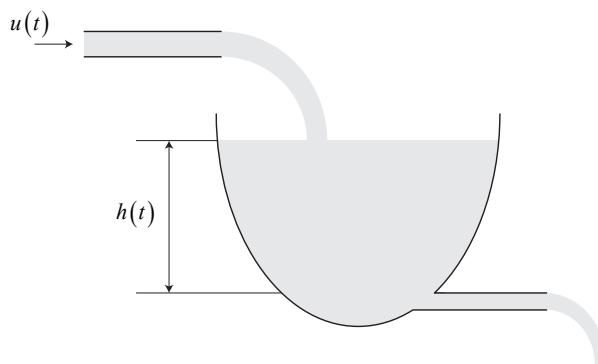


Figure 9.15: Fluid level control in a tank.

The dynamic model of the tank is

$$\frac{d}{dt} \left[ \int_{h_0}^{h(t)} A(y) dy \right] = u(t) - a\sqrt{2gh(t)} \quad (9.3.3)$$

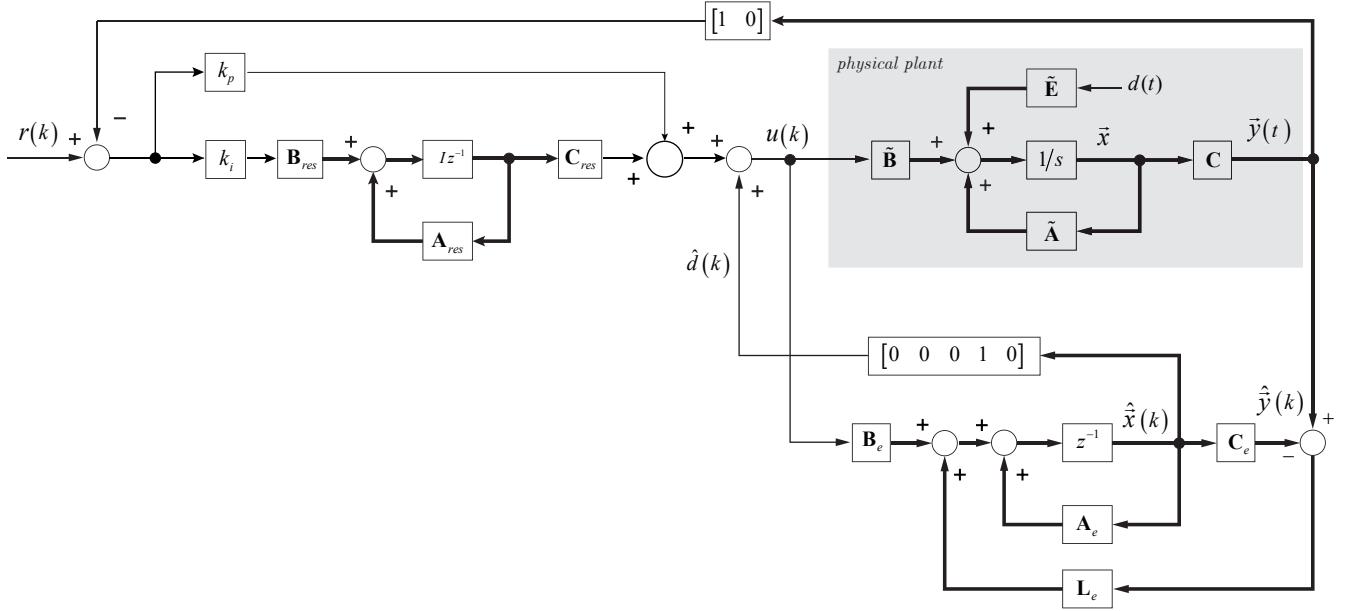


Figure 9.16: Control architecture for harmonic compensator.

To implement the control blocks we have to design the **Proportional Resonant** control and a state observer. As previously seen the **Proportional Resonant** control matrices (for

where  $A(h)$  is the cross section of the tank and  $a$  is the cross section of the outlet pipe. If the initial level  $h_0$  is quite different from the desired level  $h_d$ , the control of  $h(t)$  involves a nonlinear regulation problem.

The dynamic Eq. 9.3.3 can be rewritten as

$$\frac{d}{dt} \left[ \int_{h_0}^{h(t)} A(y) dy \right] = \frac{\partial}{\partial h} \left[ \int_{h_0}^{h(t)} A(y) dy \right] \frac{dh(t)}{dt} = A(h) \dot{h}(t) = u(t) - a\sqrt{2gh(t)} \quad (9.3.4)$$

then

$$A(h) \dot{h}(t) = u(t) - a\sqrt{2gh(t)} \quad (9.3.5)$$

If  $u(t)$  is chosen as

$$u(t) = a\sqrt{2gh(t)} + A(h)\nu \quad (9.3.6)$$

with  $\nu$  being an *equivalent input* to be specified, the resulting dynamics is linear

$$\dot{h}(t) = \nu \quad (9.3.7)$$

Choosing  $\nu$  as

$$\nu = \dot{h}_d(t) - \alpha \tilde{h}(t) \quad (9.3.8)$$

with  $\tilde{h}(t) = h(t) - h_d(t)$  being the level error, and  $\alpha \in \mathbb{R}^+$ . The resulting closed loop dynamics become

$$\dot{\tilde{h}}(t) + \alpha \tilde{h}(t) = 0 \quad (9.3.9)$$

which is asymptotic stable for every  $\alpha \in \mathbb{R}^+$ . The effective input law is determined by the nonlinear control law

$$u(t) = a\sqrt{2gh(t)} - A(h)\alpha \tilde{h}(t) \quad (9.3.10)$$

Note that, in the control law of Eq. 9.3.10, the first part on the right-hand side is used to provide the output flow  $a\sqrt{2gh(t)}$ , while the second part is used to raise the fluid level according to the desired linear dynamics of Eq. 9.3.9.  $\triangleleft$

state space representation) can be derived as follows

$$\mathbf{A}_{res} = \begin{bmatrix} \cos(\omega_0 t_s) & \frac{\sin(\omega_0 t_s)}{\omega_0} \\ -\omega_0 \sin(\omega_0 t_s) & \cos(\omega_0 t_s) \end{bmatrix} \quad (9.3.11)$$

$$\mathbf{B}_{res} = \begin{bmatrix} \frac{1-\cos(\omega_0 t_s)}{\omega_0^2} \\ \frac{\sin \omega_0 t_s}{\omega_0} \end{bmatrix}, \quad \mathbf{C}_{res} = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad (9.3.12)$$

The observer can be derived as follows

$$\begin{cases} \hat{\vec{x}}_e(k+1) = \mathbf{A}_e \hat{\vec{x}}_e(k+1) + \mathbf{B}_e u(t) + \mathbf{L}_e (\vec{y}(k) - \hat{\vec{y}}(k)) \\ \hat{\vec{y}}(k) = \mathbf{C}_e \hat{\vec{x}}_e(k) \end{cases} \quad (9.3.13)$$

where  $\mathbf{C}_e = \tilde{\mathbf{C}}_e$  and  $\mathbf{A}_e, \mathbf{B}_e$  derive from discretization of the matrix  $\tilde{\mathbf{A}}_e$  and  $\tilde{\mathbf{B}}_e$  which can be obtained by Taylor approximation.

### 9.3.2.1 Simulation Results

In the following, simulation results are shown considering the following data:

$R_1 = 100 \text{ m}\Omega$	$L_1 = 250 \text{ }\mu\text{H}$	$C_1 = 400 \text{ }\mu\text{F}$
$R_2 = 100 \text{ m}\Omega$	$L_2 = 60 \text{ }\mu\text{H}$	$t_s = 100 \text{ }\mu\text{s}$
$\omega_0 = 2\pi 50 \text{ rad s}^{-1}$	$\omega_n = 5\omega_0$	$d = 0.35 \text{ V}$

Table 9.2: Components data of the filter and load.

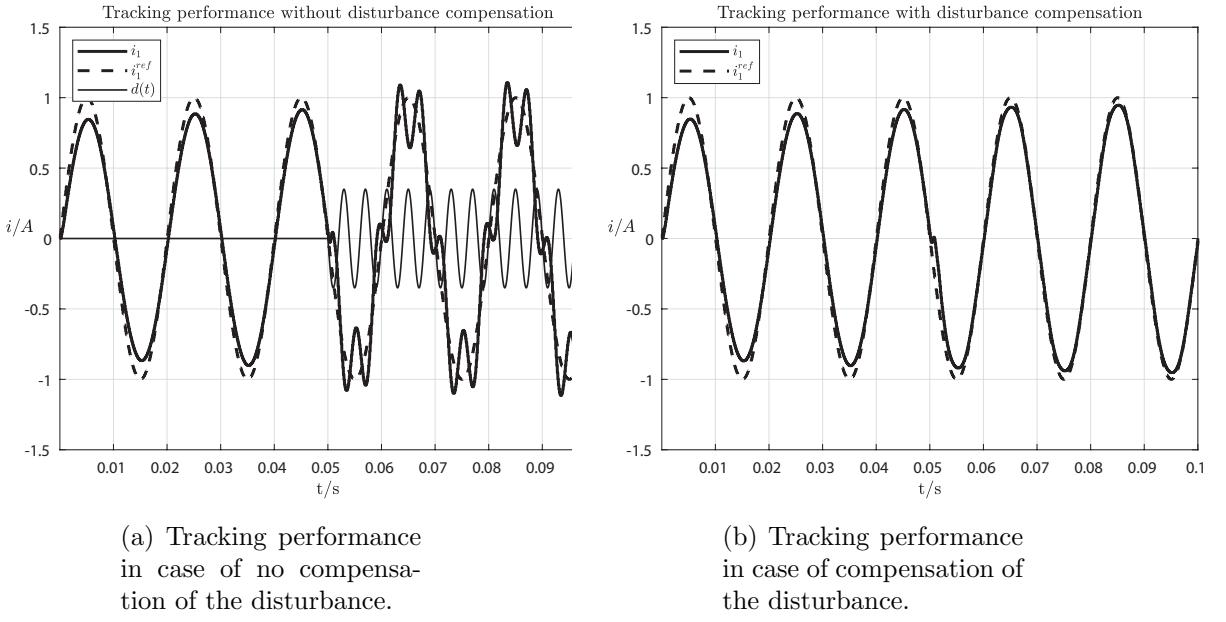
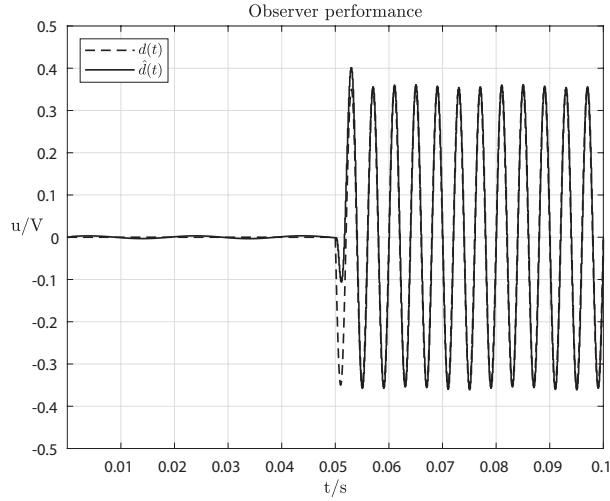


Figure 9.17: Simulation results.


 Figure 9.18: Performance estimation of the disturbance  $d(t)$ .

## 9.4 Permanent-Magnet Linear Actuator - Moving Coil

In the following project we are going to consider a linear actuator consisting of a conductive bar of mass  $m$  in sliding contact with a pair of stationary conducting rails as shown in Figure 9.19. The rails are connected to a voltage source  $u(t)$ . The bar is moving through a constant uniform magnetic field  $\vec{B}^M$  with a time dependent velocity  $v(t)$  relative to the rails. We can consider the rails an inertial reference frame. This model could be useful for modeling a plotter or in general a linear actuator.

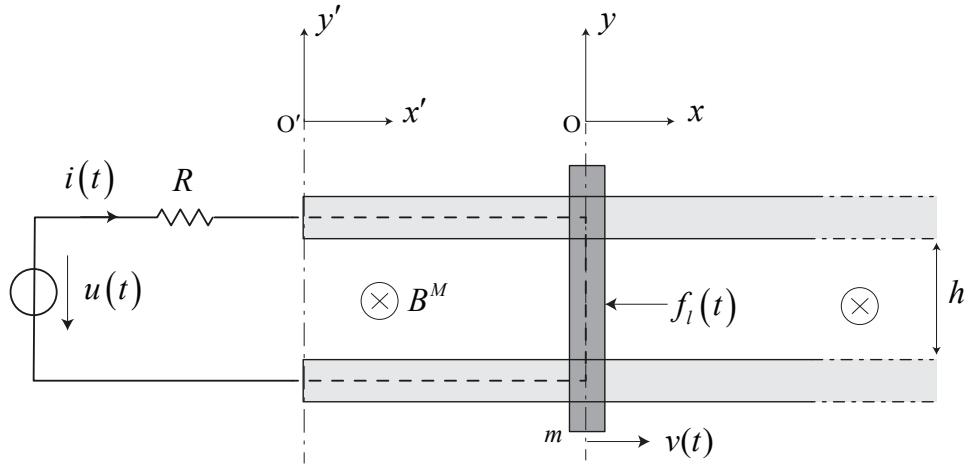


Figure 9.19: Linear actuator.

#### 9.4.1 Model Derivation

Consider

$$\psi(t) = \int_S \vec{B} \cdot d\vec{s} \quad (9.4.1)$$

the flux linkage to the circuit. At this point we have to point out that the magnetic field  $\vec{B}$  include two main contributes: one constant term which comes from the external permanent magnet  $\vec{B}^M$  and one which is generated by the flowing current  $i$ , that means  $\vec{B} = \vec{B}_i + \vec{B}^M$ .

Applying the general Kirchhoff's law we obtain

$$u(t) - Ri(t) + \frac{d\psi(t)}{dt} = 0 \quad (9.4.2)$$

The term  $d\psi(t)/dt$  represents the voltage induced in the circuit by a time rate of change of magnetic flux through the circuit. The change in flux can be due to a change in circuit current and/or the movement of the bar. Hence, the magnetic flux can be written as function of space and current  $\psi(i, x)$

$$\psi(t) = \int_S \vec{B} \cdot d\vec{s} = \int_S (\vec{B}_i + \vec{B}^M) \cdot d\vec{s} = \psi(i, x) \quad (9.4.3)$$

Hence the total time derivative of  $\psi(i, x)$  can be written as follows

$$\frac{d\psi(i, x)}{dt} = \frac{\partial\psi(i, x)}{\partial i(t)} \frac{di(t)}{dt} + \frac{\partial\psi(i, x)}{\partial x(t)} \frac{dx(t)}{dt} \quad (9.4.4)$$

where  $h$  (length of the bar) and  $B^M$  (magnitude of the permanent magnet magnetic field) are considered constant parameters

$$\frac{\partial\psi(i, x)}{\partial x(t)} \frac{dx(t)}{dt} = \frac{\partial Bxh}{\partial x} v(t) \approx B^M h v(t) \quad (9.4.5)$$

and

$$\frac{\partial\psi(i, x)}{\partial i(t)} \frac{di(t)}{dt} = L(x) \frac{di}{dt} \approx L \frac{di}{dt} \quad (9.4.6)$$

Eq. 9.4.2 becomes

$$u(t) - Ri(t) - L \frac{di(t)}{dt} - B^M h v(t) = 0 \quad (9.4.7)$$

The expression of the force due to the current  $i$  in the magnetic field  $\vec{B}^M$  is given as follows  
**(integration path follows the current direction)**

$$f_m = i(t) \int_h^0 d\vec{l} \times \vec{B}^M = B^M h i(t) \quad \text{positive x-direction} \quad (9.4.8)$$

The complete set of system equations becomes

$$\begin{cases} \frac{dx(t)}{dt} = v(t) \\ \frac{dv(t)}{dt} = \frac{B^M h}{m} i(t) - \frac{b}{m} v(t) - \frac{1}{m} f_l(t) \\ \frac{di(t)}{dt} = -\frac{R}{L} i(t) - \frac{1}{L} B^M h v(t) + \frac{1}{L} u(t) \end{cases} \quad (9.4.9)$$

the first two equations represent the dynamic motion of the bar where the force  $f_m(t) = B^M h i(t)$  is generated by the interaction of the current  $i(t)$  and the magnetic field  $\vec{B}^M$ . The third equation represents the dynamic of the electrical circuit which generates the current  $i(t)$  by applying the voltage  $u(t)$ . The equivalent auto-inductance  $L$  represents the effect of the Faraday's law.

The current  $i(t)$  and the position  $x(t)$  are measured, hence, the state space representation can be represented as follows

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}u(t) + \tilde{\mathbf{E}}d(t) \\ \vec{y}(t) = \tilde{\mathbf{C}}\vec{x}(t) \end{cases}$$

$$\begin{bmatrix} \frac{dx(t)}{dt} \\ \frac{dv(t)}{dt} \\ \frac{di(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{b}{m} & \frac{B^M h}{m} \\ 0 & -\frac{B^M h}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \\ i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ -\frac{1}{m} \\ 0 \end{bmatrix} f_l(t)$$

$$\tilde{\mathbf{A}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{b}{m} & \frac{B^M h}{m} \\ 0 & -\frac{B^M h}{L} & -\frac{R}{L} \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix}$$

$$\tilde{\mathbf{E}} = \begin{bmatrix} 0 \\ -\frac{1}{m} \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

#### 9.4.2 Control Problem

Given the following data:

1. Rails friction  $b = 20 \text{ N s m}^{-1}$
2. Length of the bar  $h = 0.2 \text{ m}$
3. Mass of the bar  $m = 0.03 \text{ kg}$
4. Magnitude of the permanent magnet magnetic field  $B^M = 1.2 \text{ T}$
5. Equivalent circuit inductance  $L = 1 \text{ mH}$
6. Equivalent circuit resistance  $R = 1 \Omega$
7.  $f_l(t) = b_2 v(t)$  is an unmodeled viscosity, where  $b_2 = 0.1 \text{ N s m}^{-1}$  for case (1) and  $b_2 = 1 \text{ N s m}^{-1}$  for case (2)

we want to design a controller which is able to position the bar (plotter head) from a initial point to a given final point without any residual error.

#### 9.4.3 Control Design without Load Estimator

To meet the control requirements a possible implementation is a state feedback control with integrator. Obviously the full state is not measurable, hence a full state observer will be also implemented.

The selected control layout is reported in Figure 9.20. To implement the control structure we have to calculate the state feedback vector which includes the gain corresponding to the integral block  $\begin{bmatrix} \mathbf{K}_x & k_i \end{bmatrix}$  and a state observer. For both purposes we have to calculate the corresponding  $\mathbf{A}$  and  $\mathbf{B}$  matrices in discrete time domain as follows

$$\mathbf{A} = \mathbf{I} + \tilde{\mathbf{A}} t_s \quad (9.4.10)$$

$$\mathbf{B} = \tilde{\mathbf{B}} t_s \quad (9.4.11)$$

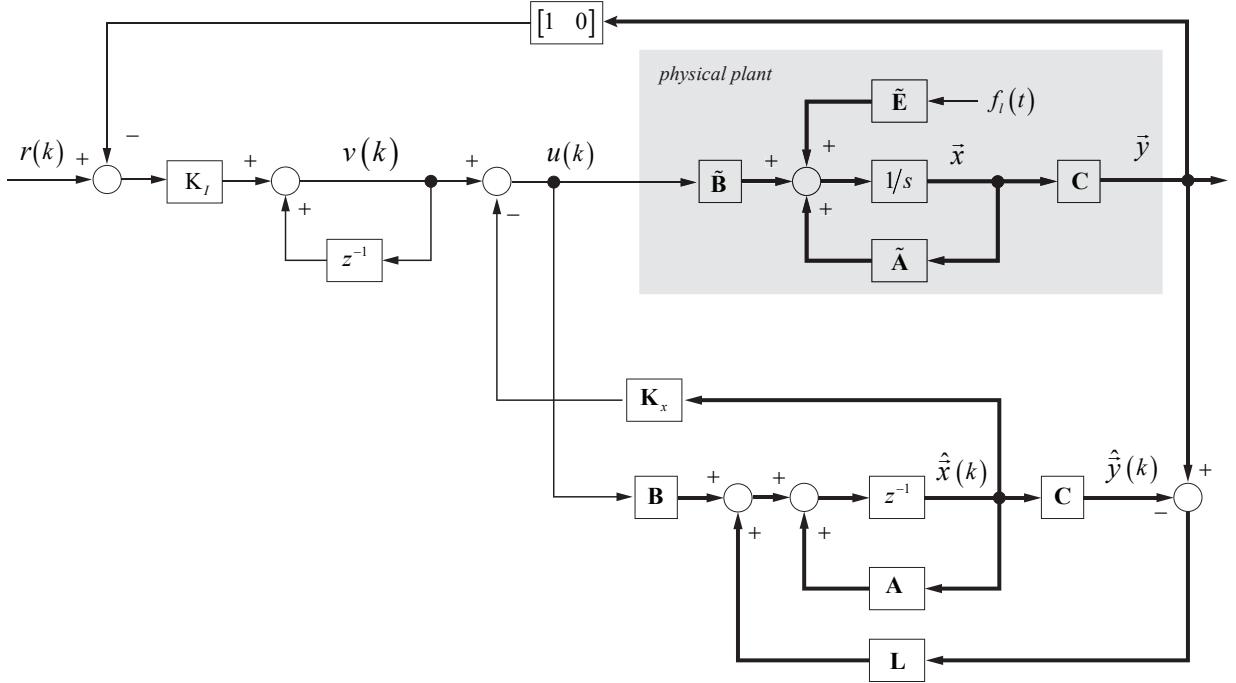


Figure 9.20: Control architecture.

Once the matrices  $\mathbf{A}$  and  $\mathbf{B}$  are obtained we can proceed implementing the extended system to calculate  $[K_I \quad \mathbf{K}_x]$  as follows

$$\mathbf{A}' = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & 0 \end{bmatrix} = (n+1) \times (n+1) \text{ matrix}$$

$$\mathbf{B}' = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} = (n+1) \text{ column vector}$$

Obtained  $\mathbf{A}'$  and  $\mathbf{B}'$  we can proceed calculating the state feedback vector  $\mathbf{K}'$  from the Ackermann's formula

$$\mathbf{K}' = [0 \quad 0 \quad \dots \quad 0 \quad 1] \mathbf{M}'^{-1} q(\mathbf{A}')$$

where  $\mathbf{M}' = [\mathbf{B}' \quad \mathbf{A}'\mathbf{B}' \quad \dots \quad (\mathbf{A}')^{n+m-1} \mathbf{B}']$ . Ackermann's formula uses the **Cayley–Hamilton** theorem and  $q(z)$  is the desired final polynomial characteristics.

The **Cayley–Hamilton** theorem states that  $\mathbf{A}'$  satisfies its own characteristic equation, hence

$$q(\mathbf{A}') = \mathbf{A}'^n + \alpha_1 \mathbf{A}'^{n-1} + \dots + \alpha_{n-1} \mathbf{A}' + \alpha_n \mathbf{I} = \mathbf{0} \quad (9.4.12)$$

Once obtained the matrix  $\mathbf{K}'$  we can use the formula

$$[\mathbf{K}_x \quad k_i] = \left\{ \mathbf{K}' + \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \right\} \begin{bmatrix} \mathbf{A} - \mathbf{I}_4 & \mathbf{B} \\ \mathbf{C}\mathbf{A} & \mathbf{C}\mathbf{B} \end{bmatrix}^{-1} \quad (9.4.13)$$

to obtain the state feedback gains  $\begin{bmatrix} \mathbf{K}_x & k_i \end{bmatrix}$ .

Regarding the state observer we have just to calculate the matrix  $\mathbf{L}$ . In this case the system is SIMO because the output vector  $\vec{y}$  and obviously  $\hat{\vec{y}}$  have dimension two and we have to use the Matlab command `place()`. The observer block becomes

$$\begin{aligned}\hat{\vec{x}}(k+1) &= \mathbf{A} \hat{\vec{x}}(k) + \mathbf{B} u(k) + \mathbf{L} (\vec{y}(k) - \hat{\vec{y}}(k)) \\ \hat{\vec{y}}(k) &= \mathbf{C} \hat{\vec{x}}(k)\end{aligned}\quad (9.4.14)$$

as reported in Figure 9.20

#### 9.4.3.1 Simulation results

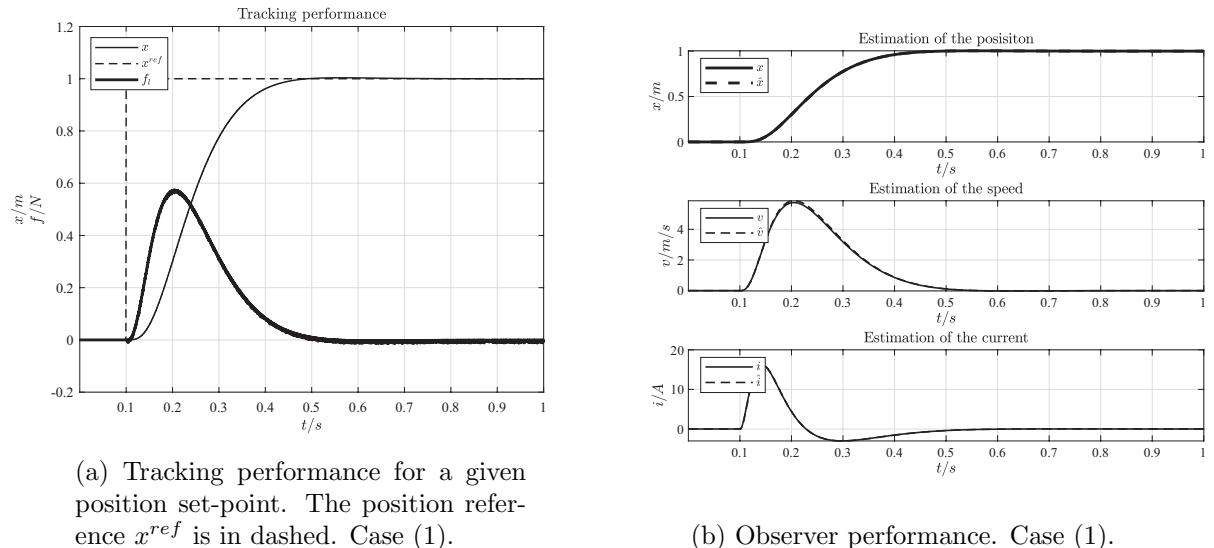
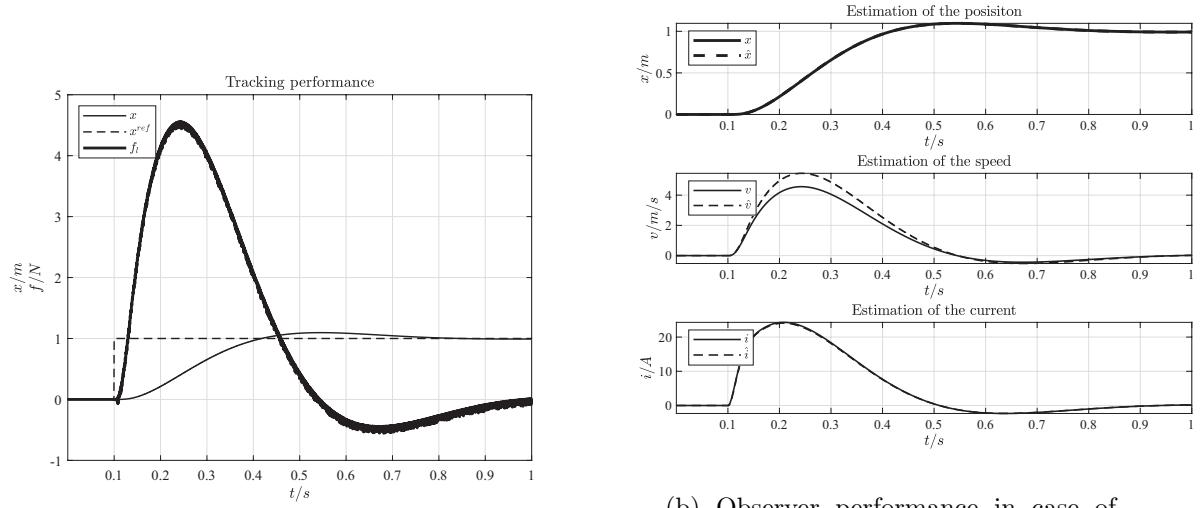


Figure 9.21: Simulation results.

**Limits of the model** - The observer we have implemented presents some limits in particular in case (2) where unmodeled friction becomes relevant the estimation of the speed ( $v(t)$ ) accumulate an error as function of the load as reported in Figure 9.22b where also the tracking performance are affected as shown in Figure 9.22a



(a) Tracking performance for a given position set-point. The position reference  $x^{ref}$  is in dashed. Case (2).

(b) Observer performance in case of high level of load force. Note the residual error in the estimation of the load force. Case (2).

Figure 9.22: Simulation results.

To avoid this degradation if the performance we would like to extend our state observer including an approximation of the dynamic of the load.

#### 9.4.4 Control Design with Load Estimator

In this section we want to extend the observer including the dynamic of the load and try to increase the robustness against friction degradation of the rails. Let's start with a possible load model.

We can suppose a load of the form (the nature of the load force is friction, which means, a force which depends by speed):

$$f_l(t) = v(t)$$

where its dynamic can be expressed as

$$\dot{f}_l(t) = \ddot{v}(t)$$

From Eq. 9.4.9 we can write

$$\dot{f}_l(t) = \ddot{v}(t) = \frac{B^M h}{m} i(t) - \frac{b}{m} v(t) - \frac{1}{m} f_l(t)$$

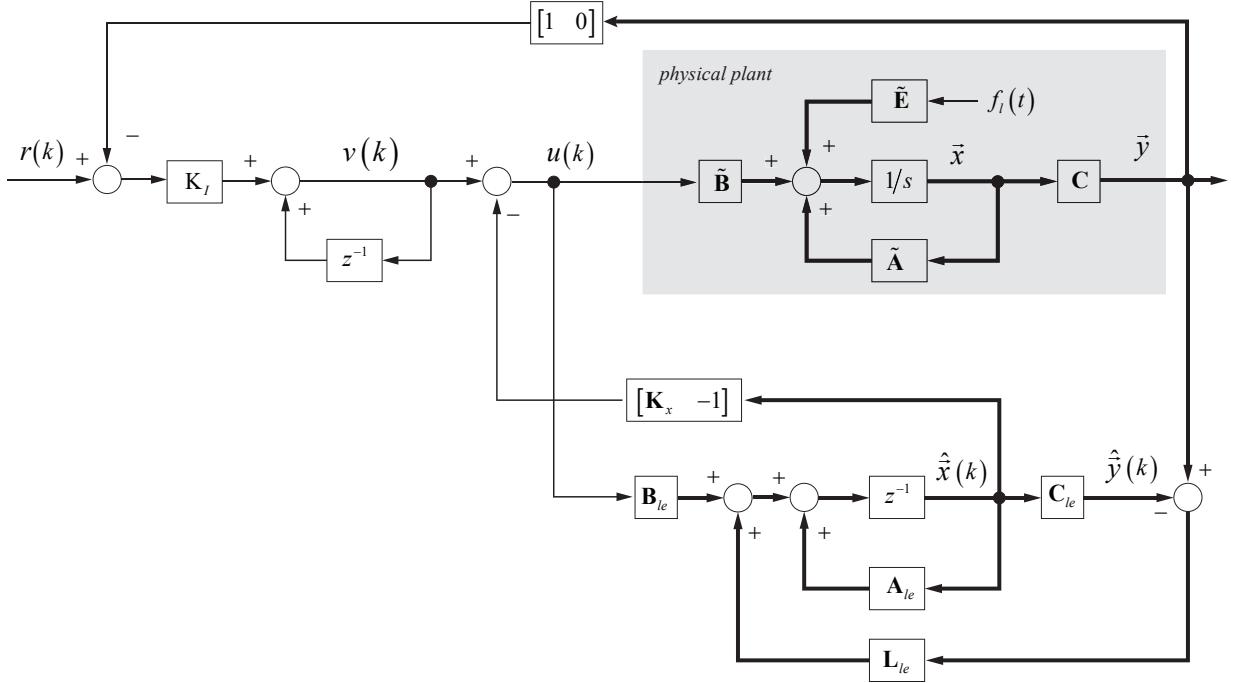


Figure 9.23: Control architecture with load estimator.

The extended (open loop) observer model can be described as follows

$$\begin{bmatrix} \frac{d\hat{x}(t)}{dt} \\ \frac{d\hat{v}(t)}{dt} \\ \frac{d\hat{i}(t)}{dt} \\ \frac{d\hat{f}_l(t)}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{b}{m} & \frac{B^M h}{m} & -\frac{1}{m} \\ 0 & -\frac{B^M h}{L} & -\frac{R}{L} & 0 \\ 0 & -\frac{b}{m} & \frac{B^M h}{m} & -\frac{1}{m} \end{bmatrix} \begin{bmatrix} \hat{x}(t) \\ \hat{v}(t) \\ \hat{i}(t) \\ \hat{f}_l(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \\ 0 \end{bmatrix} u(t) \quad (9.4.15)$$

where the closed loop form is

$$\begin{aligned} \hat{\vec{x}}_{le}(t) &= \tilde{\mathbf{A}}_{le} \hat{\vec{x}}_{le}(t) + \tilde{\mathbf{B}}_{le} \tau_1(t) + \tilde{\mathbf{L}}_{le} (\vec{y}(t) - \hat{\vec{y}}(t)) \\ \hat{\vec{y}}(t) &= \mathbf{C}_{le} \hat{\vec{x}}_{le}(t) \end{aligned} \quad (9.4.16)$$

$$\text{where } \vec{x}_{le} = [x(t) \ v(t) \ i(t) \ f_l(t)]^T, \ \mathbf{C}_{le} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\tilde{\mathbf{A}}_{le} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{b}{m} & \frac{B^M h}{m} & -\frac{1}{m} \\ 0 & -\frac{B^M h}{L} & -\frac{R}{L} & 0 \\ 0 & -\frac{b}{m} & \frac{B^M h}{m} & -\frac{1}{m} \end{bmatrix} \quad (9.4.17)$$

The system  $(\mathbf{A}_{le}, \mathbf{C}_{le})$  is fully observable. After the discretization

$$\begin{aligned}\tilde{\mathbf{A}}_{le} &\rightarrow \mathbf{A}_{le} \\ \tilde{\mathbf{B}}_{le} &\rightarrow \mathbf{B}_{le}\end{aligned}\quad (9.4.18)$$

$$\mathbf{A}_{le} = \mathbf{I} + \tilde{\mathbf{A}}_{le} t_s \quad (9.4.19)$$

or

$$\mathbf{B}_{le} = \tilde{\mathbf{B}}_{le} t_s \quad (9.4.20)$$

we obtain the following state observer system

$$\begin{aligned}\hat{x}_{le}(k+1) &= \mathbf{A}_{le} \hat{x}_{le}(k) + \mathbf{B}_{le} \tau_1(t) + \mathbf{L}_{le} (\vec{y}(k) - \hat{y}(k)) \\ \hat{y}(k) &= \mathbf{C}_{le} \hat{x}_{le}(k)\end{aligned}\quad (9.4.21)$$

Where matrix  $\mathbf{L}_{le}$  is calculated applying `place()` formula considering the system  $(\mathbf{A}_{le}, \mathbf{C}_{le})$ .

#### 9.4.4.1 Simulation Results

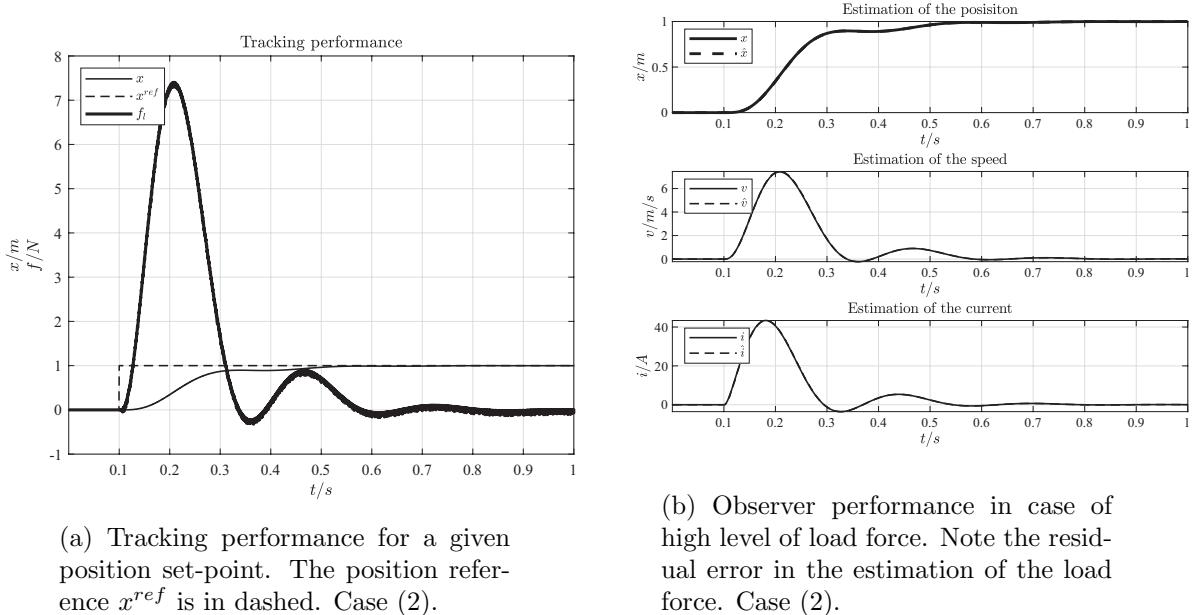


Figure 9.24: Simulation results.

**Summary** - In this example we have seen an application of the state feedback with integral control. This kind of approach is useful when it is necessary to achieve a zero error steady state tracking from a constant reference and the plant doesn't contain an integrator. In addition, we have implemented a state observer, at first glance, without including the dynamic of the load, which is here considered as a disturbance or plant parameter deviation (like ageing in the

rails viscosity). We have seen that this plant deviation can degrade the performance of the servo and increasing the time to set, because the load enter in the system like a disturbance and the controller has to compensate it, spending “time” and “energy” which results in a modification of the global dynamics.

In the second part we tried to modelize the load dynamics supposing it comes from a not modeled viscosity of the rails. Increasing the observer (in terms of states elements), namely, including the dynamics of the load, we were able to estimate the load and to reduce, at least partially, the degradation of the tracking performance.

## 9.5 Pendulum on Cart

In this section we propose an application of state feedback and state observer control applied to an inverter pendulum placed on a cart. The system is represented in Figure 9.25.

The scope of the project is to design a controller which permits to track the cart, from a given position reference, keeping the pendulum as close as possible to its vertical equilibrium point.

### 9.5.1 Model Derivation

According Figure 9.25 the system is governed (control input) by the force  $f_m(t)$ . The quantity  $f_d(t)$  represents an external load or disturbance which we consider constant or with a lower dynamics in comparison to the pendulum on cart.

The generalized coordinates which describe the system are

- The cart position  $x(t)$
- The pendulum angle  $\vartheta(t)$

Both the wheel of the cart as well as the mass of the pendulum are subjected to a viscosity  $b_0$  and  $b_1$  respectively.

To derive the dynamical equations which govern the system, we apply the Euler-Lagrange equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = Q \quad i = 1, 2, \dots, n \quad (9.5.1)$$

where the Lagrange function  $\mathcal{L}$  is

$$\mathcal{L} = E_{kin} - E_{pot} \quad (9.5.2)$$

where

$$\begin{aligned} E_{kin} &= \frac{1}{2}m_0\dot{x}^2 + \frac{1}{2}m_1(\dot{x} + l\dot{\vartheta} \cos \vartheta)^2 + \frac{1}{2}m_1(l\dot{\vartheta} \sin \vartheta)^2 + \frac{1}{2}J_1\dot{\vartheta}^2 \\ &= \frac{1}{2}(m_0 + m_1)\dot{x}^2 + \frac{1}{2}(m_1l^2 + J_1)\dot{\vartheta}^2 + m_1l\dot{\vartheta}\dot{x} \cos \vartheta \end{aligned} \quad (9.5.3)$$

and

$$E_{pot} = m_1gl \cos \vartheta. \quad (9.5.4)$$

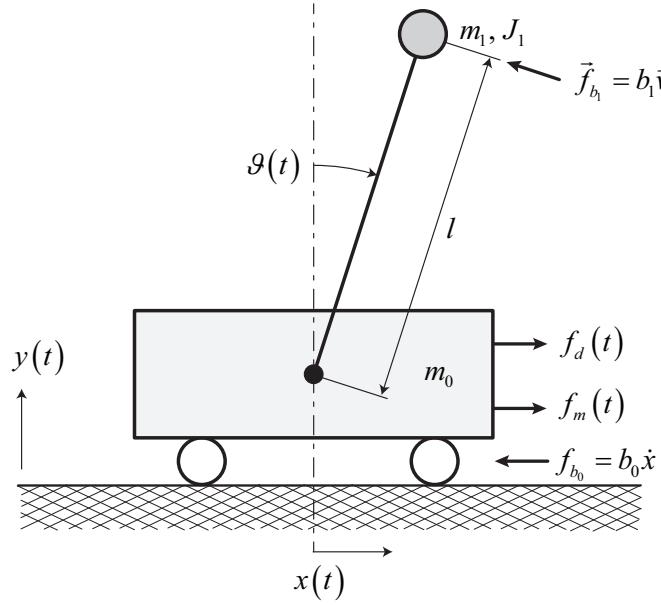


Figure 9.25: Pendulum on cart.

Hence, the Lagrange function  $\mathcal{L}$  become

$$\mathcal{L} = \frac{1}{2}(m_0 + m_1)\dot{x}^2 + \frac{1}{2}(m_1l^2 + J_1)\dot{\vartheta}^2 + m_1l\dot{\vartheta}\dot{x}\cos\vartheta - m_1gl\cos\vartheta. \quad (9.5.5)$$

Now, we suppose to apply Eq. (9.5.1) for the variables  $x$  and  $\vartheta$  and we obtain the following dynamic equations

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{x}} - \frac{\partial\mathcal{L}}{\partial x} = f_m + f_d - b_0\dot{x} \quad (9.5.6)$$

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot{\vartheta}} - \frac{\partial\mathcal{L}}{\partial\vartheta} = -b_1\dot{\vartheta} \quad (9.5.7)$$

which results in

$$(m_0 + m_1)\ddot{x} + m_1l\ddot{\vartheta}\cos\vartheta - m_1l\dot{\vartheta}^2\sin\vartheta = f_m - b_0\dot{x} \quad (9.5.8)$$

and

$$\begin{aligned} & (m_1l^2 + J_1)\ddot{\vartheta} + m_1l\ddot{x}\cos\vartheta - m_1l\dot{x}\dot{\vartheta}\sin\vartheta + m_1l\dot{x}\dot{\vartheta}\sin\vartheta - m_1gl\sin\vartheta = \\ & = (m_1l^2 + J_1)\ddot{\vartheta} + m_1l\ddot{x}\cos\vartheta = -b_1\dot{\vartheta} \end{aligned} \quad (9.5.9)$$

respectively

$$\begin{cases} (m_0 + m_1)\ddot{x} + m_1l\ddot{\vartheta}\cos\vartheta - m_1l\dot{\vartheta}^2\sin\vartheta = f_m + f_d - b_0\dot{x} \\ (m_1l^2 + J_1)\ddot{\vartheta} + m_1l\ddot{x}\cos\vartheta = -b_1\dot{\vartheta} \end{cases} \quad (9.5.10)$$

The solution of the system of Eq. (9.5.10) brings to the following canonical representation

$$\begin{cases} \dot{x} = v \\ \dot{v} = \frac{-m_1 l \cos \vartheta (-b_1 \omega + g m_1 l \sin \vartheta) + (J_1 + m_1 l^2)(f_m + f_d - b_0 v + m_1 l \omega^2 \sin \vartheta)}{(m_0 + m_1)(J_1 + m_1 l^2) - (m_1 l \cos \vartheta)^2} \\ \dot{\vartheta} = \omega \\ \dot{\omega} = \frac{(m_0 + m_1)(-b_1 \omega + g m_1 l \sin \vartheta) - m_1 l \cos \vartheta (f_m + f_d - b_0 v + m_1 l \omega^2 \sin \vartheta)}{(m_0 + m_1)(J_1 + m_1 l^2) - (m_1 l \cos \vartheta)^2} \end{cases} \quad (9.5.11)$$

As mention at the beginning of this section, the scope of this project is to design a controller which permits to track the cart, from a given position reference, keeping the pendulum as close as possible to its vertical equilibrium point.

Assuming the state vector of the system as

$$\vec{x}(t) = [x(t) \quad v(t) \quad \vartheta(t) \quad \omega(t)]^T \quad (9.5.12)$$

and the system Eq. (9.5.11) can be written as follows

$$\dot{\vec{x}} = \vec{f}(x, v, \vartheta, \omega) \quad (9.5.13)$$

the equilibrium point can be represented as  $\vec{x}_0 = [0 \quad 0 \quad 0 \quad 0]^T$  (with  $f_{m,0} = 0$ ). The system represented by Eq. (9.5.11) is, of course, nonlinear and, written in such a way, is not suitable for pole placement technique. That means, we have now to find a linear representation of the system around the equilibrium point  $\vec{x}_0 = [0 \quad 0 \quad 0 \quad 0]^T$  applying Taylor rules.

Lets now write the model in matrix notation (that mean the linear equivalent representation of Eq. (9.5.11)) and after that we write down the steps necessary to derive the matrices  $\tilde{\mathbf{A}}$  and  $\tilde{\mathbf{B}}$  (here, we assume  $f_d(t) = 0$ ):

$$\dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}f_m(t) + \tilde{\mathbf{E}}f_d(t) \quad (9.5.14)$$

where

$$\tilde{\mathbf{A}} = \left. \frac{\partial \vec{f}}{\partial \vec{x}} \right|_{\vec{x}_0, f_{m,0}} \quad (9.5.15)$$

and

$$\tilde{\mathbf{E}} = \tilde{\mathbf{B}} = \left. \frac{\partial \vec{f}}{\partial f_m} \right|_{\vec{x}_0, f_{m,0}} \quad (9.5.16)$$

Recalling that  $\vec{f} = [f_1 \ f_2 \ f_3 \ f_4]^T$  we obtain the following matrices elements<sup>||</sup>

$$\frac{\partial f_1}{\partial \vec{x}} = \vec{0} \quad (9.5.17)$$

$$\frac{\partial f_2}{\partial x} = 0 \quad (9.5.18)$$

$$\frac{\partial f_2}{\partial v} \Big|_{\vec{x}_0, f_{m,0}} = -\frac{b_0(J_1 + m_1 l^2)}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.19)$$

$$\frac{\partial f_2}{\partial \vartheta} \Big|_{\vec{x}_0, f_{m,0}} = -\frac{g m_1^2 l^2}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.20)$$

$$\frac{\partial f_2}{\partial \omega} \Big|_{\vec{x}_0, f_{m,0}} = \frac{b_1 m_1 l}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.21)$$

$$\frac{\partial f_3}{\partial \vec{x}} = \vec{0} \quad (9.5.22)$$

$$\frac{\partial f_4}{\partial x} = 0 \quad (9.5.23)$$

$$\frac{\partial f_4}{\partial v} \Big|_{\vec{x}_0, f_{m,0}} = \frac{b_0 m_1 l}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.24)$$

$$\frac{\partial f_4}{\partial \vartheta} \Big|_{\vec{x}_0, f_{m,0}} = \frac{g m_1 (m_0 + m_1) l}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.25)$$

$$\frac{\partial f_4}{\partial \omega} \Big|_{\vec{x}_0, f_{m,0}} = -\frac{b_1 (m_0 + m_1)}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.26)$$

$$\frac{\partial f_2}{\partial f_m} \Big|_{\vec{x}_0, f_{m,0}} = \frac{J_1 + m_1 l^2}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.27)$$

$$\frac{\partial f_4}{\partial f_m} \Big|_{\vec{x}_0, f_{m,0}} = -\frac{m_1 l}{(m_0 + m_1)(J_1 + m_1 l^2) - m_1^2 l^2} \quad (9.5.28)$$

---

<sup>||</sup>remind that

$$\frac{\partial}{\partial \vec{x}} \vec{f}(\vec{x}) \Big|_{\vec{x}_0, \vec{u}_0} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\vec{x}_0, \vec{u}_0} = \tilde{\mathbf{A}}, \quad \frac{\partial}{\partial \vec{u}} \vec{f}(\vec{x}) \Big|_{\vec{x}_0, \vec{u}_0} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix}_{\vec{x}_0, \vec{u}_0} = \tilde{\mathbf{B}}$$

Hence, the linearized model can be written as follows

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}f_m(t) + \tilde{\mathbf{E}}f_d(t) \\ \vec{y}(t) = \mathbf{C}\vec{x}(t) \end{cases} \quad (9.5.29)$$

where

$$\tilde{\mathbf{A}} = \frac{1}{\mathcal{D}} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -b_0(J_1 + m_1 l)^2 & -g(m_1 l^2) & b_1 m_1 l \\ 0 & 0 & 0 & 1 \\ 0 & b_0 m_1 l & g m_1 (m_0 + m_1) l & -b_1 (m_0 + m_1) \end{bmatrix} \quad (9.5.30)$$

$$\tilde{\mathbf{E}} = \tilde{\mathbf{B}} = \frac{1}{\mathcal{D}} \begin{bmatrix} 0 \\ (J_1 + m_1 l^2) \\ 0 \\ -m_1 l \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \vec{C}_1 \\ \vec{C}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (9.5.31)$$

where  $\mathcal{D} = (m_0 + m_1)(J_1 + m_1 l^2) - (m_1 l)^2$ .

### 9.5.2 Control Implementation

The control architecture which is here intended to implement is reported in Figure 9.26. The control architecture is made by the following parts

- A state feedback with integrator
- A state observer

as second step, will be also proposed an improvement of the above architecture which consist of

- A state feedback with integrator
- A load compensator
- A state observer which include the load estimator

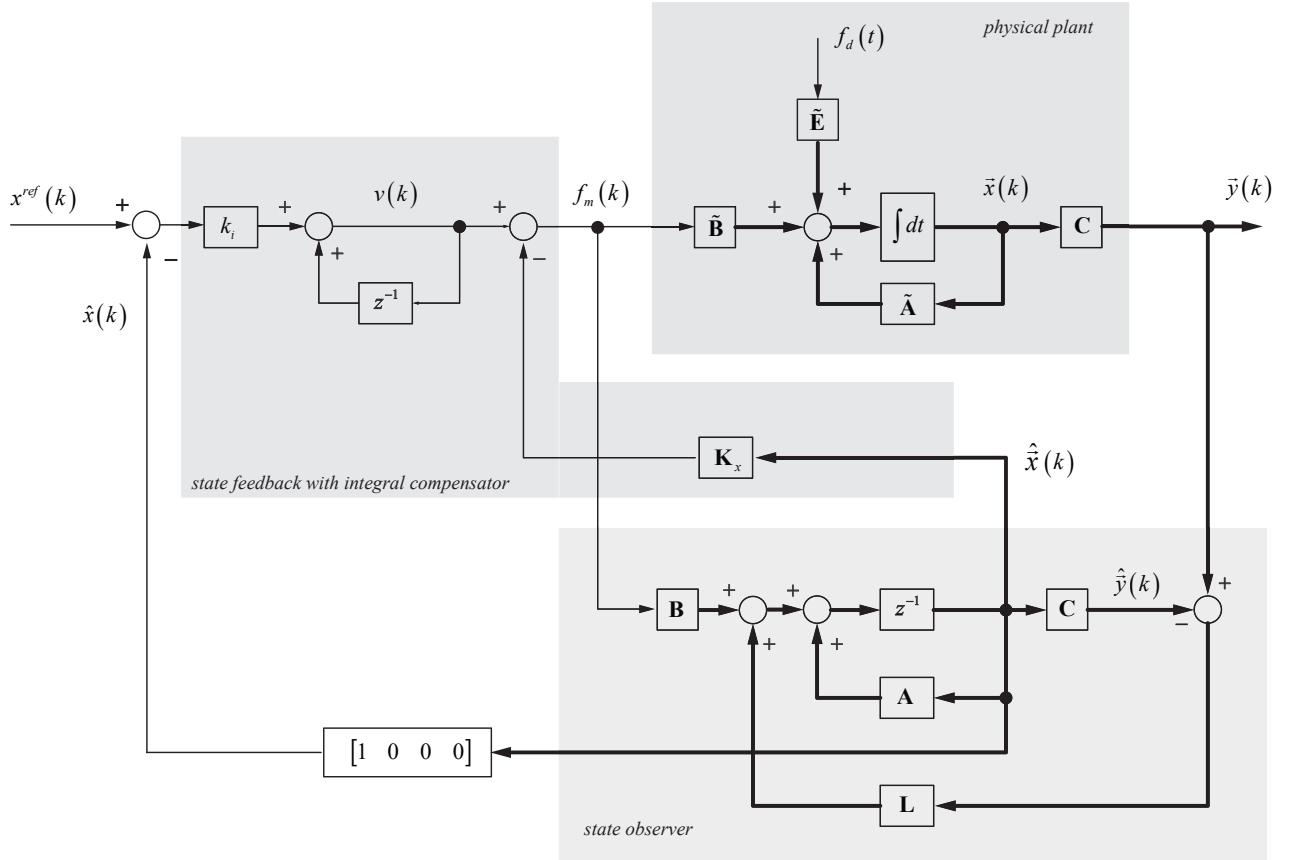


Figure 9.26: Control architecture for *pendulum on cart* problem - discrete time domain.

The control system will be implemented in discrete time domain, hence, as first step, we have to calculate the matrices which correspond to the equivalent discrete time domain representation

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}} \vec{x}(t) + \tilde{\mathbf{B}} f_m(t) \\ \vec{y}(t) = \tilde{\mathbf{C}} \vec{x}(t) \end{cases} \Rightarrow \begin{cases} \vec{x}(k+1) = \mathbf{A} \vec{x}(k) + \mathbf{B} f_m(k) \\ \vec{y}(k) = \mathbf{C} \vec{x}(k) \end{cases} \quad (9.5.32)$$

The matrices  $\mathbf{A}$  and  $\mathbf{B}$  can be determined as follows

$$\begin{aligned} \mathbf{A} &= \mathbf{I} + t_s \tilde{\mathbf{A}} \\ \mathbf{B} &= \tilde{\mathbf{B}} t_s \end{aligned} \quad (9.5.33)$$

The state feedback matrix  $\mathbf{K}_x$  as well as the integral gain  $k_i$  will be derived by pole placement as follows.

The state feedback with integrator design will reduce to the calculation of the matrix

$\begin{bmatrix} \mathbf{K}_x & k_i \end{bmatrix}$  as follows

$$\begin{bmatrix} \mathbf{K}_x & k_i \end{bmatrix} = \left\{ \mathbf{K}' + \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \right\} \begin{bmatrix} \mathbf{A} - \mathbf{I}_4 & \mathbf{B} \\ \vec{C}_1 \mathbf{A} & \vec{C}_1 \mathbf{B} \end{bmatrix}^{-1} \quad (9.5.34)$$

where  $\mathbf{K}'$  is given by Ackermann's formula using the system  $(\mathbf{A}_e, \mathbf{B}_e)$ , where

$$\mathbf{A}_e = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \vec{0} & 0 \end{bmatrix} \quad \mathbf{B}_e = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T \quad (9.5.35)$$

and

$$\mathbf{K}' = \text{acker}\left(\mathbf{A}_e, \mathbf{B}_e, \exp(t_s \vec{p}_{sfi})\right) \quad (9.5.36)$$

Regarding the observer block we can consider the following system equation:

$$\begin{aligned} \hat{\vec{x}}(k+1) &= \mathbf{A} \hat{\vec{x}}(k) + \mathbf{B} f_m(k) + \mathbf{L} (\vec{y}(k) - \hat{\vec{y}}(k)) \\ \hat{\vec{y}}(k) &= \mathbf{C} \hat{\vec{x}}(k) \end{aligned} \quad (9.5.37)$$

Where  $\hat{\vec{y}}(k) = [\hat{x}(k) \quad \hat{\vartheta}(k)]^T$ .

The matrix  $\mathbf{L}$  is calculated applying the pole placement method to the system  $(\mathbf{A}, \mathbf{C})$ .

$$\mathbf{L} = \left[ \text{acker}\left(\mathbf{A}^T, \mathbf{C}^T, \exp(t_s \vec{p}_{so})\right) \right]^T \quad (9.5.38)$$

Simulation results of this preliminary control architecture can be seen in Figure 9.28.

### 9.5.3 Control Implementation with load observer

Now we propose an extension of the fir control architecture, in particular a modelization of the disturbance will be proposed and it will be integrated into a extended model which will be used to estimate both the state of the original system as well as the unknown load.

In order to implement the  $f_d(t)$  estimator, we assume that the dynamic of the disturbance  $f_d(t)$  is slower than the main dynamic of the plant, that can be modelized by the differential equation

$$\dot{f}_d(t) = 0 \quad (9.5.39)$$

The new model will be a merge of system of Eq. (9.5.42) and system of Eq. (9.5.39), that mean the new state vector will be

$$\vec{x}_{le}(t) = [x(t) \quad v(t) \quad \vartheta(t) \quad \omega(t) \quad f_d(t)]^T \quad (9.5.40)$$

and the new representation of the inverted pendulum on cart will be as follows

$$\begin{cases} \dot{\vec{x}}_{le}(t) = \tilde{\mathbf{A}}_{le}\vec{x}_{le}(t) + \tilde{\mathbf{B}}_{le}f_m(t) \\ \vec{y}(t) = \mathbf{C}_{le}\vec{x}_{le}(t) \end{cases} \quad (9.5.41)$$

where  $\vec{x}_{le}(t) = [x(t) \ v(t) \ \vartheta(t) \ \omega(t) \ f_d(t)]^T$  and where

$$\tilde{\mathbf{A}}_{le} = \frac{1}{\mathcal{D}} \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -b_0(J_1 + m_1l)^2 & -g(m_1l^2) & b_1m_1l & (J_1 + m_1l^2) \\ 0 & 0 & 0 & 1 & 0 \\ 0 & b_0m_1l & gm_1(m_0 + m_1)l & -b_1(m_0 + m_1) & -m_1l \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.5.42)$$

$$\tilde{\mathbf{B}}_{le} = \frac{1}{\mathcal{D}} \begin{bmatrix} 0 \\ (J_1 + m_1l^2) \\ 0 \\ -m_1l \\ 0 \end{bmatrix}, \quad \mathbf{C}_{le} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (9.5.43)$$

where  $\mathcal{D} = (m_0 + m_1)(J_1 + m_1l^2) - (m_1l)^2$ .

$$\begin{aligned} \mathbf{A}_{le} &= \mathbf{I} + t_s \tilde{\mathbf{A}}_{le} \\ \mathbf{B}_{le} &= \tilde{\mathbf{B}}_{le} t_s \end{aligned} \quad (9.5.44)$$

The system we have built just now will be used as plant for an observer which will generate an estimation of the original state vector plus an estimation of the load, as follows

$$\begin{aligned} \hat{\vec{x}}_{le}(k+1) &= \mathbf{A}_{le} \hat{\vec{x}}_{le}(k) + \mathbf{B}_{le} f_m(k) + \mathbf{L}_{le} (\vec{y}(k) - \hat{\vec{y}}(k)) \\ \hat{\vec{y}}(k) &= \mathbf{C}_{le} \hat{\vec{x}}_{le}(k) \end{aligned} \quad (9.5.45)$$

Where  $\hat{\vec{y}}(k) = [\hat{x}(k) \ \hat{\vartheta}(k)]^T$ .

The matrix  $\mathbf{L}_{le}$  is calculated applying the pole placement method to the system  $(\mathbf{A}_{le}, \mathbf{C}_{le})$ .

$$\mathbf{L}_{le} = \left[ acker \left( \mathbf{A}_{le}^T, \mathbf{C}_{le}^T, \exp(t_s \vec{p}_{le}^{so}) \right) \right]^T \quad (9.5.46)$$

Simulation results of this preliminary control architecture can be seen in Figure 9.28.

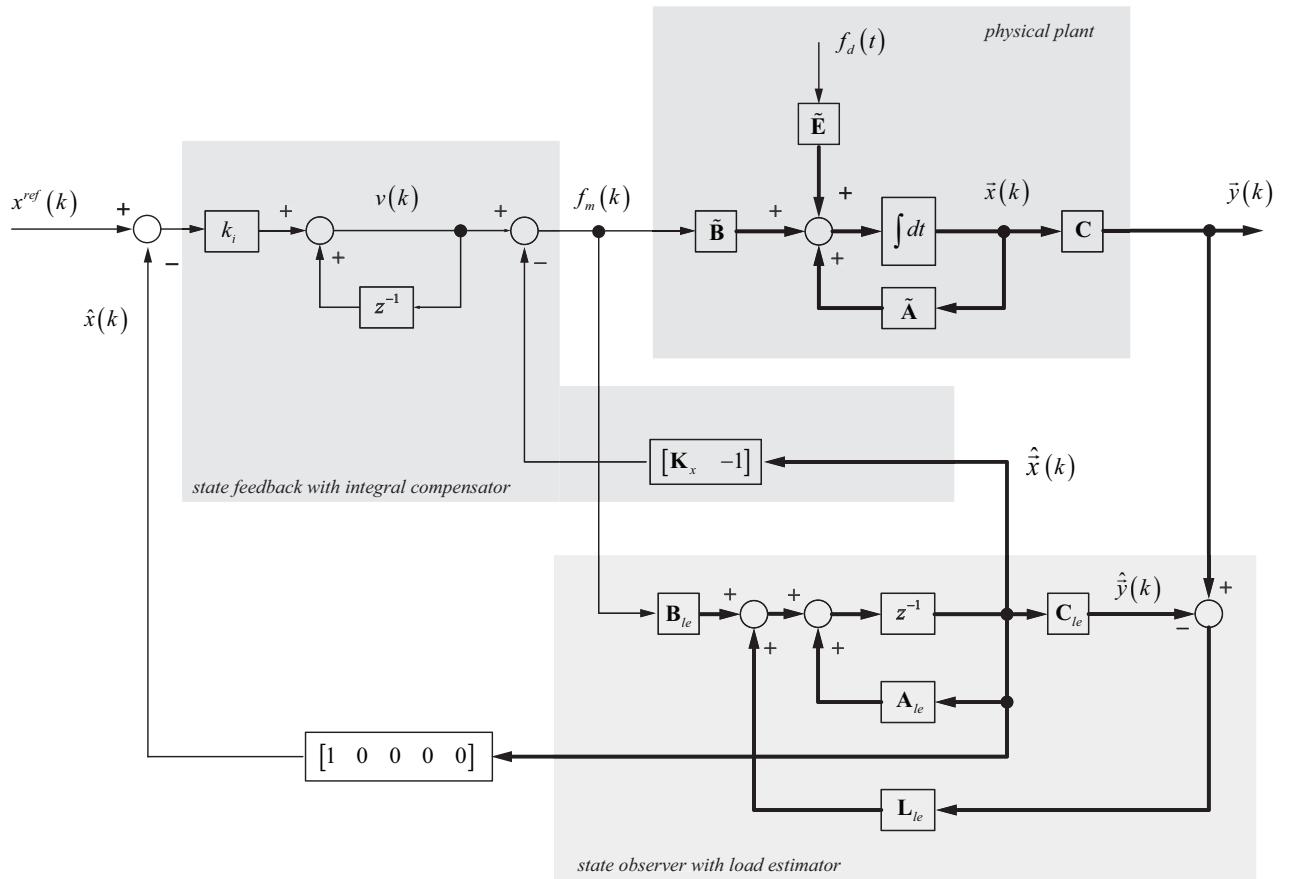
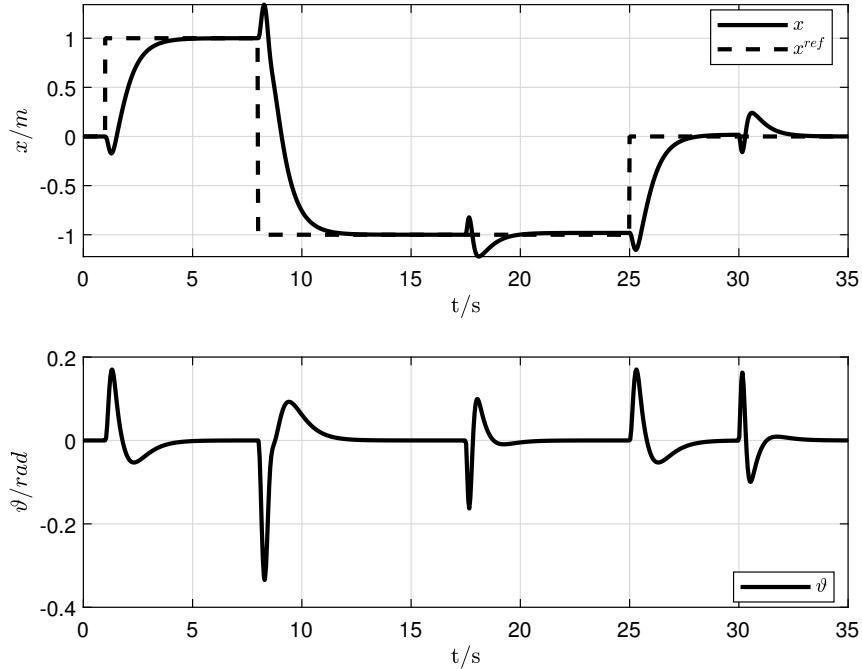
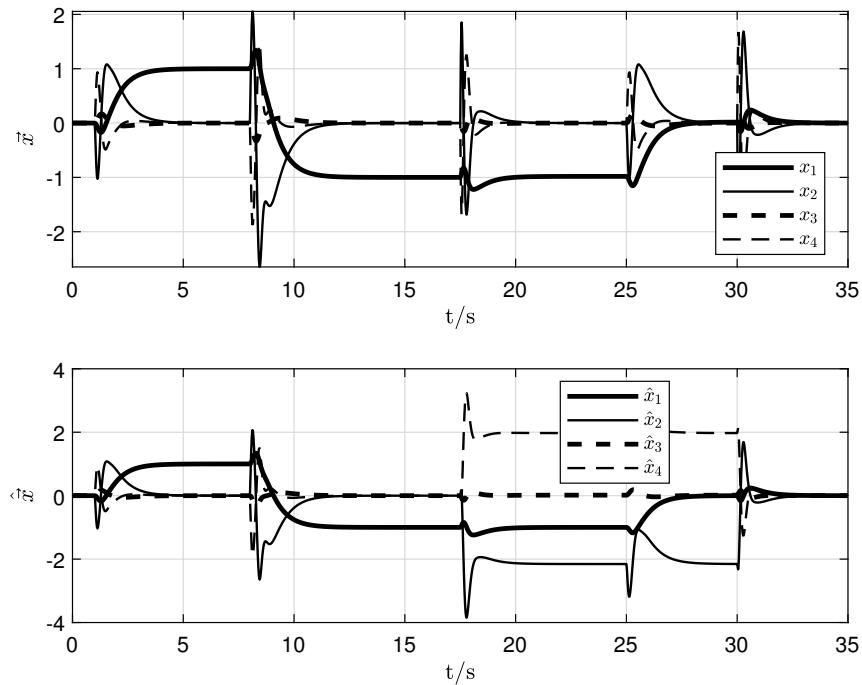


Figure 9.27: Control architecture for *pendulum on cart* problem with load estimator - discrete time domain.

#### 9.5.4 Simulation Results



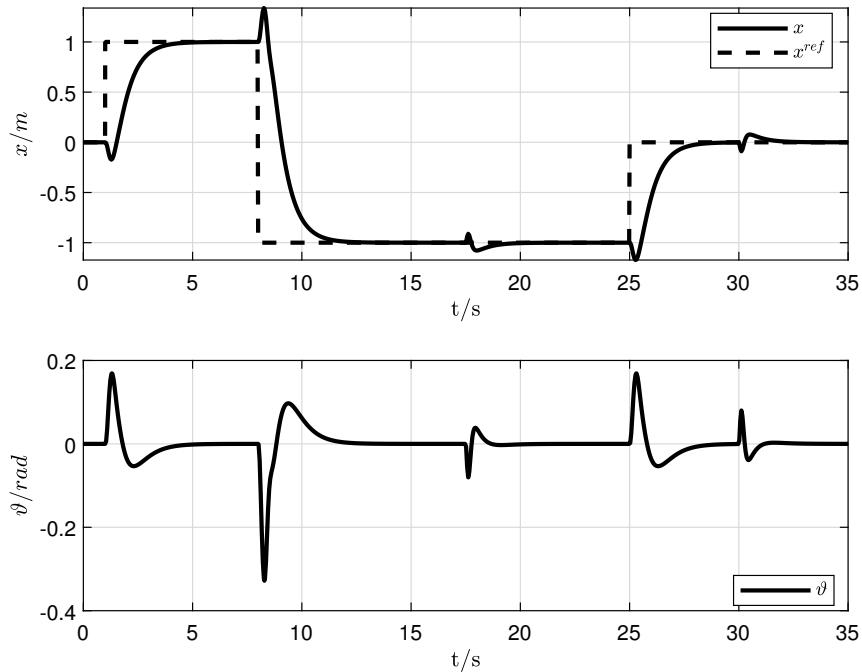
(a) Upper subplot: tracking performance for a given position set-point. The position reference  $x^{ref}$  is in dashed. Bottom subplot: inverter pendulum angle behaviour during cart movement.



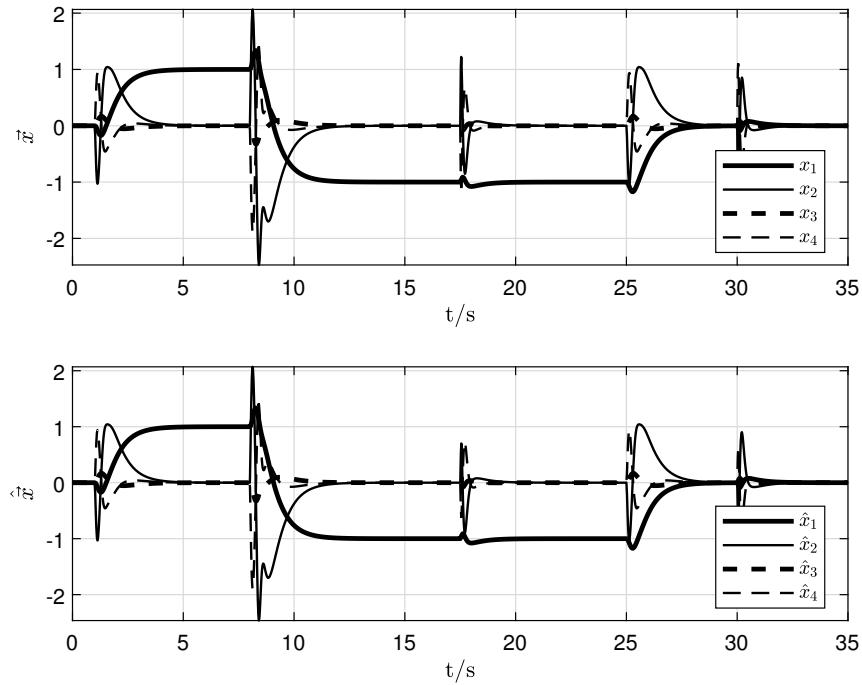
(b) Upper subplot: state vector dynamics during cart and pendulum movement. Bottom subplot: estimation of the above state vector.

Figure 9.28: Simulation results for the case with no load estimation.

---



(a) Upper subplot: tracking performance for a given position set-point. The position reference  $x^{ref}$  is in dashed. Bottom subplot: inverter pendulum angle behaviour during cart movement.



(b) Upper subplot: state vector dynamics during cart and pendulum movement. Bottom subplot: estimation of the above state vector.

Figure 9.29: Simulation results for the case with load estimation and compensation.

## 9.6 Segway

In this section we propose an application of state feedback and state observer control applied to a segway. The system is represented in Figure 9.30.

The scope of the project is to design a controller which permits to move the segway keeping the vertical bar as close as possible to its vertical equilibrium point.

### 9.6.1 Model Derivation

According Figure 9.30 the system is governed (control input) by the torque  $\tau_m(t)$ .

The generalized coordinates which describe the system are

- The wheel angle  $\alpha(t)$
- The bar angle  $\vartheta(t)$

Both, the wheel and the bar of respectively of mass  $m_1$  and  $m_2$  and inertia  $J_1$  and  $J_2$  are subjected to viscosity force described by the coefficient  $b_1$  and  $b_2$  respectively.

To derive the dynamical equations which govern the system, we apply the Euler-Lagrange equations.

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = Q \quad i = 1, 2, \dots, n \quad (9.6.1)$$

where the Lagrange function  $\mathcal{L}$  is

$$\mathcal{L} = E_{kin} - E_{pot} \quad (9.6.2)$$

To derive the total kinetic energy we first write the vector speed for both masses which are subject to move

Let  $\vec{v}_1$  be the speed of the wheel. It can be expressed as follows

$$\vec{v}_1 = r_1 \dot{\alpha} \vec{i} \quad (9.6.3)$$

Let  $\vec{v}_2$  be the speed of the bar. It can be expressed as follows

$$\vec{v}_2 = \left( r_1 \dot{\alpha} + r_2 \dot{\vartheta} \cos \vartheta \right) \vec{i} - r_2 \dot{\vartheta} \sin \vartheta \vec{j} \quad (9.6.4)$$

The total kinetic energy can be written as follows

$$E_{kin} = \frac{1}{2} m_1 v_1^2 + \frac{1}{2} J_1 \dot{\alpha}^2 + \frac{1}{2} m_2 v_2^2 + \frac{1}{2} J_2 \dot{\vartheta}^2 \quad (9.6.5)$$

where

$$v_1^2 = (r_1 \dot{\alpha})^2 \quad (9.6.6)$$

and

$$v_2^2 = (r_1 \dot{\alpha} + r_2 \dot{\vartheta} \cos \vartheta)^2 + (r_2 \dot{\vartheta} \sin \vartheta)^2 \quad (9.6.7)$$

The potential energy can be expressed as follows

$$E_{pot} = m_2 g r_2 \cos \vartheta. \quad (9.6.8)$$

The resulting Lagrangian is

$$\mathcal{L} = \frac{1}{2} \left[ (m_1 + m_2) r_1^2 + J_1 \right] \dot{\alpha}^2 + \frac{1}{2} \left[ m_2 r_2^2 + J_2 \right] \dot{\vartheta}^2 + m_2 r_2 (r_1 \dot{\alpha} \dot{\vartheta} - g) \quad (9.6.9)$$

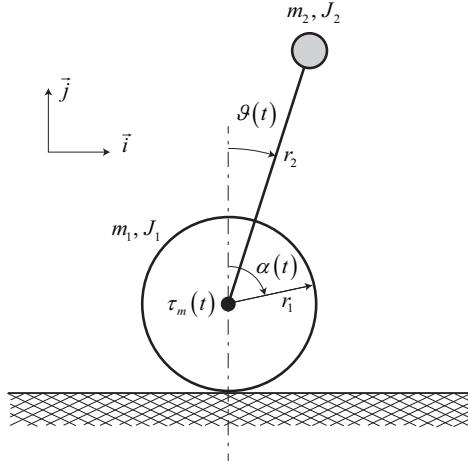


Figure 9.30: Segway.

Now, we suppose to apply Eq. (9.6.1) for the variables  $\alpha$  and  $\vartheta$  and we obtain the following dynamic equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\alpha}} - \frac{\partial \mathcal{L}}{\partial \alpha} = \tau_m - b_1 \dot{\alpha} - b_2 (\dot{\alpha} + \dot{\vartheta}) \quad (9.6.10)$$

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\vartheta}} - \frac{\partial \mathcal{L}}{\partial \vartheta} = \tau_m - b_1 \dot{\alpha} \quad (9.6.11)$$

which results in following set of equations

$$\begin{cases} \left[ (m_1 + m_2) r_1^2 + J_1 \right] \ddot{\alpha} + m_2 r_1 r_2 \left[ \ddot{\vartheta} \cos \vartheta - \dot{\vartheta}^2 \sin \vartheta \right] + (b_1 + b_2) \dot{\alpha} + b_2 \dot{\vartheta} - \tau_m = 0 \\ (m_2 r_2^2 + J_2) \ddot{\vartheta} + m_2 r_1 r_2 \left[ \ddot{\alpha} \cos \vartheta + g/r_1 \sin \vartheta \right] + b_2 \dot{\alpha} + b_2 \dot{\vartheta} + \tau_m = 0 \end{cases} \quad (9.6.12)$$

solving the system of Eq. (9.6.12) respect to  $\ddot{\alpha}$  and  $\ddot{\vartheta}$  we obtain the following system

$$\left\{ \begin{array}{l} \dot{\alpha} = \omega_1 \\ \dot{\omega}_1 = \frac{2}{\mathcal{D}} \left[ m_2 r_1 r_2 \cos \vartheta \left( \tau_m + b_2 \omega_1 + b_2 \omega_2 + g m_2 r_2 \sin \vartheta \right) - 2 \left( J_2 + m_2 r_2^2 \right) \right. \\ \quad \left. \left( - \tau_m + b_1 \omega_1 + b_2 \omega_1 + b_2 \omega_2 - m_2 r_1 r_2 \omega_2^2 \sin \vartheta \right) \right] \\ \dot{\vartheta} = \omega_2 \\ \dot{\omega}_2 = - \frac{2}{\mathcal{D}} \left[ \left[ J_1 + (m_1 + m_2) r_1^2 \right] \left( \tau_m + b_2 \omega_1 + b_2 \omega_2 + g m_2 r_2 \sin \vartheta \right) + \right. \\ \quad \left. m_2 r_1 r_2 \cos \vartheta \left( \tau_m - b_1 \omega_1 - b_2 \omega_1 - b_2 \omega_2 + m_2 r_1 r_2 \omega_2^2 \sin \vartheta \right) \right] \end{array} \right. \quad (9.6.13)$$

where

$$\mathcal{D} = 2J_1(J_2 + m_2 r_2^2) + r_1^2 \left[ 2J_2(m_1 + m_2) + m_2(2m_1 + m_2)r_2^2 \right] - m_2^2 r_1^2 r_2^2 \cos 2\vartheta \quad (9.6.14)$$

Let's now derive the linearized model around the equilibrium point

$$\vec{x}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \tau_m^0 = 0 \quad (9.6.15)$$

Lets to define the transition function as well

$$\left\{ \begin{array}{l} f_1(\alpha, \omega_1, \vartheta, \omega_2) = \omega_1 \\ f_2(\alpha, \omega_1, \vartheta, \omega_2) = \frac{2}{\mathcal{D}} \left[ m_2 r_1 r_2 \cos \vartheta \left( \tau_m + b_2 \omega_1 + b_2 \omega_2 + g m_2 r_2 \sin \vartheta \right) \right. \\ \quad \left. - 2 \left( J_2 + m_2 r_2^2 \right) \left( - \tau_m + b_1 \omega_1 + b_2 \omega_1 + b_2 \omega_2 - m_2 r_1 r_2 \omega_2^2 \sin \vartheta \right) \right] \\ f_3(\alpha, \omega_1, \vartheta, \omega_2) = \omega_2 \\ f_4(\alpha, \omega_1, \vartheta, \omega_2) = - \frac{2}{\mathcal{D}} \left[ \left[ J_1 + (m_1 + m_2) r_1^2 \right] \left( \tau_m + b_2 \omega_1 + b_2 \omega_2 + g m_2 r_2 \sin \vartheta \right) + \right. \\ \quad \left. m_2 r_1 r_2 \cos \vartheta \left( \tau_m - b_1 \omega_1 - b_2 \omega_1 - b_2 \omega_2 + m_2 r_1 r_2 \omega_2^2 \sin \vartheta \right) \right] \end{array} \right. \quad (9.6.16)$$

and we calculate the following terms\*\*

$$\frac{\partial f_1}{\partial \omega_1} \Big|_{\vec{x}_0, \tau_m^0} = 1, \quad \frac{\partial f_3}{\partial \omega_2} \Big|_{\vec{x}_0, \tau_m^0} = 1 \quad (9.6.17)$$

$$\frac{\partial f_2}{\partial \omega_1} \Big|_{\vec{x}_0, \tau_m^0} = \frac{1}{\mathcal{D}_0} \left[ 2b_2 m_2 r_1 r_2 - 2(b_1 + b_2)(J_2 + m_2 r_2^2) \right] \quad (9.6.18)$$

$$\frac{\partial f_2}{\partial \vartheta} \Big|_{\vec{x}_0, \tau_m^0} = \frac{1}{\mathcal{D}_0} \left[ 2g m_2^2 r_1 r_2^2 \right] \quad (9.6.19)$$

$$\frac{\partial f_2}{\partial \omega_2} \Big|_{\vec{x}_0, \tau_m^0} = \frac{1}{\mathcal{D}_0} \left[ 2b_2 (m_2 r_1 r_2 - J_2 - m_2 r_2^2) \right] \quad (9.6.20)$$

$$\frac{\partial f_4}{\partial \omega_1} \Big|_{\vec{x}_0, \tau_m^0} = \frac{1}{\mathcal{D}_0} \left[ -2b_2 [J_1(m_1 + m_2)r_1^2] + 2(b_1 + b_2)m_2 r_1 r_2 \right] \quad (9.6.21)$$

$$\frac{\partial f_4}{\partial \vartheta} \Big|_{\vec{x}_0, \tau_m^0} = \frac{1}{\mathcal{D}_0} \left[ -2g m_2 r_2 [J_1 + (m_1 + m_2)r_1^2] \right] \quad (9.6.22)$$

$$\frac{\partial f_4}{\partial \omega_2} \Big|_{\vec{x}_0, \tau_m^0} = \frac{1}{\mathcal{D}_0} \left[ -2[b_2(J_1 + (m_1 + m_2)r_1^2) - b_2 m_2 r_1 r_2] \right] \quad (9.6.23)$$

$$\frac{\partial f_2}{\partial \tau_m} \Big|_{\vec{x}_0, \tau_m^0} = \frac{2(J_2 + m_2 r_1 r_2 + m_2 r_2^2)}{\mathcal{D}_0} \quad (9.6.24)$$

$$\frac{\partial f_4}{\partial \tau_m} \Big|_{\vec{x}_0, \tau_m^0} = \frac{-2(J_1 + (m_1 + m_2)r_1^2 + m_2 r_2 r_2)}{\mathcal{D}_0} \quad (9.6.25)$$

where

$$\mathcal{D}_0 = -m_2^2 r_1^2 r_2^2 + 2J_1(J_2 + m_2 r_2^2) + r_1^2 [2J_2(m_1 + m_2) + m_2(2m_1 + m_2)r_2^2] \quad (9.6.26)$$

Using the results from Eq. (9.6.17) to Eq. (9.6.25) we can build the equivalent linear model

---

\*\*remind that

$$\frac{\partial}{\partial \vec{x}} \vec{f}(\vec{x}) \Big|_{\vec{x}_0, \vec{u}_0} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\vec{x}_0, \vec{u}_0} = \tilde{\mathbf{A}}, \quad \frac{\partial}{\partial \vec{u}} \vec{f}(\vec{x}) \Big|_{\vec{x}_0, \vec{u}_0} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix}_{\vec{x}_0, \vec{u}_0} = \tilde{\mathbf{B}}$$

around the equilibrium point

$$\begin{aligned}\dot{\vec{x}}(t) &= \tilde{\mathbf{A}}\vec{x}(t) + \tilde{\mathbf{B}}\tau_m(t) \\ \vec{y}(t) &= \mathbf{C}\vec{x}(t)\end{aligned}\quad (9.6.27)$$

where

$$\vec{x}(t) = \begin{bmatrix} \alpha(t) \\ \omega_1(t) \\ \vartheta(t) \\ \omega_2(t) \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (9.6.28)$$

The design of the controller is similar to what has been developed for the inverted pendulum on cart.

## 9.7 Lithium-ion Battery Management by the use of Extended Kalman Filter

In the following section we will propose a control problem concerning the estimation of the state of charge of a battery, the main characteristic of the control can be here summarized

- The lithium-ion battery model will be presented
- The state of charge estimation will be implemented by the use the EKF
- The use of an additional EKF block to estimate the capacity of the battery will presented

The design steps can be also summarized as well

- Derivation of the linearized model along the estimated state trajectory
- Discretization of the battery

### Nomenclature

- $z(t)$  [per unit] state of charge
- $Q$  [A h] capacity of the battery
- $h(t)$  [per unit] hysteresys of the output voltage of the battery: same state of charge can result in different output voltages.
- $C_1$  [F] double layer membrane capacity
- **OCV** [V] open circuit voltage, this term is directly affected by the state of the charge plus additional parameter (like temperature) which are not here taken into account. The description of the open circuit voltage as function of the state of charge is made by a transcendental analytical function, but can be also tabulated.

Let's start writing down the continuous time domain model of the battery, as well

**Enhanced self-correcting (ESC) model state equations - continuous time domain**

$$\begin{cases} \frac{dz(t)}{dt} = -\frac{1}{Q}i(t) \\ \frac{di_{R_1}(t)}{dt} = -\frac{1}{R_1 C_1}(i_{R_1}(t) - i(t)) \\ \frac{dh(t)}{dt} = -\left|\frac{i(t)}{Q}\right| h(t) - \left|\frac{i(t)}{Q}\right| \text{sign}(i) \end{cases} \quad (9.7.1)$$

$$v(t) = \mathfrak{N} \cdot \mathbf{OCV}(z(t)) + Mh(t) - i_{R_1}(t)R_1 - i(t)R_0 \quad (9.7.2)$$

$$\mathbf{OCV}(z) = E_{-1} \exp(-z\alpha) + E_0 + E_1 z + E_2 z^2 + E_3 z^3 + E_{\log} \log(1-z)$$

where  $\mathfrak{N}$  is the number of cells connected in series.

where the equivalent discrete time representation is as follows

**Enhanced self-correcting (ESC) model state equations - discrete time domain**

$$\begin{cases} z[k+1] = z[k] - \frac{t_s}{Q} \eta[k] i[k] \\ i_{R_1}[k+1] = \exp\left(-\frac{t_s}{R_1 C_1}\right) i_{R_1}[k] + \left[1 - \exp\left(-\frac{t_s}{R_1 C_1}\right)\right] i[k] \\ h[k+1] = \exp\left(-\left|\frac{i[k] t_s}{Q}\right|\right) h[k] - \left[1 - \exp\left(-\left|\frac{i[k] t_s}{Q}\right|\right)\right] \text{sign}(i[k]) \\ v[k] = \mathfrak{N} \cdot \mathbf{OCV}(z[k]) + Mh[k] - R_1 i_{R_1}[k] - R_0 i[k] \end{cases} \quad (9.7.3)$$

where  $\mathfrak{N}$  is the number of cells connected in series, and

$$\mathbf{OCV}(z) = E_{-1} \exp(-z\alpha) + E_0 + E_1 z + E_2 z^2 + E_3 z^3 + E_{\log} \log(1-z) \quad (9.7.4)$$

where  $z$  belong to an open set as  $z \in (0, 1)$ , and

- $E_{-1} = -1.031 \text{ V}$	- $E_3 = 0 \text{ V}$	- $R_1 = 15 \text{ m}\Omega$
- $\alpha = 35$	- $E_{\log} = -0.05 \text{ V}$	- $C_1 = 10 \text{ F}$
- $E_0 = 3.685 \text{ V}$	- $\mathfrak{N} = 185$	- $Q = 400 \text{ A h}$
- $E_1 = 0.015 \text{ V}$	- $M = 125 \text{ V}$	
- $E_2 = 0 \text{ V}$	- $R_0 = 15 \text{ m}\Omega$	

Figure 9.31 shows the equivalent electrical circuit of the battery model.

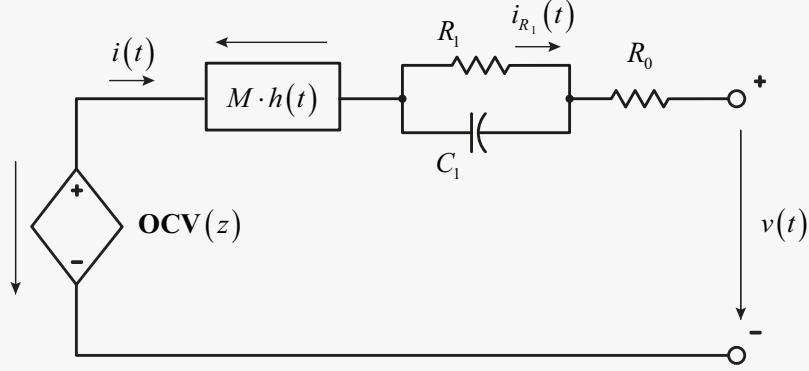


Figure 9.31: The enhanced self-correcting cell model equivalent circuit.

### 9.7.1 State Observer Implementation with EKF

Eq. (9.7.5) reports the set of equations necessary to implement the algorithm of EKF.

#### Extended Kalman Filter Algorithm

$$\hat{x}(k|k) = f[\hat{x}(k|k-1), u(k|k-1)] \quad (9.7.5)$$

$$\mathbf{P}(k|k) = \mathbf{A}(k|k-1)\mathbf{P}(k|k-1)\mathbf{A}(k|k-1)^T + \mathbf{Q} \quad (9.7.6)$$

$$\mathbf{L}(k) = \mathbf{P}(k|k)\mathbf{C}^T(k|k)\left(\mathbf{C}(k|k)\mathbf{P}(k|k)\mathbf{C}^T(k|k) + \mathbf{R}\right)^{-1} \quad (9.7.7)$$

$$\hat{x}(k+1|k) = \hat{x}(k|k) + \mathbf{L}(k)\left(y(k) - g[\vec{x}(k|k), u(y)(k|k-1)]\right) \quad (9.7.8)$$

$$\mathbf{P}(k+1|k) = (\mathbf{I} - \mathbf{L}(k)\mathbf{C}(k|k))\mathbf{P}(k|k) \quad (9.7.9)$$

where  $u(k|k-1) = i(k)$  and  $y(k) = i(k)$

Taking a look into Eq. (9.7.5) we can see we need to find an expression for the time variant matrices

$$\begin{aligned} & \mathbf{C}(k|k) \\ & \mathbf{A}(k|k-1). \end{aligned} \quad (9.7.10)$$

**Continuous time domain** The model represented by Eq. (9.7.1) can be rewritten as follows

$$\begin{aligned} \frac{d}{dt}\vec{x}(t) &= f[\vec{x}(t), u(t)] \\ y(t) &= g[\vec{x}(t), u(t)] \end{aligned} \quad (9.7.11)$$

where  $u(t) = i(t)$  and  $y(t) = v(t)$ . Eq. (9.7.11) can be rewritten as follows

$$\begin{aligned} \frac{d}{dt}\vec{x}(t) &= f[\vec{x}(t), u(t)] \Rightarrow \\ f[\vec{x}(t), u(t)] &= \begin{cases} \frac{d}{dt}z(t) = -\frac{t_s}{Q}i(t) \\ \frac{d}{dt}i_{R_1}(t) = -\frac{1}{R_1C_1}[i_{R_1}(t) - i(t)] \\ \frac{d}{dt}h(t) = -|\frac{i(t)}{Q}|h(t) - |\frac{i(t)}{Q}|\text{sign}(i(t)) \end{cases} \end{aligned} \quad (9.7.12)$$

$$y(t) = g[\vec{x}(t), u(t)] \Rightarrow$$

$$g[\vec{x}(t), u(t)] = \begin{cases} v(t) = \mathfrak{N} \cdot \mathbf{OCV}(z(t)) + Mh(t) - R_1i_{R_1}(t) - R_0i(t) \\ \mathbf{OCV}(z) = E_{-1} \exp(-z(t)\alpha) + E_0 + E_1z(t) + E_{\log} \log(1 - z(t)) \end{cases} \quad (9.7.13)$$

**Discrete time domain** The model represented by Eq. (9.7.3) can be rewritten as follows

$$\begin{aligned} \vec{x}(k+1) &= f[\vec{x}(k), u(k)] \\ y(k) &= g[\vec{x}(k), u(k)] \end{aligned} \quad (9.7.14)$$

where  $u(k) = i(k)$  and  $y(k) = v(k)$ . Eq. (9.7.14) can be rewritten as follows

$$\begin{aligned} \vec{x}(k+1) &= f[\vec{x}(k), u(k)] \Rightarrow \\ f[\vec{x}(k), u(k)] &= \begin{cases} z[k+1] = z[k] - \frac{t_s}{Q}\eta[k]i[k] \\ i_{R_1}[k+1] = \exp\left(-\frac{t_s}{R_1C_1}\right)i_{R_1}[k] + \left[1 - \exp\left(-\frac{t_s}{R_1C_1}\right)\right]i[k] \\ h[k+1] = \exp\left(-\left|\frac{i[k]t_s}{Q}\right|\right)h[k] - \left[1 - \exp\left(-\left|\frac{i[k]t_s}{Q}\right|\right)\right]\text{sign}(i[k]) \end{cases} \end{aligned} \quad (9.7.15)$$

$$y(k) = g[\vec{x}(k), u(k)] \Rightarrow$$

$$g[\vec{x}(k), u(k)] = \begin{cases} v[k] = \mathfrak{N} \cdot \mathbf{OCV}(z[k]) + Mh[k] - R_1i_{R_1}[k] - R_0i[k] \\ \mathbf{OCV}(z) = E_{-1} \exp(-z\alpha) + E_0 + E_1z + E_{\log} \log(1 - z) \end{cases} \quad (9.7.16)$$

The matrices of Eq.(9.7.10) can be determined as follows

$$\frac{\partial}{\partial \vec{x}} \vec{f}(\vec{x}) \Big|_{\vec{x}=\hat{\vec{x}}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\hat{\vec{x}}(t)} = \tilde{\mathbf{A}}(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{1}{R_1 C_1} & 0 \\ 0 & 0 & -\left|\frac{i(t)}{Q}\right| \end{bmatrix} = \tilde{\mathbf{A}}_1 |i(t)| \quad (9.7.17)$$

$$\begin{aligned} \frac{\partial}{\partial \vec{x}} g(\vec{x}) \Big|_{\vec{x}=\hat{\vec{x}}} &= \begin{bmatrix} \frac{\partial g}{\partial x_1} & \cdots & \frac{\partial g}{\partial x_n} \end{bmatrix}_{\hat{\vec{x}}(t)} = \tilde{\mathbf{C}}(t) = \\ &= \begin{bmatrix} -\alpha E_{-1} e^{-\alpha z(t)} + E_1 - E_{\log}/(1 - z(t)) & -R_1 & M \end{bmatrix} \end{aligned} \quad (9.7.18)$$

where  $\hat{\vec{x}}$  is the estimated state vector from the EKF.

The matrix  $\mathbf{A}(k|k-1)$  can be determined as follows

$$\begin{aligned} \tilde{\mathbf{A}}(t) \rightarrow \mathbf{A}(k|k-1) &= \mathbf{I} + t_s \tilde{\mathbf{A}}_1 |i(k)| + \frac{1}{2} \left[ t_s \tilde{\mathbf{A}}_1 |i(k)| \right]^2 + \cdots = \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & e^{-\frac{t_s}{R_1 C_1}} & 0 \\ 0 & 0 & e^{-\left|i(k)\right| \frac{t_s}{Q}} \end{bmatrix} \end{aligned} \quad (9.7.19)$$

The matrix  $\mathbf{C}(k|k)$  can be determined as follows

$$\begin{aligned} \mathbf{C}(t) &= \begin{bmatrix} -\alpha E_{-1} e^{-\alpha z(t)} + E_1 - E_{\log}/(1 - z(t)) & -R_1 & M \end{bmatrix} \rightarrow \\ &\rightarrow \mathbf{C}(k|k) = \begin{bmatrix} -\alpha E_{-1} e^{-\alpha z(k|k)} + E_1 - E_{\log}/(1 - z(k|k)) & -R_1 & M \end{bmatrix} \end{aligned} \quad (9.7.20)$$

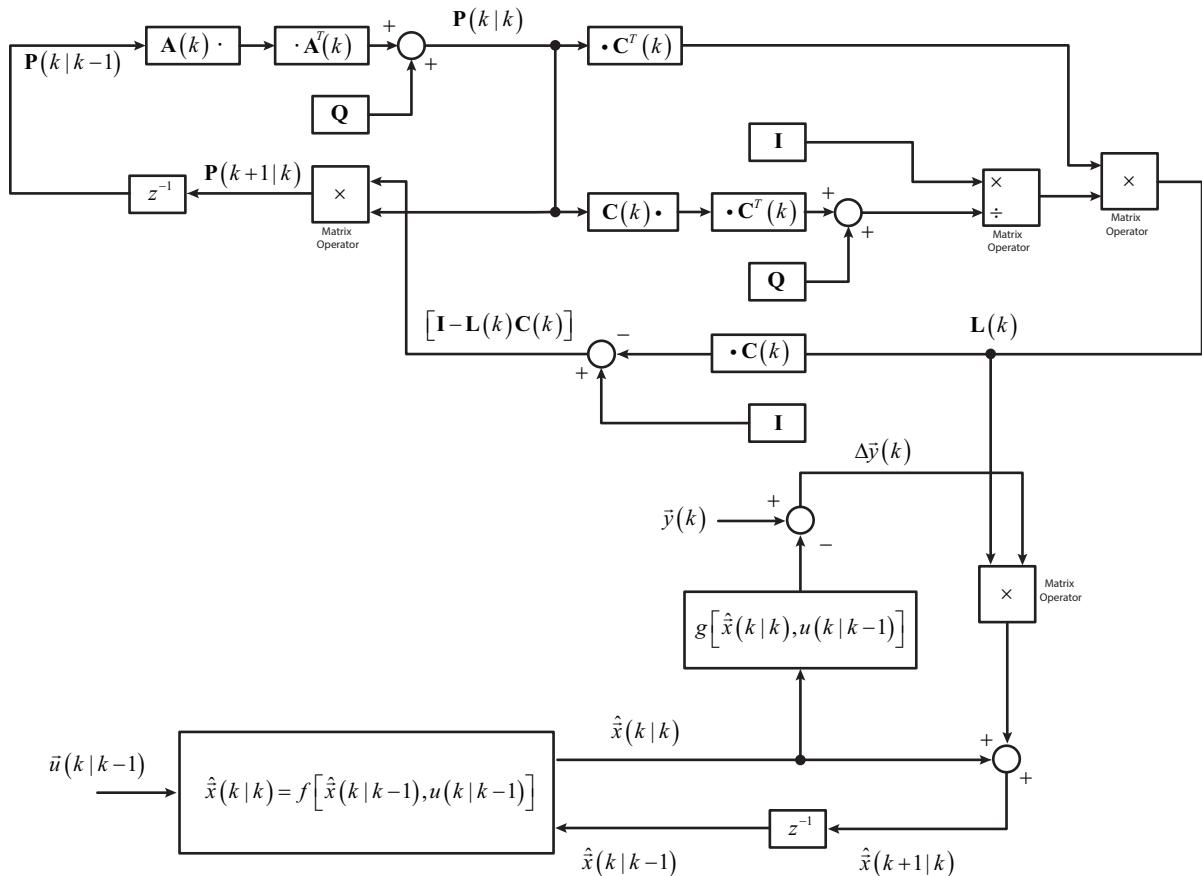
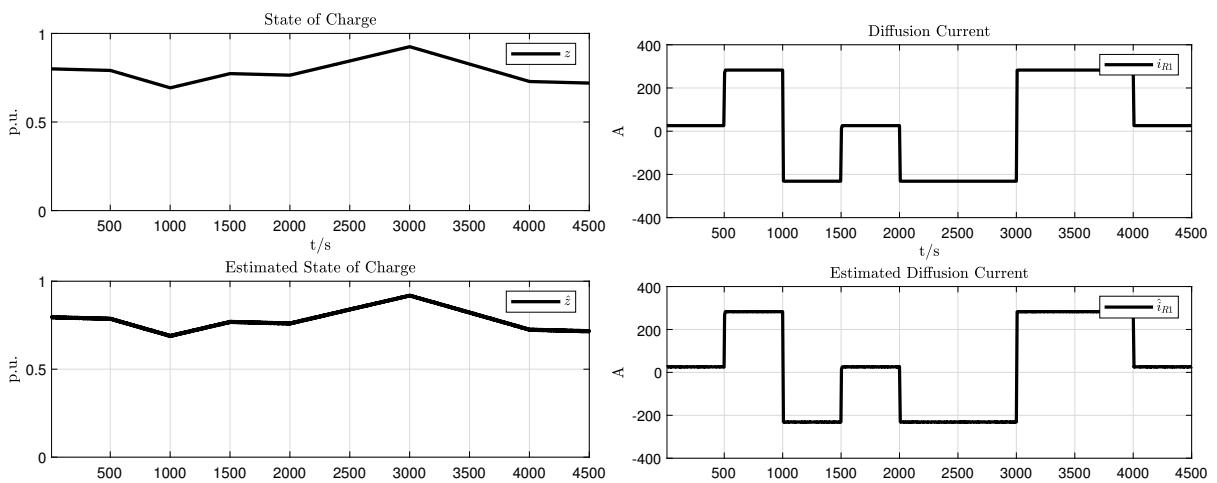


Figure 9.32: Extended Kalman Filter algorithm for the lithium ion battery.

### 9.7.2 Simulation results



(a) State of charge  $z(t)$  and estimated state of charge  $\hat{z}(t)$ .

(b) Diffusion current  $i_{R_1}$  and estimated diffusion current  $\hat{i}_{R_1}$ .

Figure 9.33: Lithium-ion battery manager - simulation results.

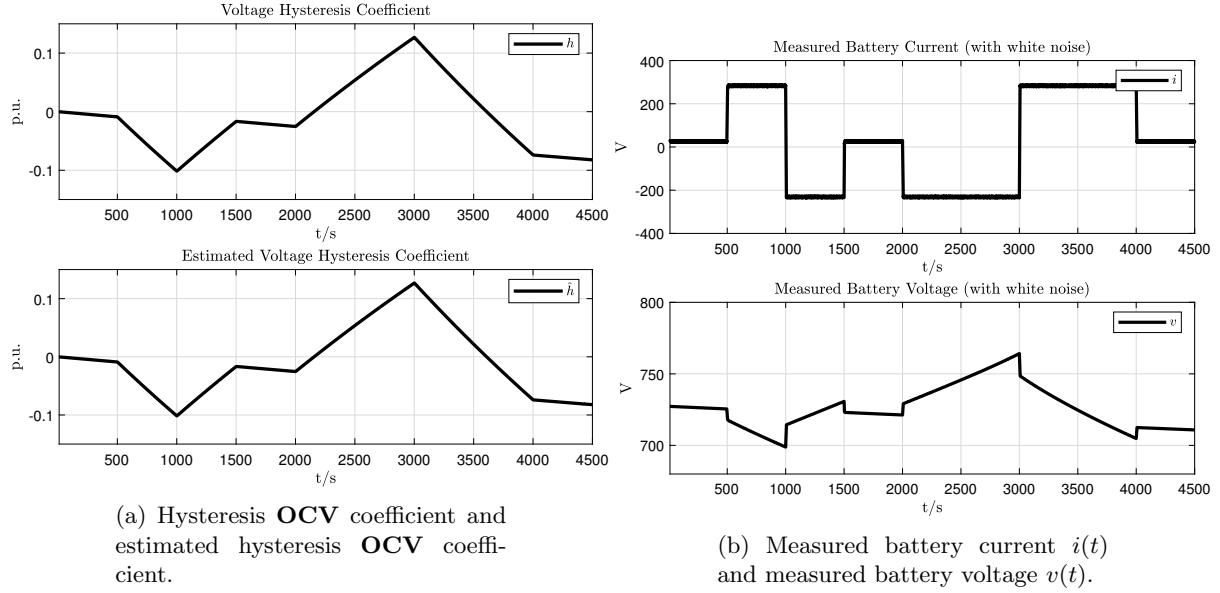


Figure 9.34: Lithium-ion battery manager - simulation results.

### 9.7.3 Adaptive State Observer Implementation with EKF

△

## 9.8 PMSM MPC with EKF

### 9.8.1 Introduction

In this section we are going to develop a model predictive control (MPC) for a permanent magnet synchronous motor (PMSM) by the use of an observer based on the extended Kalman filter theory (EKF).

Before to start with the design of the control structure a short remind of the PMSM mathematical model is here reported, see also chapter 8.

The  $\alpha\beta$  model representation of the PMSM is here reported

### PMSM Mathematical Model in Stationary Reference Frame

$$\left\{ \begin{array}{l} \vec{u}_{\alpha\beta}(t) - \mathbf{R}^{\alpha\beta} \vec{i}_{\alpha\beta}(t) - \mathbf{L}^{\alpha\beta} \frac{d\vec{i}_{\alpha\beta}(t)}{dt} - \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r = 0 \\ \frac{d\vec{\psi}_{\alpha\beta}^r(t)}{dt} = \omega(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{\psi}_{\alpha\beta}^r(t) \quad \vec{\psi}_{\alpha\beta}^r(0) = [\Psi^M \quad 0]^T \\ \frac{d}{dt} \omega_m(t) = \frac{1}{J} \left[ \frac{3}{2} p [\psi_{\alpha}^r(t)i_{\beta}(t) - \psi_{\beta}^r(t)i_{\alpha}(t)] - \tau_l(t) - b\omega_m(t) \right] \\ \frac{d}{dt} \theta(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (9.8.1)$$

or in a more explicit representation

$$\left\{ \begin{array}{l} \frac{d}{dt} i_{\alpha}(t) = -\frac{R_s}{L_{\alpha}} i_{\alpha}(t) + \frac{\omega(t)}{L_{\alpha}} \psi_{\beta}(t) + \frac{1}{L_{\alpha}} u_{\alpha}(t) \\ \frac{d}{dt} i_{\beta}(t) = -\frac{R_s}{L_{\beta}} i_{\beta}(t) - \frac{\omega(t)}{L_{\beta}} \psi_{\alpha}(t) + \frac{1}{L_{\beta}} u_{\beta}(t) \\ \frac{d}{dt} \psi_{\alpha}^r(t) = -\omega(t) \psi_{\beta}^r(t) \\ \frac{d}{dt} \psi_{\beta}^r(t) = \omega(t) \psi_{\alpha}^r(t) \\ \frac{d}{dt} \omega_m(t) = \frac{1}{J} \left[ \frac{3}{2} p [\psi_{\alpha}^r(t)i_{\beta}(t) - \psi_{\beta}^r(t)i_{\alpha}(t)] - \tau_l(t) - b\omega_m(t) \right] \\ \frac{d}{dt} \theta(t) = \omega(t) = p\omega_m(t) \end{array} \right. \quad (9.8.2)$$

where  $p$  is the number of poles pairs,  $b$  the bearing viscosity and  $\tau_l(t)$  the external and unknown load (disturbance).

Remind also the following nomenclature

- $i_{\alpha}(t)$  - is the  $\alpha$  component of the stator current.
- $i_{\beta}(t)$  - is the  $\beta$  component of the stator current.
- $u_{\alpha}(t)$  - is the  $\alpha$  component of the stator input voltage.
- $u_{\beta}(t)$  - is the  $\beta$  component of the stator input voltage.
- $\psi_{\alpha}^r(t)$  - is the  $\alpha$  component of the permanent magnet rotor flux linkage to the stator.
- $\psi_{\beta}^r(t)$  - is the  $\beta$  component of the permanent magnet rotor flux linkage to the stator.

- $\omega_m(t)$  - is the mechanical rotor speed.
- $\omega(t)$  - is the stator electrical pulsation related to the mechanical rotor speed by the relation  $\omega(t) = p\omega_m(t)$ .
- $p$  - is the number of pole pairs of the rotor.
- $\vartheta_m(t)$  - is the mechanical rotor position.
- $\vartheta(t)$  - is the stator electrical phase related to the mechanical rotor position by the relation  $\vartheta(t) = p\vartheta_m(t)$ .
- $\tau_m(t)$  is the electromagnetic torque generated by the PMSM
- $\tau_l(t)$  is the external load and is unknown.

### 9.8.2 Control design

In order to design a proper control structure we summarize here the main working hypothesis which we are takein into account

- The PMSM is isotropic, that means  $L_\alpha = L_\beta = L_s$ .
- The motor is driven by an inverter based on space vector modulation.
- The only stator current are available for the measure.
- The time constant of the mechanic system is twenty times greater than the time constant of the electrical and fluxes systems.

**The scope of the control design is to implement a rotor speed control with zero tracking error in steady state condition.**

As first draft we can draw the control architecture be made by the following substructures

- A speed control based on PI-control.
- A substructure which convert the torque reference (as output of the speed control) into a set of current references  $i_\alpha^{ref}$  and  $i_\beta^{ref}$ . Remind the motor is a three phase system and that means must be represented by two components.
- A current control based on model predictive (MPC) in  $\alpha\beta$  coordinates, see also Chapter 8.
- A rotor flux vector estimator based on the Extended Kalman Filter.
- A rotor speed estimator used to calculate  $\hat{\omega}$  from  $\hat{\vartheta}$  based on double integrator observer.

A schematic overview of the proposed control architecture is shown in Figure (9.35)

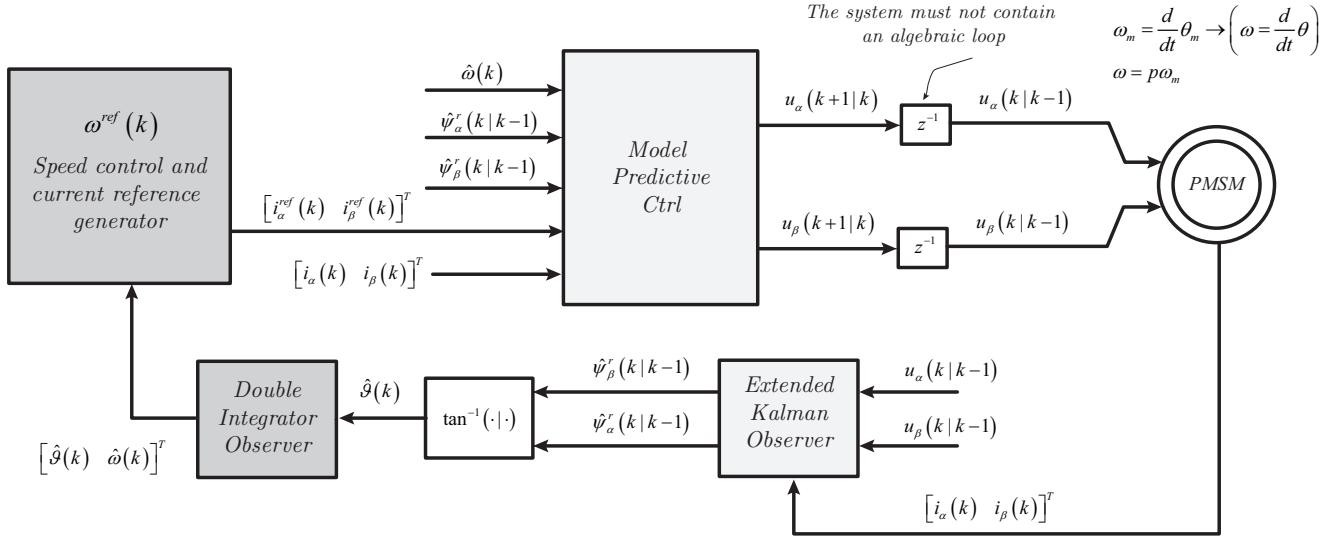


Figure 9.35: PMSM control architecture.

In order to implement the selected control architecture the following mathematical substructures shall be investigated

- A discrete time representation of the mathematical model of the PMSM.
- The concept of the space vector and its correlation with an IGBT-based Inverter.
- A trajectory tracking cost function for the MTP. MPC is an optimal control and needs a cost function to be used to derive the optimal inverter output voltage.
- Torque to current reference generator (the mechanical side *works* with torque but the electrical side *works* with currents).
- A discrete time representation of the fluxes observer
- A discrete time representation of the double integrator observer, see also Example 1.11.

### 9.8.2.1 Model discretization

As already mentioned, in order to implement our control structure a discrete-time representation of the system of Eq. (9.8.2) shall be derived. By the use of the Euler approximation for the derivatives using the sampling time  $t_s$ , that is

$$\frac{d}{dt}x(t) \approx \frac{x[(k+1)t_s] - x(kt_s)}{t_s} = \frac{x(k+1) - x(k)}{t_s} \quad (9.8.3)$$

we obtain the following discrete time representation of the PMSM model

**Discrete-time representation of the PMSM**

$$\left\{ \begin{array}{l} i_\alpha(k+1) = \left(1 - t_s \frac{R_s}{L_\alpha}\right) i_\alpha(k) + t_s \frac{\omega(k)}{L_\alpha} \psi_\beta(k) + \frac{t_s}{L_\alpha} u_\alpha(k) \\ i_\beta(k+1) = \left(1 - t_s \frac{R_s}{L_\beta}\right) i_\beta(k) - t_s \frac{\omega(k)}{L_\beta} \psi_\alpha(k) + \frac{t_s}{L_\beta} u_\beta(k) \\ \psi_\alpha^r(k+1) = \psi_\alpha^r(k) - t_s \omega(k) \psi_\beta^r(k) \\ \psi_\beta^r(k+1) = \psi_\beta^r(k) + t_s \omega(k) \psi_\alpha^r(k) \\ \omega_m(k+1) = \omega_m(k) + \frac{t_s}{J} \left[ \frac{3}{2} p \left[ \psi_\alpha^r(k) i_\beta(k) - \psi_\beta^r(k) i_\alpha(k) \right] - \tau_l(k) - b \omega_m(k) \right] \\ \theta(k+1) = \theta(k) + t_s \omega(k) = \theta(k) + t_s p \omega_m(k) \end{array} \right. \quad (9.8.4)$$

where  $p$  is the number of poles pairs,  $b$  the bearing viscosity and  $\tau_l(t)$  the external and unknown load (disturbance).

The model of Eq. (9.8.4) can be used to implement the MPC as well as the EKF, anyway to implement the EKF we proceed in a little bit different way: we first propose a linearization of Eq. (9.8.2) along the estimated state vector  $\hat{\vec{x}}$  (where we obtain a linear model) and after that we discretize by Taylor as already seen. This procedure will be described in section 9.8.3

### 9.8.3 Extended Kalman Filter (EKF) Observer

In this section an application of the Extended Kalman filter is presented, where a PMSM rotor fluxes estimator is proposed and implemented. We will proceed using the  $\alpha\beta$  model representation reported in Eq. (9.8.2).

A different, but equivalent, representation of the PMSM can be re-written as follows

$$\left\{ \begin{array}{l} \frac{d}{dt} \vec{x}(t) = \vec{g}(\vec{x}) + \mathbf{B} \vec{u}(t) \\ \vec{g}(t) = \mathbf{C} \vec{x}(t) \end{array} \right. \quad (9.8.5)$$

where

$$\vec{x} = \begin{bmatrix} i_\alpha \\ i_\beta \\ \psi_\alpha^r \\ \psi_\beta^r \\ \vartheta \\ \omega \end{bmatrix}, \quad \vec{u} = \begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} \quad (9.8.6)$$

and where

$$\mathbf{B} = \frac{1}{L_s} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.8.7)$$

and where

$$\frac{d}{dt} \vec{x}(t) = \vec{g}(\vec{x}) \Rightarrow \begin{cases} \frac{d}{dt} i_\alpha(t) = -\frac{R_s}{L_s} i_\alpha(t) + \frac{1}{L_s} \omega(t) \psi_\beta^r(t) + \frac{1}{L_s} u_\alpha(t) \\ \frac{d}{dt} i_\beta(t) = -\frac{R_s}{L_s} i_\beta(t) - \frac{1}{L_s} \omega(t) \psi_\alpha^r(t) + \frac{1}{L_s} u_\beta(t) \\ \frac{d}{dt} \psi_\alpha^r(t) = -\omega(t) \psi_\beta^r(t) \\ \frac{d}{dt} \psi_\beta^r(t) = \omega(t) \psi_\alpha^r(t) \\ \frac{d}{dt} \vartheta(t) = \omega(t) \\ \frac{d}{dt} \omega(t) = \frac{1}{J} \left[ \frac{3}{2} (\psi_\alpha^r(t) i_\beta(t) - \psi_\beta^r(t) i_\alpha(t)) - \frac{1}{p} \tau_l(t) - \frac{1}{p} b \omega_m(t) \right] \end{cases} \quad (9.8.8)$$

we are considering  $L_\alpha = L_\beta = L_s$ .

**For the purpose of the EKF implementation a reduced dynamic description is taken into account**, in particular it is assumed the dynamic of the mechanical system is more than one order slower than the electrical system and for this assumption we decide to represent the system of Eq. (9.8.8) as follows

$$\frac{d}{dt} \vec{x}(t) = \vec{f}(\vec{x}) \Rightarrow \begin{cases} \frac{d}{dt} i_\alpha(t) = -\frac{R_s}{L_s} i_\alpha(t) + \frac{1}{L_s} \omega(t) \psi_\beta^r(t) + \frac{1}{L_s} u_\alpha(t) \\ \frac{d}{dt} i_\beta(t) = -\frac{R_s}{L_s} i_\beta(t) - \frac{1}{L_s} \omega(t) \psi_\alpha^r(t) + \frac{1}{L_s} u_\beta(t) \\ \frac{d}{dt} \psi_\alpha^r(t) = -\omega(t) \psi_\beta^r(t) \\ \frac{d}{dt} \psi_\beta^r(t) = \omega(t) \psi_\alpha^r(t) \\ \frac{d}{dt} \vartheta(t) = \omega(t) \\ \frac{d}{dt} \omega(t) = 0 \end{cases} \quad (9.8.9)$$

where the mechanical dynamic has been **intentionally** approximated to a stationary mo-

tion.

In order to implement the EKF we now linearize the system of Eq. (9.8.9) along the estimated state space trajectory  $\hat{\vec{x}}$ , as follows

$$\frac{\partial}{\partial \vec{x}} \vec{f}(\vec{x}) \Big|_{\vec{x}=\hat{\vec{x}}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\hat{\vec{x}}(t)} = \tilde{\mathbf{A}}_1 + \hat{\omega}(t)\tilde{\mathbf{A}}_2 + \hat{\psi}_\alpha^r(t)\tilde{\mathbf{A}}_3 + \hat{\psi}_\beta^r(t)\tilde{\mathbf{A}}_4 = \tilde{\mathbf{A}}(t) \quad (9.8.10)$$

where  $\hat{\vec{x}}$  is the estimated state vector from the EKF.

The matrices  $\tilde{\mathbf{A}}_1$ ,  $\tilde{\mathbf{A}}_2$ ,  $\tilde{\mathbf{A}}_3$  and  $\tilde{\mathbf{A}}_4$  are as follows

$$\tilde{\mathbf{A}}_1 = \begin{bmatrix} -\frac{R_s}{L_s} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{R_s}{L_s} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \tilde{\mathbf{A}}_2 = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{L_s} & 0 & 0 \\ 0 & 0 & -\frac{1}{L_s} & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{\mathbf{A}}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{L_s} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \tilde{\mathbf{A}}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \frac{1}{L_s} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.8.11)$$

The discretization of the transition matrix  $\tilde{\mathbf{A}}(t)$  can be approximated by Taylor as follows

$$\mathbf{A}(k|k-1) = \mathbf{I} + \tilde{\mathbf{A}}_1 t_s + \hat{\omega}(k|k-1)\tilde{\mathbf{A}}_2 t_s + \hat{\psi}_\alpha^r(k|k-1)\tilde{\mathbf{A}}_3 t_s + \hat{\psi}_\beta^r(k|k-1)\tilde{\mathbf{A}}_4 t_s \quad (9.8.12)$$

The EKF set of equations can be written as follows

### The Extended Kalman Filter Algorithm

$$\hat{\vec{x}}(k|k) = \mathbf{A}(k|k-1)\hat{\vec{x}}(k|k-1) + \mathbf{B}\vec{u}(k|k-1) \quad (9.8.13)$$

$$\mathbf{P}(k|k) = \mathbf{A}(k|k-1)\mathbf{P}(k|k-1)\mathbf{A}(k|k-1)^T + \mathbf{Q} \quad (9.8.14)$$

$$\mathbf{L}(k) = \mathbf{P}(k|k)\mathbf{C}^T \left( \mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R} \right)^{-1} \quad (9.8.15)$$

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \left( \vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k) \right) \quad (9.8.16)$$

$$\mathbf{P}(k+1|k) = (\mathbf{I} - \mathbf{L}(k)\mathbf{C})\mathbf{P}(k|k) \quad (9.8.17)$$

where

$$\mathbf{A}(k|k-1) = \mathbf{I} + \tilde{\mathbf{A}}_1 t_s + \hat{\omega}(k|k-1)\tilde{\mathbf{A}}_2 t_s + \hat{\psi}_\alpha^r(k|k-1)\tilde{\mathbf{A}}_3 t_s + \hat{\psi}_\beta^r(k|k-1)\tilde{\mathbf{A}}_4 t_s \quad (9.8.18)$$

and where

$$\mathbf{B} = \begin{bmatrix} \frac{1}{L_s} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{L_s} & 0 & 0 & 0 & 0 \end{bmatrix}^T \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.8.19)$$

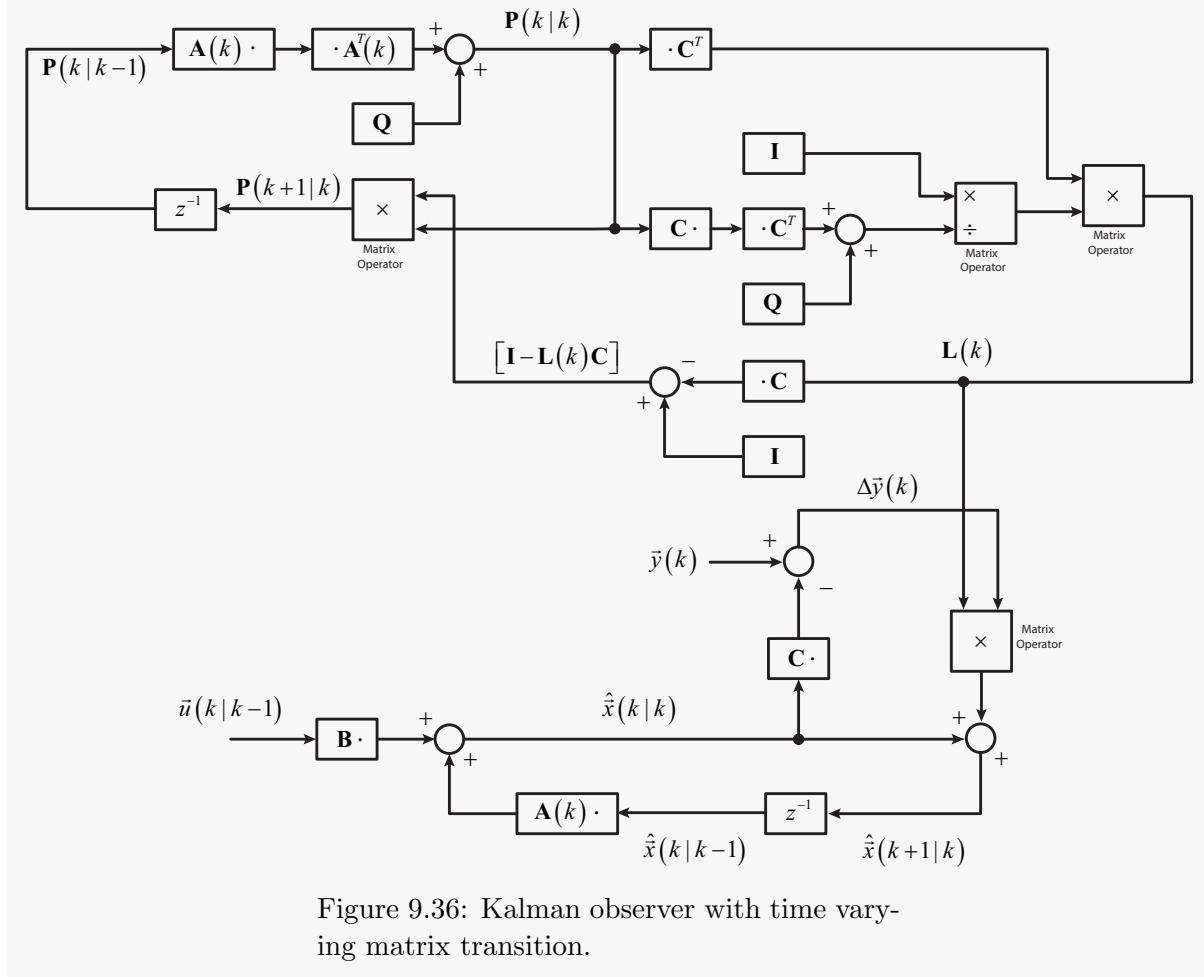


Figure 9.36: Kalman observer with time varying matrix transition.

#### 9.8.4 EKF Observer for nonlinear PMSM

In addition to the previous EKF observer, we consider now the case where the torque expression is taken into account into the linearization along the observed state trajectory.

$$\begin{cases}
 \frac{d}{dt} i_\alpha(t) = -\frac{R_s}{L_s} i_\alpha(t) + \frac{1}{L_s} \omega(t) \psi_\beta^r(t) + \frac{1}{L_s} u_\alpha(t) \\
 \frac{d}{dt} i_\beta(t) = -\frac{R_s}{L_s} i_\beta(t) - \frac{1}{L_s} \omega(t) \psi_\alpha^r(t) + \frac{1}{L_s} u_\beta(t) \\
 \frac{d}{dt} \psi_\alpha^r(t) = -\omega(t) \psi_\beta^r(t) \\
 \frac{d}{dt} \psi_\beta^r(t) = \omega(t) \psi_\alpha^r(t) \\
 \frac{d}{dt} \vartheta(t) = \omega(t) \\
 \frac{d}{dt} \omega(t) = \frac{p}{J} \left\{ \frac{3}{2} p \left[ \psi_\alpha^r(t) i_\beta(t) - \psi_\beta^r(t) i_\alpha(t) \right] - \tau_l(t) - \frac{b}{p} \omega(t) \right\}
 \end{cases} \quad (9.8.20)$$

where we assumed  $L_\alpha = L_\beta = L_s$  and  $\frac{1}{p}\omega(t) = \omega_m(t)$ .

$$\begin{aligned}
 \frac{\partial}{\partial \vec{x}} \vec{g}(\vec{x}) \Big|_{\vec{x}=\hat{\vec{x}}} &= \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \dots & \frac{\partial g_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_n}{\partial x_1} & \dots & \frac{\partial g_n}{\partial x_n} \end{bmatrix}_{\hat{\vec{x}}(t)} \\
 &= \begin{bmatrix} -\frac{R_s}{L_s} & 0 & 0 & \frac{\hat{\omega}}{L_s} & 0 & \frac{\hat{\psi}_\beta}{L_s} \\ 0 & -\frac{R_s}{L_s} & -\frac{\hat{\omega}}{L_s} & 0 & 0 & -\frac{\hat{\psi}_\alpha}{L_s} \\ 0 & 0 & 0 & -\hat{\omega} & 0 & -\hat{\psi}_\beta \\ 0 & 0 & \hat{\omega} & 0 & 0 & \hat{\psi}_\alpha \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{3}{2} \frac{p^2}{J} \hat{\psi}_\beta & \frac{3}{2} \frac{p^2}{J} \hat{\psi}_\alpha & \frac{3}{2} \frac{p^2}{J} \hat{i}_\beta & -\frac{3}{2} \frac{p^2}{J} \hat{i}_\alpha & 0 & -\frac{b}{J} \end{bmatrix} \quad (9.8.21) \\
 &= \tilde{\mathbf{A}}_1 + \hat{\omega}(t) \tilde{\mathbf{A}}_2 + \hat{\psi}_\alpha^r(t) \tilde{\mathbf{A}}_3 + \hat{\psi}_\beta^r(t) \tilde{\mathbf{A}}_4 + \hat{i}_\alpha(t) \tilde{\mathbf{A}}_5 + \hat{i}_\beta(t) \tilde{\mathbf{A}}_6
 \end{aligned}$$

where  $\hat{\vec{x}}$  is the estimated state vector from the EKF.

The matrices  $\tilde{\mathbf{A}}_1$ ,  $\tilde{\mathbf{A}}_2$ ,  $\tilde{\mathbf{A}}_3$ ,  $\tilde{\mathbf{A}}_4$ ,  $\tilde{\mathbf{A}}_5$  and  $\tilde{\mathbf{A}}_6$  are as follows

$$\tilde{\mathbf{A}}_1 = \begin{bmatrix} -\frac{R_s}{L_s} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{R_s}{L_s} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\frac{b}{J} \end{bmatrix}, \quad \tilde{\mathbf{A}}_2 = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{L_s} & 0 & 0 \\ 0 & 0 & -\frac{1}{L_s} & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.8.22)$$

$$\tilde{\mathbf{A}}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{L_s} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{3}{2} \frac{p^2}{J} & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \tilde{\mathbf{A}}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \frac{1}{L_s} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{3}{2} \frac{p^2}{J} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9.8.23)$$

$$\tilde{\mathbf{A}}_5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{3}{2} \frac{p^2}{J} & 0 & 0 \end{bmatrix}, \quad \tilde{\mathbf{A}}_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{2} \frac{p^2}{J} & 0 & 0 & 0 \end{bmatrix} \quad (9.8.24)$$

The discretization of the transition matrix  $\tilde{\mathbf{A}}(t)$  can be approximated by Taylor as follows

$$\begin{aligned} \mathbf{A}(k|k-1) = & \mathbf{I}_6 + \tilde{\mathbf{A}}_1 t_s + \hat{\omega}(k|k-1) \tilde{\mathbf{A}}_2 t_s + \\ & + \hat{\psi}_\alpha^r(k|k-1) \tilde{\mathbf{A}}_3 t_s + \hat{\psi}_\beta^r(k|k-1) \tilde{\mathbf{A}}_4 t_s + \\ & + \hat{i}_\alpha(k|k-1) \tilde{\mathbf{A}}_5 t_s + \hat{i}_\beta(k|k-1) \tilde{\mathbf{A}}_6 t_s. \end{aligned} \quad (9.8.25)$$

### 9.8.5 Space Vector (SV) and Model Predictive (MP) Modulators

In this section a brief overview of the space vector modulation (*SVM*) is reported. Figure 9.37 depicts the voltages which a three phase inverter can actuate as function of the switches state in  $\alpha\beta$  fixed reference frame representation.

Figure 9.38 shows an example of a generic voltage actuation using a combination of adjacent state voltage vectors. The space vector hexagon can be divided into six sectors. For a given output voltage, the corresponding sector have to be identified. Once the sector is identified the “realization” of the output vector is made by a time combination of the adjacent three space vector voltages ( $\vec{v}_n$ ,  $\vec{v}_{n+1}$ , and  $\vec{v}_0$ ).

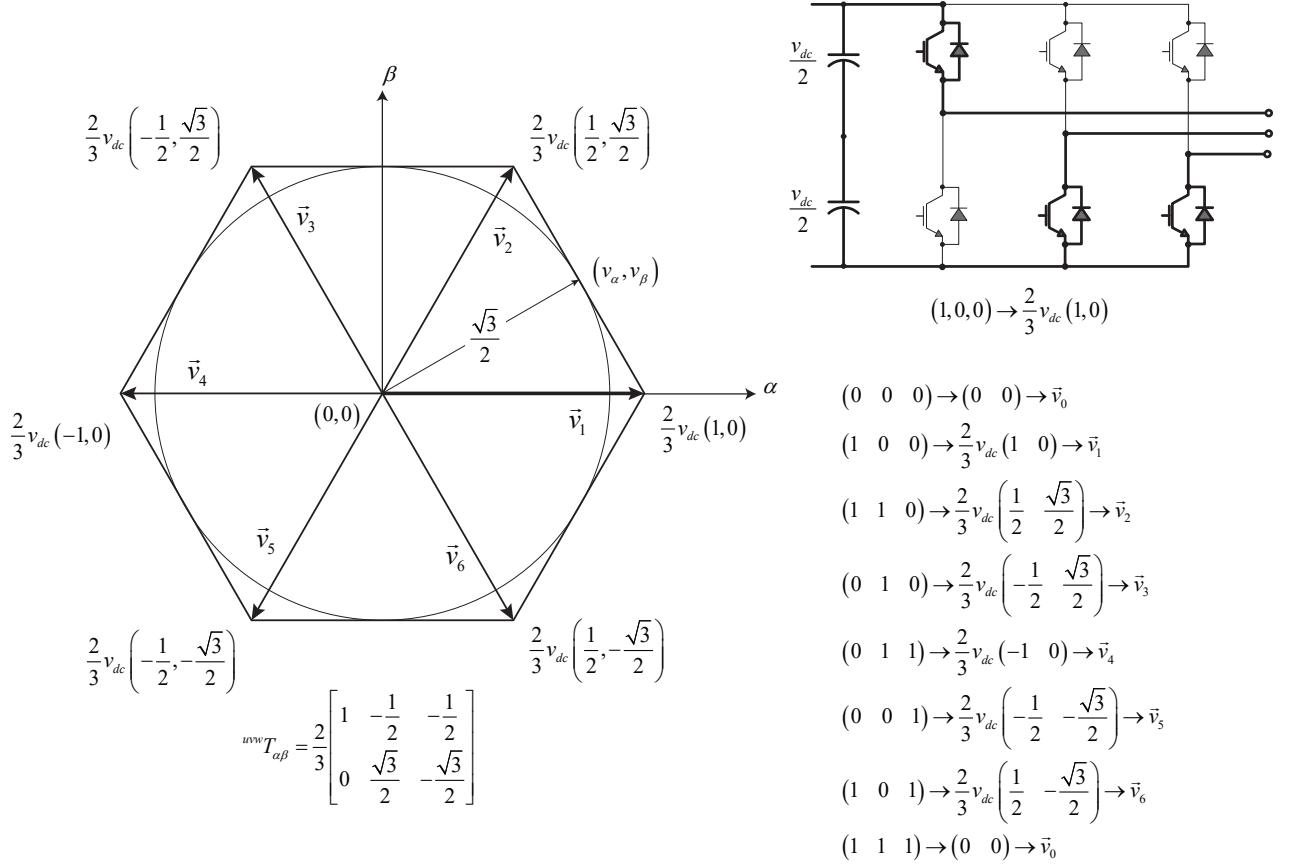


Figure 9.37: Representation of the space vector modulation and relation with inverter switches state.

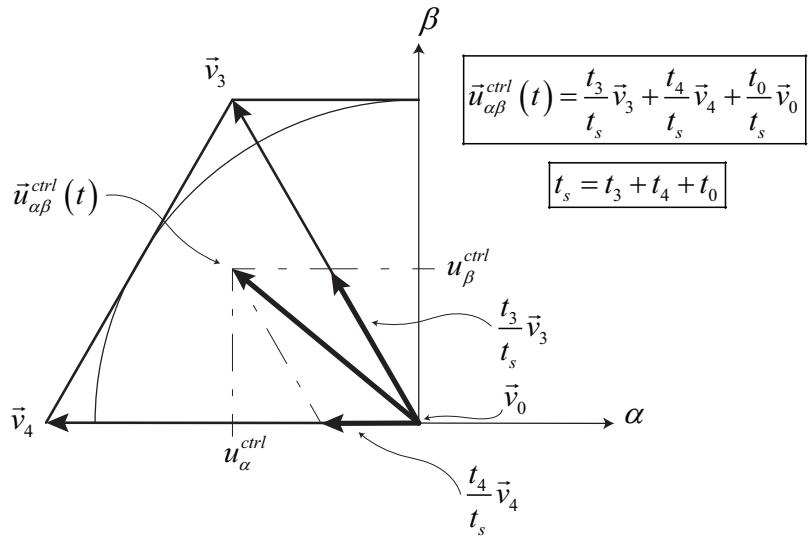


Figure 9.38: Space vector representation of a generic voltage vector  $\vec{u}_{\alpha\beta}^{ctrl}$ .

Define  $t_s$  the switching period of the SVM, any vector  $\vec{v}_{\alpha\beta}$  (in some case the notation  $\vec{v}_{\xi\eta}$  is used) located in the sector  $n$  of the space vector hexagon can be “realized” by the time

combination of the adjacent vectors ( $\vec{v}_n$ ,  $\vec{v}_{n+1}$ , and  $\vec{v}_0$ ), as follows

$$\vec{v}_{\alpha\beta} = \frac{t_n}{t_s}\vec{v}_n + \frac{t_{n+1}}{t_s}\vec{v}_{n+1} + \frac{t_0}{t_s}\vec{v}_0 \quad (9.8.26)$$

where  $t_n + t_{n+1} + t_0 = t_s$ .

In Figure 9.39–Figure 9.42 the relation between the state of the inverter switching bridge and the space voltage vector representation in  $\alpha\beta$  is depicted.

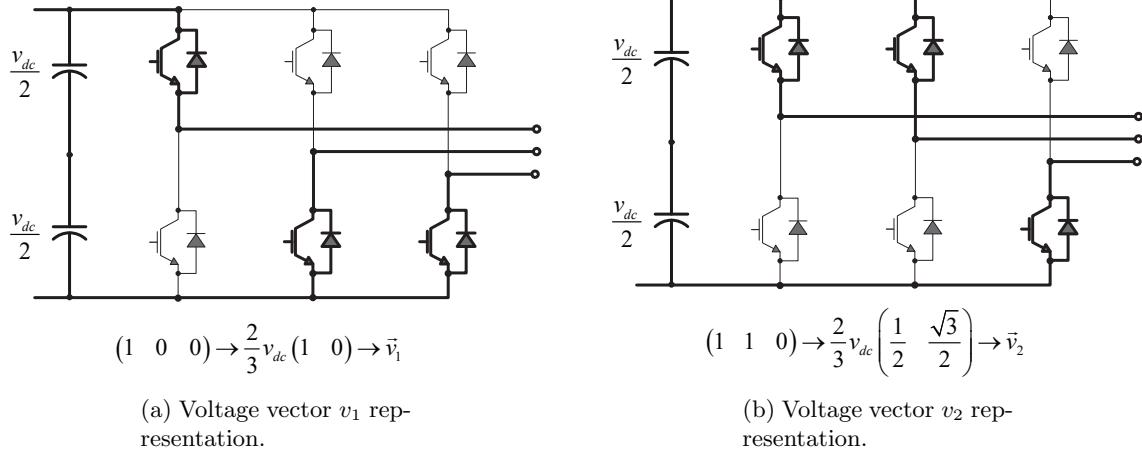


Figure 9.39: Inverter state representation of the voltages  $v_1$  and  $v_2$ .

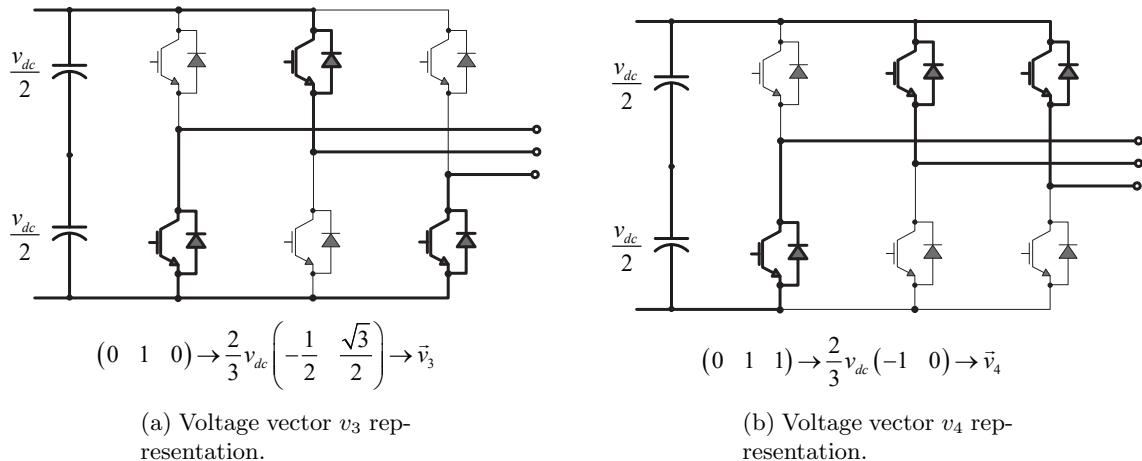


Figure 9.40: Inverter state representation of the voltages  $v_3$  and  $v_4$ .

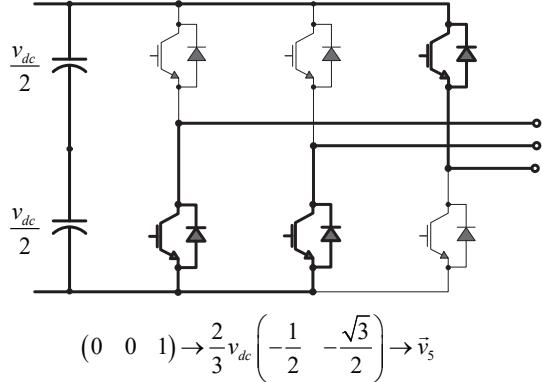
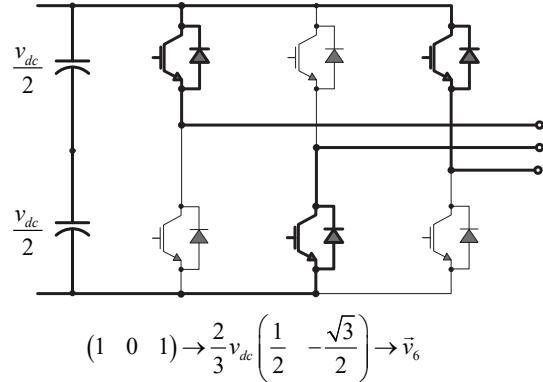

 (a) Voltage vector  $v_5$  representation.

 (b) Voltage vector  $v_6$  representation.

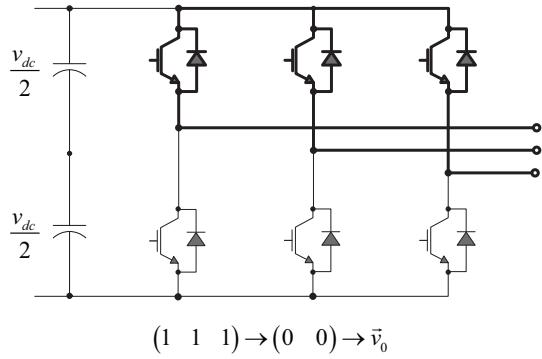
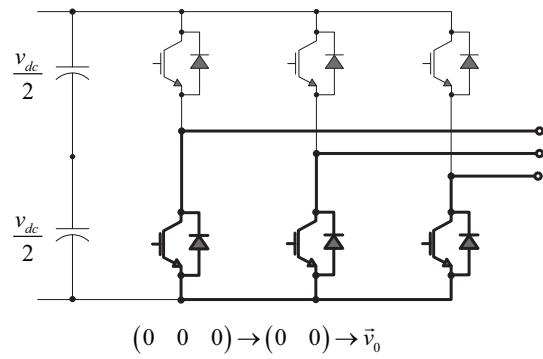
 Figure 9.41: Inverter state representation of the voltages  $v_5$  and  $v_6$ .

 (a) Voltage vector  $v_0$  representation - case 1.

 (b) Voltage vector  $v_0$  representation - case 2.

 Figure 9.42: Inverter state representations of the voltage  $v_0$ .

Concerning the MP-control implementation it is necessary to identify all possible  $(u_\alpha(k), u_\beta(k))$  output voltage and also define the constraint. For a given couple of components  $(u_\alpha(k-1), u_\beta(k-1))$  exist a finite possible set  $\mathcal{U}(k)$  of  $(u_\alpha(k), u_\beta(k))$  components as shown in the Table 9.3

$(u_\alpha(k-1), u_\beta(k-1))$	$\mathcal{U}(k)$
$(0, 0)$	$\frac{2}{3} \left\{ (0, 0); (1, 0); \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right); \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right); (-1, 0); \left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right); \left(\frac{1}{2}, -\frac{\sqrt{3}}{2}\right) \right\}$
$\frac{2}{3}(1, 0)$	$\frac{2}{3} \left\{ (0, 0); (1, 0); \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right); \left(\frac{1}{2}, -\frac{\sqrt{3}}{2}\right) \right\}$
$\frac{2}{3}\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$	$\frac{2}{3} \left\{ (0, 0); \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right); (1, 0); \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \right\}$
$\frac{2}{3}\left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$	$\frac{2}{3} \left\{ (0, 0); \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right); (-1, 0); \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \right\}$
$\frac{2}{3}(-1, 0)$	$\frac{2}{3} \left\{ (0, 0); (-1, 0); \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right); \left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right) \right\}$
$\frac{2}{3}\left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right)$	$\frac{2}{3} \left\{ (0, 0); \left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right); (-1, 0); \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \right\}$
$\frac{2}{3}\left(\frac{1}{2}, -\frac{\sqrt{3}}{2}\right)$	$\frac{2}{3} \left\{ (0, 0); \left(\frac{1}{2}, -\frac{\sqrt{3}}{2}\right); (1, 0); \left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right) \right\}$

Table 9.3: Set of admissible output voltage  $\mathcal{U}(k)$  from a given voltage vector  $(u_\alpha \ u_\beta)$ .

The knowledge of the space vector commutation law and in particular the knowledge of the set  $\mathcal{U}(k)$  because it will affect the implementation of the MPC.

### 9.8.6 Current Reference Generator

In this subsection we derive the relation between the electromagnetic torque which per PMSM has to provide and the stator current. Basically we will derive the set of current reference

$$\tau_m^{ref} \rightarrow (i_\alpha^{ref}, i_\beta^{ref}) \quad (9.8.27)$$

where  $\tau_m^{ref}$  is the output of the speed control.

As shown in a previous section the electromagnetic torque can be expressed as follows

$$\tau_m = \frac{3}{2}p (\psi_\alpha^r i_\beta - \psi_\beta^r i_\alpha) \quad (9.8.28)$$

where  $\vec{\psi}_{\alpha\beta}^r = [\Psi^M \cos(\vartheta) \ \Psi^M \sin(\vartheta)]^T$  are permanent magnet rotor fluxes linkage to the stator, and  $p$  the number of pole pairs.

Let  $\hat{\vartheta}$  be the estimation of the magnetic flux phase resulted from the Kalman observer

$$\hat{\vartheta} = \arctan(\hat{\psi}_\beta^r, \hat{\psi}_\alpha^r) \quad (9.8.29)$$

That means to achieve the *maximum torque per ampere* we have to build a current reference as follows

$$\begin{aligned} i_\alpha^{ref} &= -i^{ref} \sin(\hat{\theta}) \\ i_\beta^{ref} &= i^{ref} \cos(\hat{\theta}) \end{aligned} \quad (9.8.30)$$

when  $\vartheta - \hat{\vartheta} \rightarrow 0$  the applied torque results in

$$\begin{aligned}
 \tau_m &= \frac{3}{2}p (\psi_\alpha^r i_\beta - \psi_\beta^r i_\alpha) \\
 &= \begin{bmatrix} i_\alpha & i_\beta \end{bmatrix} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \psi_\alpha^r \\ \psi_\beta^r \end{bmatrix} \\
 &= \begin{bmatrix} -i^{ref} \sin(\hat{\theta}) & i^{ref} \cos(\hat{\theta}) \end{bmatrix} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Psi^M \cos(\vartheta) \\ \Psi^M \sin(\vartheta) \end{bmatrix} \\
 &= \frac{3}{2}p \Psi^M i^{ref} \quad \Leftarrow \vartheta - \hat{\vartheta} \rightarrow 0
 \end{aligned} \tag{9.8.31}$$

Eq. (9.8.31) requires that  $\vartheta - \hat{\vartheta} \rightarrow 0$  that means the EKF performance plays a fundamental role in the capability of the overall system to deliver the required amount of electromagnetic torque.

### 9.8.7 MPC with EKF Implementation

Up to now we have defined the following substructure

- Fluxes observer based on EKF
- Voltage space vector elements available using the two level Inverter.

what is missing are

- PI-based speed control, seen in example 1.6
- Double Integrator Observer, seen in example 1.11
- Model predictive control.

Let's now going to derive the MPC for the isotropic PMSM.

From the discrete time model of the PMSM described by Eq. (9.8.4) we take the first two Kirchhoff's voltage law equations that describe the dynamic of the stator current.

These two equation will be used to implement our current control based on model predictive. As from hypothesis we assume  $L_\alpha = L_\beta = L_s$  which results in

$$\begin{cases} i_\alpha(k+1) = \left(1 - t_s \frac{R_s}{L_s}\right) i_\alpha(k) + t_s \frac{\hat{\omega}(k)}{L_s} \hat{\psi}_\beta(k) + t_s \frac{1}{L_s} u_\alpha(k) \\ i_\beta(k+1) = \left(1 - t_s \frac{R_s}{L_s}\right) i_\beta(k) - t_s \frac{\hat{\omega}(k)}{L_s} \hat{\psi}_\alpha(k) + t_s \frac{1}{L_s} u_\beta(k) \end{cases} \tag{9.8.32}$$

**These two equations allow predictions of the stator currents to be calculated for each one of the seven voltage vectors generated by the inverter.**

In order to implement a tracking control problem we can assume the following cost function:

$$g = \left[ i_{\alpha}^{ref}(k) - i_{\alpha}^P(k+1) \right]^2 + \left[ i_{\beta}^{ref}(k) - i_{\beta}^P(k+1) \right]^2 \quad (9.8.33)$$

The MPC algorithm can be summarized as follows

- At every step the control determine in which output voltage sector is currently located.
- By the knowledge of the voltage sector the algorithm calculate all current prediction using all possible output voltage which can be actuated from the given voltage sector.
- By the knowledge of all current prediction we calculate the value of the cost function for each of all current prediction.
- By the knowledge of the cost function value for each current prediction the control determine the value of output voltage for which the cost function is minimal.
- The set of output voltage which minimize the cost function will be actuated at the next step.

#### 9.8.7.1 MPC algorithm implementation

A possible implementation of the MPC is here reported in the below code:

```

function[ualpha, ubeta] = mpc(vdc,Rs,Ls,ts,ialpha_ref, ...
    ibeta_ref,ialpha,ibeta,omega_hat,psi_alpha_hat,psi_beta_hat)

persistent ualpha_z ubeta_z

if isempty(ualpha_z)
    ualpha_z = 0;
    ubeta_z = 0;
end

ia0 = (1-Rs/Ls*ts)*ialpha + omega_hat/Ls*psi_beta_hat*ts;
ib0 = (1-Rs/Ls*ts)*ibeta - omega_hat/Ls*psi_alpha_hat*ts;

ia1 = ia0 + 2/3*vdc*ts/Ls;
ib1 = ib0;

ia2 = ia0+2/3*1/2*vdc*ts/Ls;
ib2 = ib0+2/3*sqrt(3)/2*vdc*ts/Ls;

ia3 = ia0-2/3*1/2*vdc*ts/Ls;
ib3 = ib2;

ia4 = ia0-2/3*vdc*ts/Ls;
ib4 = ib0;

```

```
ia5 = ia0-2/3*1/2*vdc*ts/Ls;
ib5 = ib0-2/3*sqrt(3)/2*vdc*ts/Ls;

ia6 = ia2;
ib6 = ib0-2/3*sqrt(3)/2*vdc*ts/Ls;

J0 = (ialpha_ref - ia0)^2 + (ibeta_ref - ib0)^2;
J1 = (ialpha_ref - ia1)^2 + (ibeta_ref - ib1)^2;
J2 = (ialpha_ref - ia2)^2 + (ibeta_ref - ib2)^2;
J3 = (ialpha_ref - ia3)^2 + (ibeta_ref - ib3)^2;
J4 = (ialpha_ref - ia4)^2 + (ibeta_ref - ib4)^2;
J5 = (ialpha_ref - ia5)^2 + (ibeta_ref - ib5)^2;
J6 = (ialpha_ref - ia6)^2 + (ibeta_ref - ib6)^2;

if (ualpha_z == 0) && (ubeta_z == 0)
    [m,i] = min([J0, J1, J2, J3, J4, J5, J6]);
    if i == 1
        ualpha = 0;
        ubeta = 0;
    elseif i == 2
        ualpha = 2/3*vdc;
        ubeta = 0;
    elseif i == 3
        ualpha = 2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    elseif i == 4
        ualpha = -2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    elseif i == 5
        ualpha = -2/3*vdc;
        ubeta = 0;
    elseif i == 6
        ualpha = -2/3*1/2*vdc;
        ubeta = -2/3*sqrt(3)/2*vdc;
    else
        ualpha = 2/3*1/2*vdc;
        ubeta = -2/3*sqrt(3)/2*vdc;
    end
elseif (ualpha_z == 2/3*vdc) && (ubeta_z == 0)
    [m,i] = min([J0, J1, J2, J6]);
    if i == 1
        ualpha = 0;
        ubeta = 0;
    elseif i == 2
        ualpha = 2/3*vdc;
        ubeta = 0;
    elseif i == 3
        ualpha = 2/3*1/2*vdc;
```

```
ubeta = 2/3*sqrt(3)/2*vdc;
else
    ualpha = 2/3*1/2*vdc;
    ubeta = -2/3*sqrt(3)/2*vdc;
end
elseif (ualpha_z == 2/3*1/2*vdc) && (ubeta_z == 2/3*sqrt(3)/2*vdc)
    [m,i] = min([J0, J1, J2, J3]);
    if i == 1
        ualpha = 0;
        ubeta = 0;
    elseif i == 2
        ualpha = 2/3*vdc;
        ubeta = 0;
    elseif i == 3
        ualpha = 2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    else
        ualpha = -2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    end
elseif (ualpha_z == -2/3*1/2*vdc) && (ubeta_z == 2/3*sqrt(3)/2*vdc)
    [m,i] = min([J0, J2, J3, J4]);
    if i == 1
        ualpha = 0;
        ubeta = 0;
    elseif i == 2
        ualpha = 2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    elseif i == 3
        ualpha = -2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    else
        ualpha = -2/3*vdc;
        ubeta = 0;
    end
elseif (ualpha_z == -2/3*vdc) && (ubeta_z == 0)
    [m,i] = min([J0, J3, J4, J5]);
    if i == 1
        ualpha = 0;
        ubeta = 0;
    elseif i == 2
        ualpha = -2/3*1/2*vdc;
        ubeta = 2/3*sqrt(3)/2*vdc;
    elseif i == 3
        ualpha = -2/3*vdc;
        ubeta = 0;
    else
        ualpha = -2/3*1/2*vdc;
        ubeta = -2/3*sqrt(3)/2*vdc;
```

```

        end
    elseif (ualpha_z == -2/3*1/2*vdc) && (ubeta_z == -2/3*sqrt(3)/2*vdc)
        [m,i] = min([J0, J4, J5, J6]);
        if i == 1
            ualpha = 0;
            ubeta = 0;
        elseif i == 2
            ualpha = -2/3*vdc;
            ubeta = 0;
        elseif i == 3
            ualpha = -2/3*1/2*vdc;
            ubeta = -2/3*sqrt(3)/2*vdc;
        else
            ualpha = 2/3*1/2*vdc;
            ubeta = -2/3*sqrt(3)/2*vdc;
        end
    else
        [m,i] = min([J0, J5, J6, J1]);
        if i == 1
            ualpha = 0;
            ubeta = 0;
        elseif i == 2
            ualpha = -2/3*1/2*vdc;
            ubeta = -2/3*sqrt(3)/2*vdc;
        elseif i == 3
            ualpha = 2/3*1/2*vdc;
            ubeta = -2/3*sqrt(3)/2*vdc;
        else
            ualpha = 2/3*vdc;
            ubeta = 0;
        end
    end
    ualpha_z = ualpha;
    ubeta_z = ubeta;
end

```

The control scheme for field oriented control (FOC) of the PMSM using predictive current control is shown in Figure 9.43. Here, a PI controller is used for the speed control which its output generates the current reference according to Eqs. 9.8.34.

$$\frac{d}{dt}\omega_m(t) = \frac{1}{J}\tau_m^{ref}(t) - \tau_l(t) - b\omega_m(t) \quad (9.8.34)$$

which in terms of current reference

$$\frac{d}{dt}\omega_m(t) = \frac{1}{J} \left[ \frac{3}{2}p \left[ \psi_\alpha^r(t)i_\beta^{ref}(t) - \psi_\beta^r(t)i_\alpha^{ref}(t) \right] - \tau_l(t) - b\omega_m(t) \right] \quad (9.8.35)$$

which results in

$$\begin{aligned} i_{\alpha}^{ref} &= -i^{ref} \sin(\hat{\theta}) \\ i_{\beta}^{ref} &= i^{ref} \cos(\hat{\theta}) \end{aligned} \quad (9.8.36)$$

that means exist and algebraic correlation between

$$\tau_m^{ref}(t) = \frac{3}{2} p \Psi^M i^{ref}(t) \quad (9.8.37)$$

where  $\tau_m^{ref}(t)$  is the output of the speed control block.

A model predictive controller is used for the current control block. In the predictive scheme, the discrete-time model of the machine is used for predicting the stator current components for each seven different voltage vectors generated by the inverter. The voltage vector that minimizes a given cost function is selected and applied during a whole sampling interval. The whole motor control scheme is shown in Figure 9.43

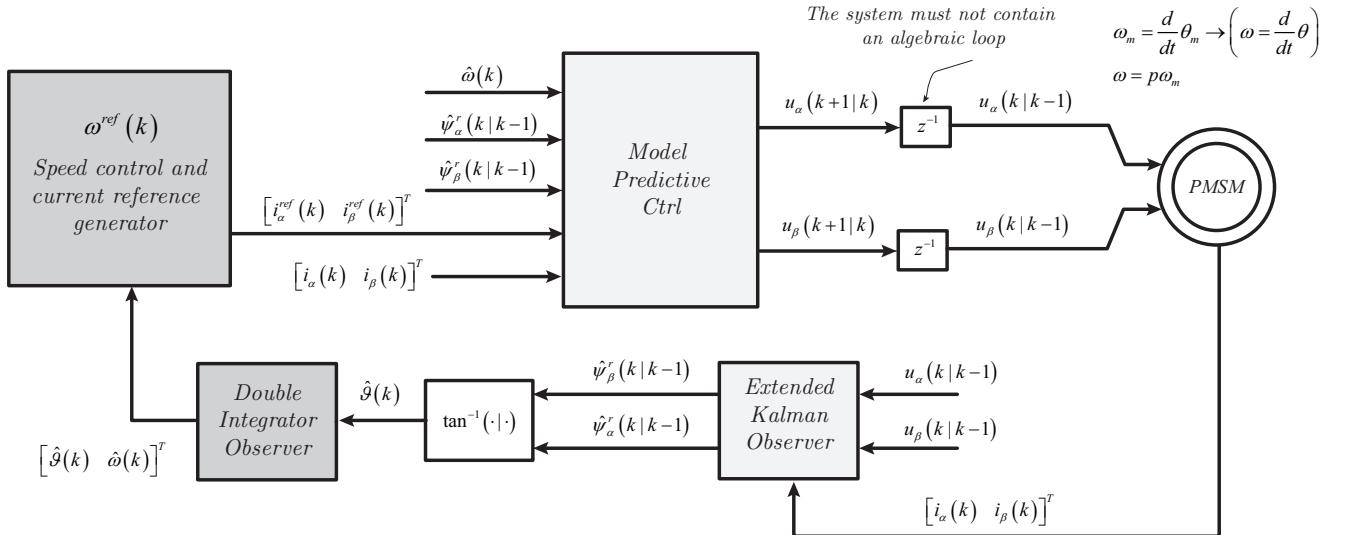


Figure 9.43: PMSM control architecture.

### 9.8.8 Simulation Results

Simulation result are here reported. At time ( $t = 15\text{s}$ ) the  $\psi^M$  permanent magnet flux is slowly reduced (as effect of the increasing temperature). An external load is applied as shown in Figure 9.45b. Simulation results for

$$\mathbf{Q} = \begin{bmatrix} \frac{R_s}{L_s} t_s & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{R_s}{L_s} t_s & 0 & 0 & 0 & 0 \\ 0 & 0 & 2t_s^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2t_s^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_s & 0 \\ 0 & 0 & 0 & 0 & 0 & t_s \end{bmatrix} \quad \mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (9.8.38)$$

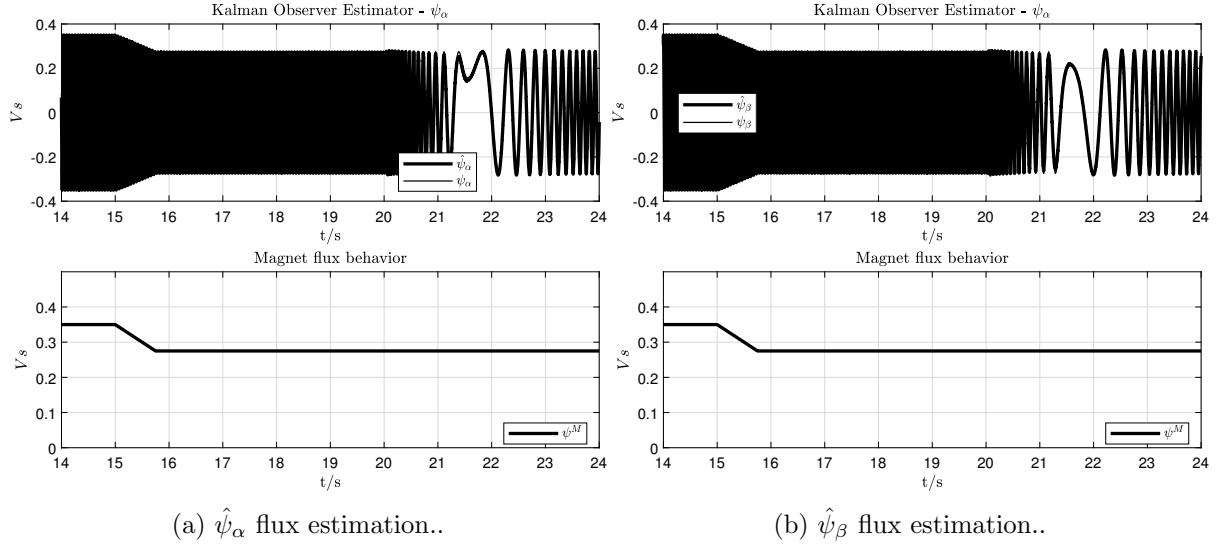


Figure 9.44: Simulation results.

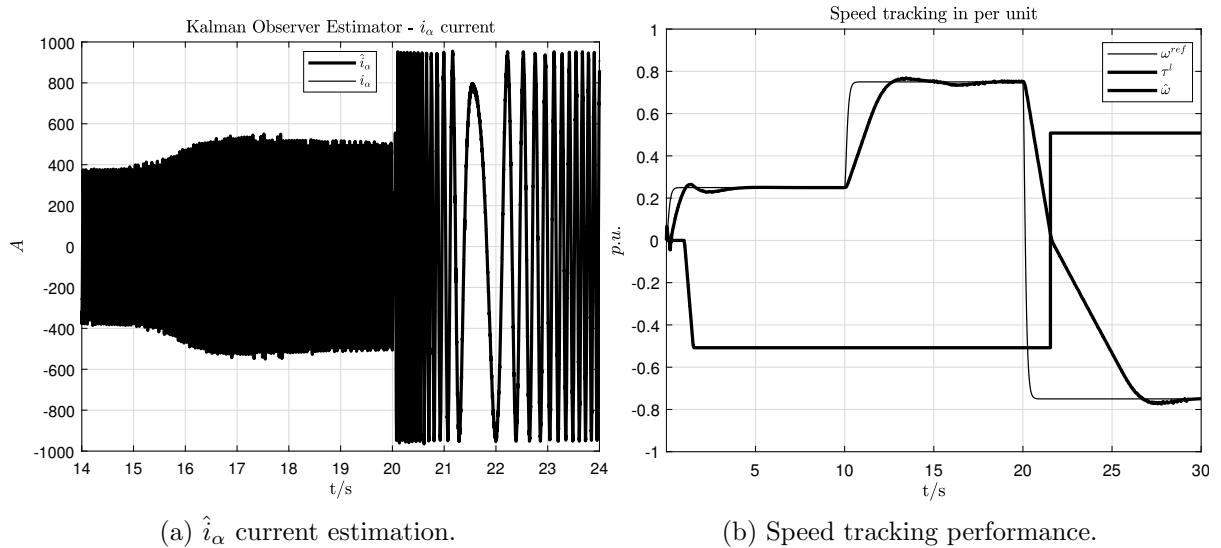


Figure 9.45: Simulation results.

## 9.9 Induction Motor Control with Kalman Filter Observer

In this section we present an application of the Kalman Filter used as state observer to an Induction Motor used as generator. The control problem can be dealigned as well

- We suppose the induction motor is driven by an external prime mover, like hydraulic turbine or wind turbine.
- The induction motor is rotating at constant speed.
- The aim of the control problem is to design a controller which regulate the electromagnetic torque according to a given torque reference.

- The only quantities available for the measure are the stator currents of the Induction Motor.

In order to implement a torque control of the IM the theory of the *Field Oriented Control* applied to the *Induction Motor* shall be taken into account. The knowledge of the rotor fluxes plays a fundamental role in the implementation of the *Field Oriented Control*. The reconstruction of the rotor fluxes components can be done by the used of the *Kalman Filter based Observer*.

The control architecture consist of

- *Kalman Filter based Observer* which by the measure of the stator current is able to reconstruct the rotor fluxes component.
- By the knowledge rotor fluxes component the electrical phase of the stator current can be calculated.
- A block which convert a given torque reference into a set of current reference. As PMSM the IM is a three phase system and requires two components to be completely determined.
- A reference frame transformation which converter the stator current expressed into the stationary reference frame ( $\alpha - \beta$ ) which are in general sin-wave, into a set of current components expressed respect the synchronous reference frame ( $x - y$ ) which are in general not-sin-wave quantities by the knowledge of the stator electrical phase.
- A block which perform a current control based on PI-control, expressed into the synchronous reference frame ( $x - y$ ).

Figure 9.46 shows the proposed control architecture.

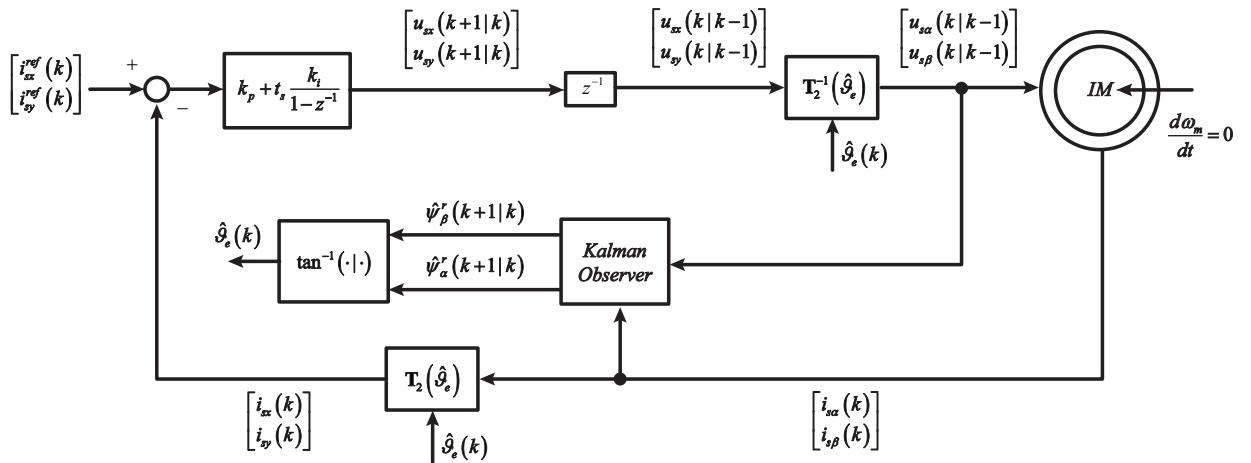


Figure 9.46: Induction motor torque control architecture with flux observer.

Before to start with the design of the control system a remind of the IM mathematical model is here reported.

### IM Dynamical Model Respect to a Stationary Reference Frame

$$\begin{aligned}
 \frac{di_{s\alpha}}{dt} &= -\gamma i_{s\alpha} + \beta \alpha \psi_{r\alpha} + \beta \omega_m \psi_{r\beta} + \frac{u_{s\alpha}}{\sigma} \\
 \frac{di_{s\beta}}{dt} &= -\gamma i_{s\beta} + \beta \alpha \psi_{r\beta} - \beta \omega_m \psi_{r\alpha} + \frac{u_{s\beta}}{\sigma} \\
 \frac{d\psi_{r\alpha}}{dt} &= -\alpha \psi_{r\alpha} - \omega_m \psi_{r\beta} + \alpha L_m i_{s\alpha} \\
 \frac{d\psi_{r\beta}}{dt} &= -\alpha \psi_{r\beta} + \omega_m \psi_{r\alpha} + \alpha L_m i_{s\beta} \\
 \frac{d\omega_m}{dt} &= \mu (\psi_{r\alpha} i_{s\beta} - \psi_{r\beta} i_{s\alpha}) - \frac{\tau_l}{J}
 \end{aligned} \tag{9.9.1}$$

where

$$\begin{aligned}
 \mu &= \frac{3}{2} \frac{L_m}{JL_r}, \quad \alpha = \frac{R_r}{L_r}, \quad \sigma = L_s \left( 1 - \frac{L_m^2}{L_s L_r} \right) \\
 \beta &= \frac{L_m}{\sigma L_r}, \quad \gamma = \frac{R_s}{\sigma} + \beta \alpha L_m
 \end{aligned} \tag{9.9.2}$$

Consider now the condition of constant speed

$$\frac{d\omega_m}{dt} = 0 \Rightarrow \omega_m = \omega_m^0 \tag{9.9.3}$$

which brings the above system from a nonlinear time-variant into a linear time-invariant as shown in the following

### IM Dynamical Model Respect to a Stationary Reference Frame and Constant Rotor Speed

$$\begin{aligned}
 \dot{\vec{x}} &= \tilde{\mathbf{A}} \vec{x} + \tilde{\mathbf{B}} \vec{u} \\
 \vec{y} &= \mathbf{C} \vec{x}
 \end{aligned} \tag{9.9.4}$$

where

$$\begin{aligned}\vec{x} &= \begin{bmatrix} i_{s\alpha} & i_{s\beta} & \psi_{r\alpha} & \psi_{r\beta} \end{bmatrix}^T \\ \vec{u} &= \begin{bmatrix} u_{s\alpha} & u_{s\beta} \end{bmatrix}^T \\ \vec{y} &= \begin{bmatrix} i_{s\alpha} & i_{s\beta} \end{bmatrix}^T \\ \tilde{\mathbf{A}} &= \begin{bmatrix} -\gamma & 0 & \beta\alpha & \beta\omega_m^0 \\ 0 & -\gamma & -\beta\omega_m^0 & \beta\alpha \\ \alpha L_m & 0 & -\alpha & -\omega_m^0 \\ 0 & \alpha L_m & \omega_m^0 & -\alpha \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} \frac{1}{\sigma} & 0 \\ 0 & \frac{1}{\sigma} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \mathbf{C} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\end{aligned}\tag{9.9.5}$$

The expression of the electromagnetic torque is as follows

$$\tau_e = \frac{3}{2} \frac{L_m}{L_r} (\psi_{r\alpha} i_{s\beta} - \psi_{r\beta} i_{s\alpha})\tag{9.9.6}$$

Here we present a short nomenclature for the IM model

- $i_{s\alpha}(t)$  - is the  $\alpha$  component of the stator current.
- $i_{s\beta}(t)$  - is the  $\beta$  component of the stator current.
- $u_{s\alpha}(t)$  - is the  $\alpha$  component of the stator input voltage.
- $u_{s\beta}(t)$  - is the  $\beta$  component of the stator input voltage.
- $\psi_{r\alpha}(t)$  - is the  $\alpha$  component of the permanent magnet rotor flux linkage to the stator.
- $\psi_{r\beta}(t)$  - is the  $\beta$  component of the permanent magnet rotor flux linkage to the stator.
- $\omega_m^0$  - is the mechanical rotor speed - constant.
- $\vartheta_e(t)$  - is the stator electrical phase.
- $\tau_e(t)$  - is the electromagnetic torque.
- $L_s = L_\sigma + L_m$  is the stator inductance.
- $L_r = L_\sigma + L_m$  is the rotor inductance.
- $L_\sigma$  is the leakage inductance.
- $L_m$  is the mutual inductance.

- $R_s$  is the stator resistance.
- $R_r$  is the rotor resistance.

We are supposing that the number of pole pairs is one.

### 9.9.1 Rotor Flux Observer

The rotor fluxes observer has been implemented by the use of the equivalent IM model represented in the Eqs. (9.9.4) to (9.9.6). This model is a LTI system in the continuous time domain and to be suitable to be used with the Kalman Filter Algorithm must be discretized according to

$$\mathbf{A} = \mathbf{I} + \tilde{\mathbf{A}} t_s \quad (9.9.7)$$

$$\mathbf{B} = \tilde{\mathbf{B}} t_s \quad (9.9.8)$$

After that the observer algorithm can be implemented.

#### The Kalman Filter Algorithm

$$\hat{\vec{x}}(k|k) = \mathbf{A} \hat{\vec{x}}(k|k-1) + \mathbf{B} \vec{u}(k|k-1) \quad (9.9.9)$$

$$\mathbf{P}(k|k) = \mathbf{A} \mathbf{P}(k|k-1) \mathbf{A}^T + \mathbf{Q} \quad (9.9.10)$$

$$\mathbf{L}(k) = \mathbf{P}(k|k) \mathbf{C}^T \left( \mathbf{C} \mathbf{P}(k|k) \mathbf{C}^T + \mathbf{R} \right)^{-1} \quad (9.9.11)$$

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \left( \vec{y}(k) - \mathbf{C} \hat{\vec{x}}(k|k) \right) \quad (9.9.12)$$

$$\mathbf{P}(k+1|k) = \left( \mathbf{I} - \mathbf{L}(k) \mathbf{C} \right) \mathbf{P}(k|k) \quad (9.9.13)$$

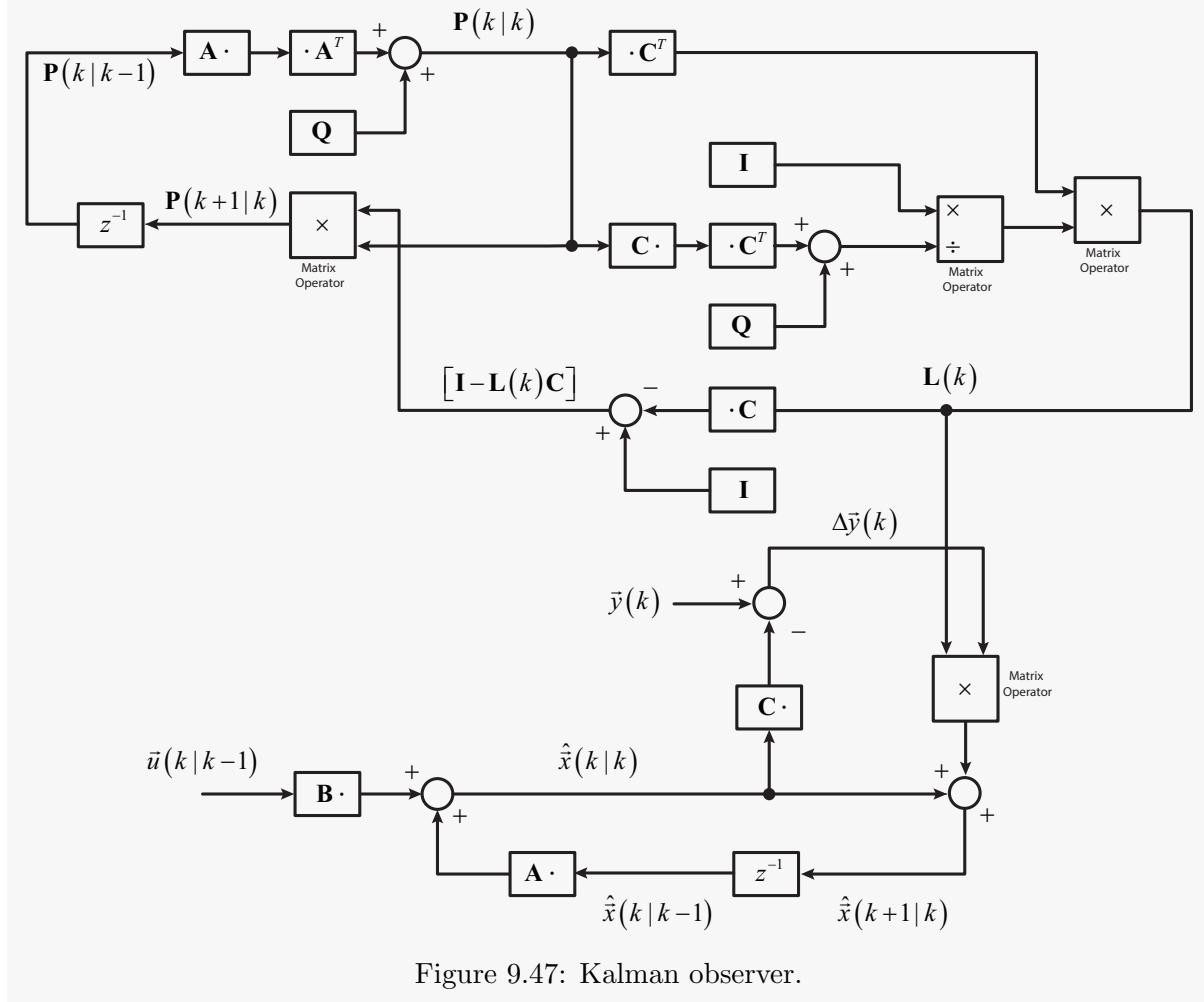


Figure 9.47: Kalman observer.

### 9.9.2 Torque Control

In this section we show the relation between the electromagnetic torque of the IM and the stator current  $i_{s\alpha}(t)$  and  $i_{s\beta}(t)$ , and in particular will derive the relation between the desired electromagnetic torque  $\tau_m^{ref}$  and the equivalent stator current references  $i_{s\alpha}^{ref}(t)$  and  $i_{s\beta}^{ref}(t)$ .

Induction motor needs to be energized, or in other words the rotating magnetic field must be established. To create of the magnetic field a constant amplitude current is created, this current component is called *magnetization current*.

Let  $i_m = \psi_m^{ref}/L_m$  be the amplitude of the magnetization current, and let  $i^\tau(t)$  be the amount of current required to generate a given torque, the set of current references  $i_{s\alpha}^{ref}(t)$  and  $i_{s\beta}^{ref}(t)$  are as follows

$$i_{s\alpha}^{ref}(t) = \frac{\psi_m^{ref}}{L_m} \cos \hat{\vartheta}_e(t) - i^\tau(t) \sin \hat{\vartheta}_e(t) \quad (9.9.14)$$

$$i_{s\beta}^{ref}(t) = i^\tau(t) \cos \hat{\vartheta}_e(t) + \frac{\psi_m^{ref}}{L_m} \sin \hat{\vartheta}_e(t)$$

where  $i_m = \psi_m^{ref}/L_m = \text{constant}$  and

$$\hat{\vartheta}_e(t) = \arctan(\hat{\psi}_{r\alpha}(t), \hat{\psi}_{r\beta}(t)) \quad (9.9.15)$$

and where  $\hat{\vartheta}_e(t) = \hat{\vartheta}_{slip}(t) + \vartheta_m(t)$  and  $\hat{\omega}_e(t) = \hat{\omega}_{slip}(t) + \omega_m^0$ .

The rotor fluxes can be represented by the following equations

$$\begin{aligned} \psi_{r\alpha}(t) &= \psi_m^{ref} \cos(\vartheta_e) \\ \psi_{r\beta}(t) &= \psi_m^{ref} \sin(\vartheta_e) \end{aligned} \quad (9.9.16)$$

combing Eq. (9.9.14) and Eq. (9.9.16) with Eq. (9.9.6) we obtain an expression of the torque as follows

$$\begin{aligned} \tau_e &= \frac{3}{2} \frac{L_m}{L_r} (\psi_{r\alpha} i_{s\beta} - \psi_{r\beta} i_{s\alpha}) \\ &= \frac{3}{2} \frac{L_m}{L_r} \left[ \psi_m^{ref} \cos \vartheta_e \left( i^\tau \cos \hat{\vartheta}_e + \frac{\psi_m^{ref}}{L_m} \sin \hat{\vartheta}_e \right) \right. \\ &\quad \left. - \psi_m^{ref} \sin \vartheta_e \left( \frac{\psi_m^{ref}}{L_m} \cos \hat{\vartheta}_e - i^\tau \sin \hat{\vartheta}_e \right) \right] \\ &= \frac{3}{2} \frac{L_m}{L_r} \psi_m^{ref} i^\tau \quad \text{if } \hat{\vartheta}_e - \vartheta_e \rightarrow 0 \end{aligned} \quad (9.9.17)$$

From the above equations we can understand that the estimation of rotor fluxes  $\hat{\psi}_{r\alpha}$  and  $\hat{\psi}_{r\beta}$  is required in order to derive the electrical phase  $\hat{\vartheta}_e(t)$ , and the accuracy or in general the performance of the state observer affects the performance of the torque actuation.<sup>††</sup>

The last step in our control design is the implementation of the current controller based on PI. These current controllers (they are two: one for the  $x$  component and one for the  $y$  component) work respect to the  $xy$ -synchronous reference frame. The natural step now is to show the structure of the reference frame transformation.

---

<sup>††</sup>Supposing  $\omega_m(t)$  is also measured, an alternative implementation of the for the rotor fluxes  $\hat{\psi}_{r\alpha}$  and  $\hat{\psi}_{r\beta}$  estimator can be the *open loop observer*.

Eq. (9.9.18) shows the equation of the rotor flux respect to the stationary reference frame

$$\begin{aligned} \frac{d\hat{\psi}_{r\alpha}(t)}{dt} &= -\alpha \hat{\psi}_{r\alpha}(t) - \omega_m(t) \hat{\psi}_{r\beta}(t) + \alpha L_m i_{s\alpha}(t) \\ \frac{d\hat{\psi}_{r\beta}(t)}{dt} &= -\alpha \hat{\psi}_{r\beta}(t) + \omega_m(t) \hat{\psi}_{r\alpha}(t) + \alpha L_m i_{s\beta}(t) \end{aligned} \quad (9.9.18)$$

where the discrete time representation can be as follows

$$\begin{aligned} \hat{\psi}_{r\alpha}(k+1) &= (1 - \alpha) \hat{\psi}_{r\alpha}(k) - t_s \omega_m(k) \hat{\psi}_{r\beta}(k) + t_s \alpha L_m i_{s\alpha}(k) \\ \hat{\psi}_{r\beta}(k+1) &= (1 - \alpha) \hat{\psi}_{r\beta}(k) + t_s \omega_m(k) \hat{\psi}_{r\alpha}(k) + t_s \alpha L_m i_{s\beta}(k) \end{aligned} \quad (9.9.19)$$

This observer requires the measurement of  $\omega_m$ ,  $i_{s\alpha}$ ,  $i_{s\beta}$  and depends on the motor parameters  $R_r$ ,  $L_r$  and  $L_m$ .

Consider first, the transformation between stationary reference frame and the synchronous reference frame as follows

$$\mathbf{T}_2(\hat{\vartheta}_e) = \begin{bmatrix} \cos \hat{\vartheta}_e & \sin \hat{\vartheta}_e \\ -\sin \hat{\vartheta}_e & \cos \hat{\vartheta}_e \end{bmatrix} \quad \mathbf{T}_2^{-1}(\hat{\vartheta}_e) = \begin{bmatrix} \cos \hat{\vartheta}_e & -\sin \hat{\vartheta}_e \\ \sin \hat{\vartheta}_e & \cos \hat{\vartheta}_e \end{bmatrix} \quad (9.9.20)$$

such that

$$\begin{bmatrix} i_{sx} \\ i_{sy} \end{bmatrix} = \begin{bmatrix} \cos \hat{\vartheta}_e & \sin \hat{\vartheta}_e \\ -\sin \hat{\vartheta}_e & \cos \hat{\vartheta}_e \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix} \quad (9.9.21)$$

and

$$\begin{bmatrix} u_{s\alpha} \\ u_{s\beta} \end{bmatrix} = \begin{bmatrix} \cos \hat{\vartheta}_e & -\sin \hat{\vartheta}_e \\ \sin \hat{\vartheta}_e & \cos \hat{\vartheta}_e \end{bmatrix} \begin{bmatrix} u_{sx} \\ u_{sy} \end{bmatrix} \quad (9.9.22)$$

By these transformation we can implement the current controllers in the synchronous reference frame domain where quantities which were rotating at pulsation  $\omega_e$  are now mapped into a dc domain.

The overall architecture can be represented as in Figure 9.46 where

$$i_{sx}^{ref}(t) = \frac{\psi_m^{ref}}{L_m} \quad (9.9.23)$$

$$i_{sy}^{ref}(t) = i^\tau(t)$$

### 9.9.3 Simulation Results

Consider the following induction motor data

$P_{nom} = 250 \text{ kW}$	$I_{nom} = 432 \text{ A (rms)}$	$V_{nom} = 400 \text{ V (rms)}$
$I_{bez} = 611 \text{ A (peak)}$	$U_{bez} = 326.6 \text{ V (peak)}$	$R_s = 2.213 \text{ m}\Omega$
$L_s = 5.445 \text{ mH}$	$R_r = 4.737 \text{ m}\Omega$	$L_r = 5.445 \text{ mH}$
$L_m = 5.326 \text{ mH}$	$\Psi^m = 1 \text{ Wb m}^{-2}$	$J = 1 \text{ kg m}^2$

Table 9.4: IM model parameters.

Applying the control architecture shown in Figure 9.47 consider the following data

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10^{-4} & 0 \\ 0 & 0 & 0 & 10^{-4} \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

The current controller are implemented by simple PI where  $k_p = 0.5$ ,  $k_i = 18$  and sampling time  $t_s = 125\mu s$ .

The internal control quantities (current and voltages) are normalized respect to the quantities  $I_{bez}$  and  $U_{bez}$ .

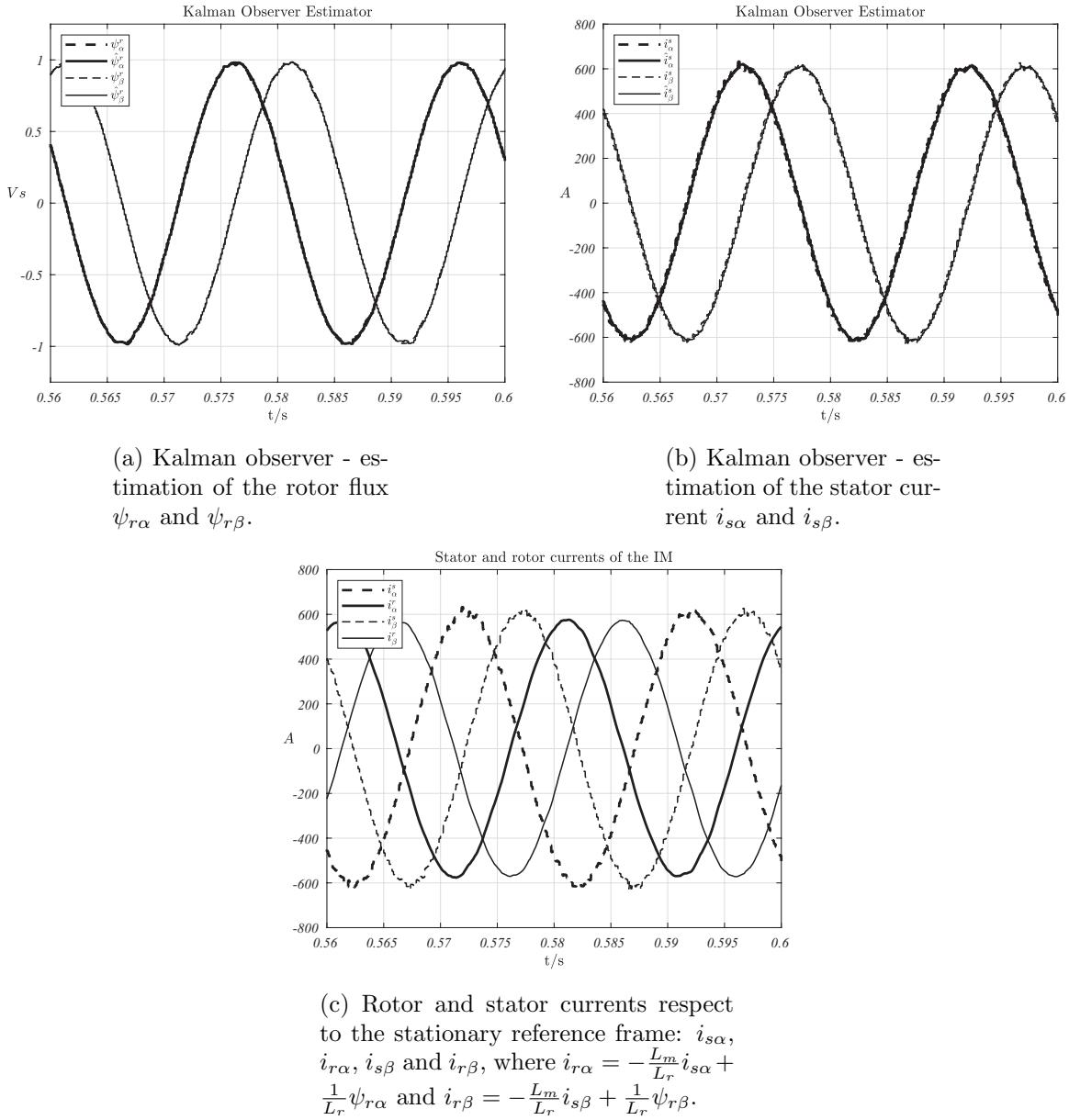


Figure 9.48: Simulation results.

## 9.10 Control of an Harmonic in Harsh Environment

This document shows a case study concerning the tracking problem of a given harmonic  $h^{\text{ref}}(t)$  signal on a generic second order plant subject to strong distortion and white noise. We can suppose the measured quantity  $y(t)$  is composed by our *under control* signal  $h(t) = H \cos [\theta(t) -$

$\phi]$  subjected to a generic distortion and white noise, which can be generalized as follows

$$\begin{aligned} y(t) &= h(t) + \sum_{n=0}^{+\infty} a_n \varphi_n(t) + w(t) \\ &= H \cos [\theta(t) - \phi] + \sum_{n=0}^{+\infty} a_n \varphi_n(t) + w(t) \end{aligned} \quad (9.10.1)$$

The following plant has been taken into account

$$H_{plant} = \frac{\omega_p^2}{s^2 + 2\zeta\omega_p s + \omega_p^2} \quad (9.10.2)$$

where  $\omega_p = 2\pi 400 \text{ rad s}^{-1}$  and  $\zeta = 1$

### 9.10.1 Magnitude control design

The control architecture which is intended to design shall be able to extract the quantity  $h(t)$  from the measured signal  $y(t)$  and shall be able to control its amplitude. The reference harmonic is a preassigned signal with a given amplitude, angular frequency and initial phase, as follows

$$h^{\text{ref}}(t) = H^{\text{ref}}(t) \cos [\omega t - \phi] \quad (9.10.3)$$

where  $H^{\text{ref}}(t)$  is the amplitude reference,  $\omega$  is the angular frequency and  $\phi$  the initial phase. The main control issue lay in the useful harmonic extraction  $h(t)$  from the noised and corrupted signal  $y(t)$ .

A possible solution to extract a given information immersed into a strong distorted signals is to use an oscillator observer with optimal gain computed by Kalman filter approach. The useful quantity ( $h[\vartheta(t)]$ ) extracted from the *internal model* filter is then  $\pi/2$  time shifted ( $h[\vartheta(t) - \pi/2]$ ) in order to derive two orthogonal signals; hence, to calculate the amplitude. See Figure 9.49

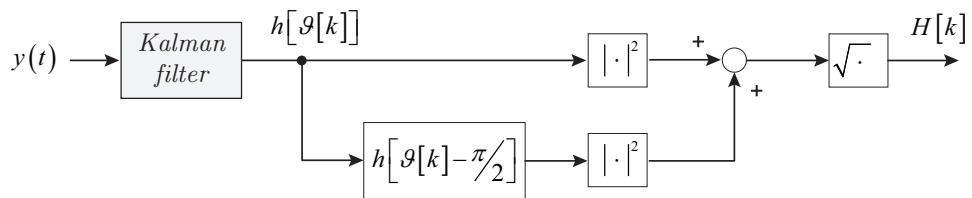


Figure 9.49: Harmonic magnitude extraction block.

The Kalman Filter which is here taken into account is based on a oscillator model as follows

$$\begin{cases} \dot{x}_1(t) = \omega x_2(t) \\ \dot{x}_2(t) = -\omega x_1(t) \end{cases} \quad (9.10.4)$$

which results in the following state space representation

$$\begin{cases} \dot{\vec{x}}(t) = \tilde{\mathbf{A}}\vec{x}(t) \\ h(t) = \mathbf{C}\vec{x}(t) \end{cases} \quad (9.10.5)$$

where  $\tilde{\mathbf{A}} = \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}$  and  $\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ .

After the discretization we obtain the following transition matrix

$$\mathbf{A} = \begin{bmatrix} \cos(\omega t_s) & \sin(\omega t_s) \\ -\sin(\omega t_s) & \cos(\omega t_s) \end{bmatrix} \quad (9.10.6)$$

**The Kalman Filter Algorithm become as follows**

$$\hat{\vec{x}}(k|k) = \mathbf{A}\hat{\vec{x}}(k|k-1) \quad (9.10.7)$$

$$\mathbf{P}(k|k) = \mathbf{A}\mathbf{P}(k|k-1)\mathbf{A}^T + \mathbf{Q} \quad (9.10.8)$$

$$\mathbf{L}(k) = \mathbf{P}(k|k)\mathbf{C}^T \left( \mathbf{C}\mathbf{P}(k|k)\mathbf{C}^T + \mathbf{R} \right)^{-1} \quad (9.10.9)$$

$$\hat{\vec{x}}(k+1|k) = \hat{\vec{x}}(k|k) + \mathbf{L}(k) \left( \vec{y}(k) - \mathbf{C}\hat{\vec{x}}(k|k) \right) \quad (9.10.10)$$

$$\mathbf{P}(k+1|k) = (\mathbf{I} - \mathbf{L}(k)\mathbf{C})\mathbf{P}(k|k) \quad (9.10.11)$$

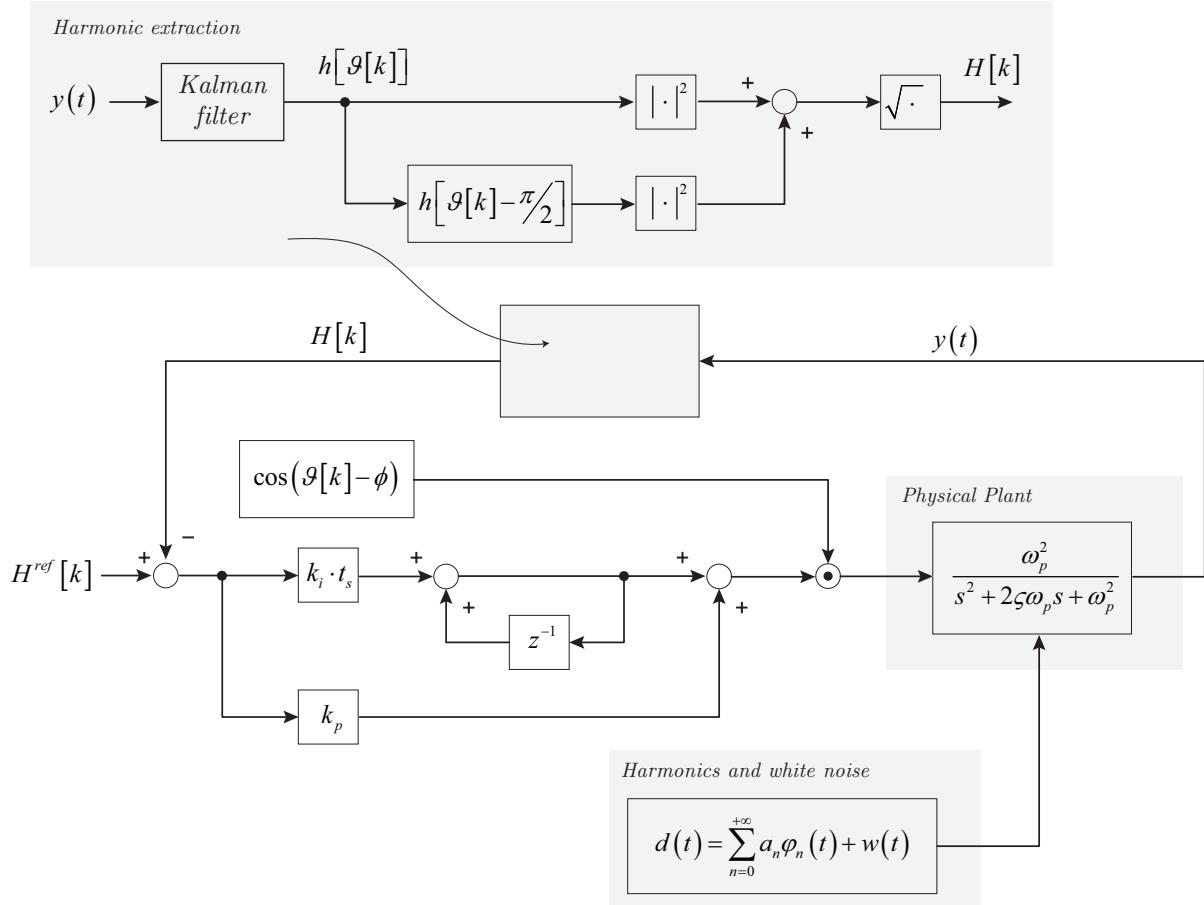


Figure 9.50: Global control layout.

Once the  $\hat{h}[k] \cos[\omega k]$  and  $\hat{h}[k] \sin[\omega k]$  signals have been extracted from the measured signal  $y(t)$  the amplitude  $H(t)$  shall be derived. The global control can be seen in Figure 9.50

### 9.10.2 Simulation results

Simulation results for

$$\mathbf{Q} = \begin{bmatrix} 10^{-5} & 0 \\ 0 & 10^{-5} \end{bmatrix} \quad \mathbf{R} = 10^4 \quad (9.10.12)$$

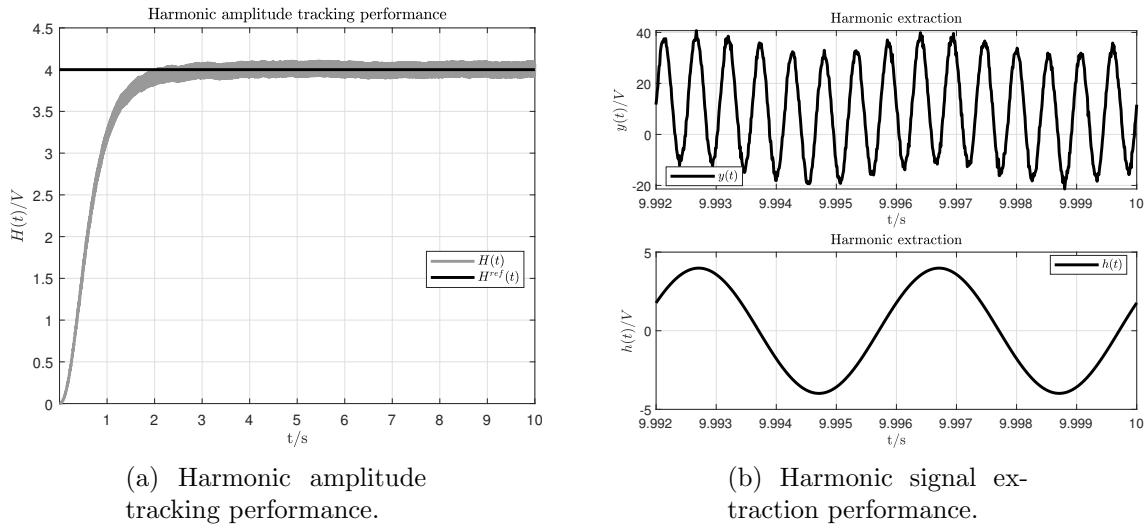


Figure 9.51: Simulation results.

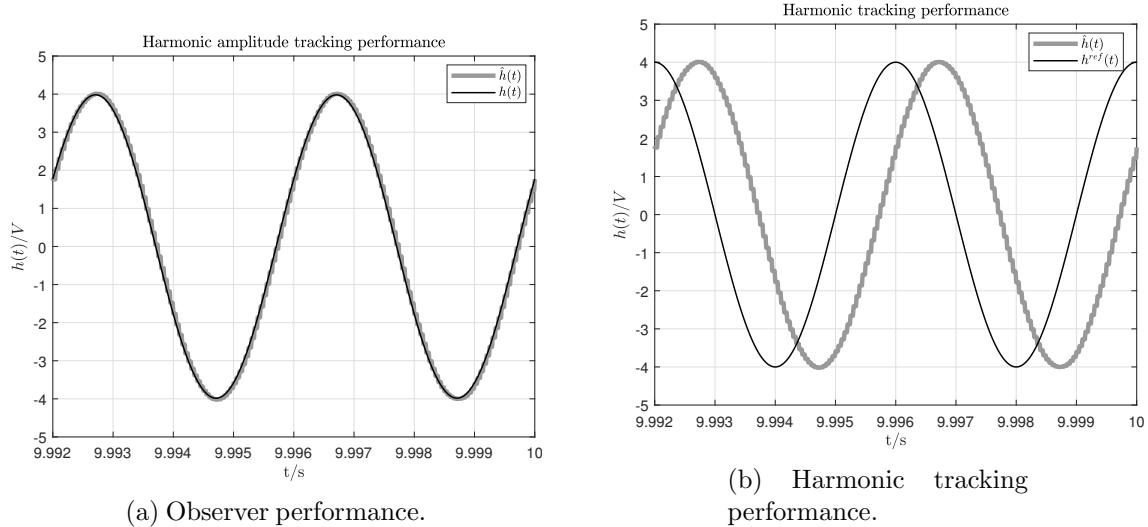


Figure 9.52: Simulation results.

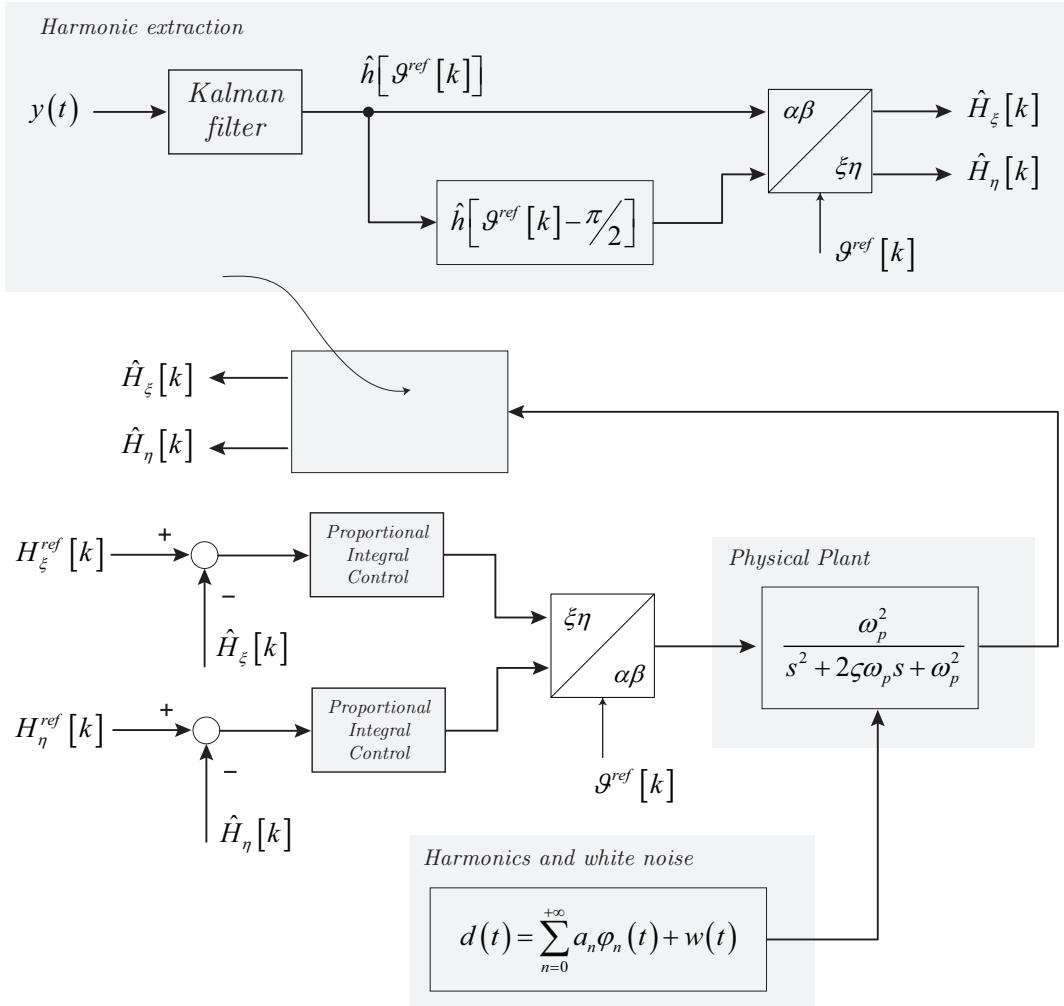


Figure 9.53: Global control layout.

### 9.10.3 Magnitude and phase control design

In order to control both amplitude and phase a *vector* for of the previous architecture control must be implemented. Figure 9.53 shows a possible implementation which use a change of reference. Along this proposed implementation the Kalman filter based on the internal model of an oscillator is still used in order to extract the a given harmonic signal ( $h(t)$ ) from a generic distorted measure  $y(t)$ . The change of reference is given by the following hermitian operators

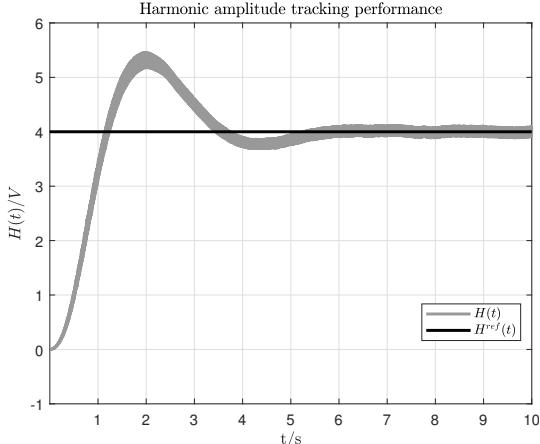
$$\xi^\eta T_{\alpha\beta}(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \quad (9.10.13)$$

$$\alpha^\beta T_{\xi\eta}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (9.10.14)$$

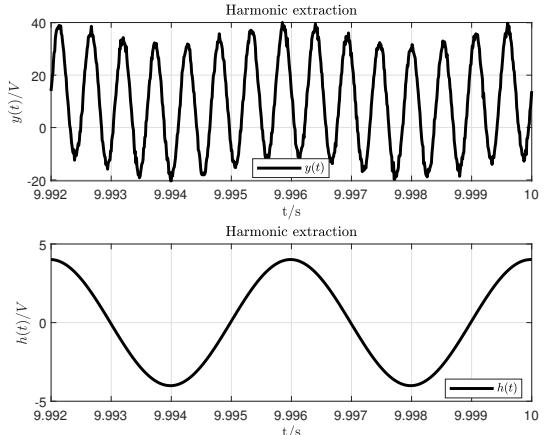
#### 9.10.4 Simulation results

Simulation results for

$$\mathbf{Q} = \begin{bmatrix} 10^{-5} & 0 \\ 0 & 10^{-5} \end{bmatrix} \quad \mathbf{R} = 10^4 \quad (9.10.15)$$

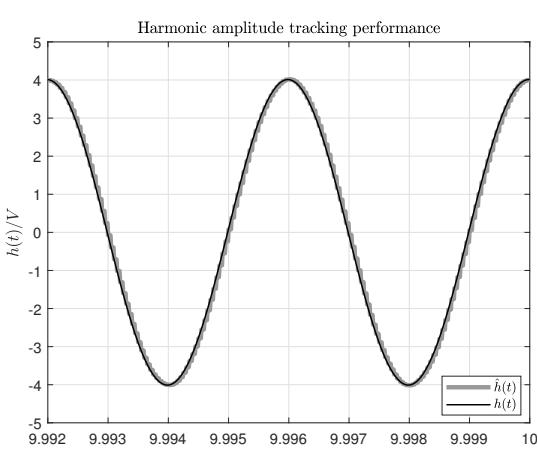


(a) Harmonic amplitude tracking performance.

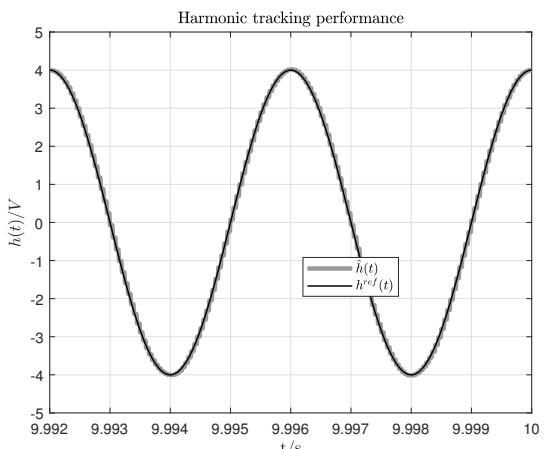


(b) Harmonic signal extraction performance.

Figure 9.54: Simulation results.



(a) Observer performance.



(b) Harmonic tracking performance.

Figure 9.55: Simulation results.

### 9.11 Magnetic Bearing Modelization and Control

The following chapter concerns the modelization and control of a two poles magnetic bearing.

### 9.11.1 Nomenclature

Here the list of variables and parameters used along the document.

– Variables:

- $y(t)$   $\left[ \text{m} \right]$ : shaft displacement from the nominal position;
- $v(t)$   $\left[ \text{m s}^{-1} \right]$ : shaft linear velocity along the control axis;
- $u_i(t)$   $\left[ \text{V} \right]$ : coils voltage input;
- $i_i(t)$   $\left[ \text{A} \right]$ : coils current;
- $R_i(t)$   $\left[ \Omega \right]$ : coils resistance;
- $L_i(t)$   $\left[ \text{H} \right]$ : effective electromagnets inductance;
- $f_i(t)$   $\left[ \text{N} \right]$ : electromagnets force;
- $f_{load}(t)$   $\left[ \text{N} \right]$ : external force.

– Parameters:

- $\lambda_i$   $\left[ \text{H m} \right]$ : linear electromagnets inductance as function of the air-gap distance;
- $s$   $\left[ \text{m} \right]$ : nominal air-gap;
- $y_{\max}$   $\left[ \text{m} \right]$ : maximum shaft displacement;
- $\sigma$   $\left[ \text{Ns m}^{-1} \right]$ : viscosity around the shaft;
- $g$   $\left[ \text{m s}^{-2} \right]$ : gravitational acceleration.

– Electromagnetic quantities (reported in scalar form):

- $H$   $\left[ \text{A m}^{-1} \right]$ : magnetic field intensity;
- $B$   $\left[ \text{T} \right]$ : magnetic flux density;
- $\phi$   $\left[ \text{Wb} \right]$ : magnetic flux;
- $M$   $\left[ \text{A m}^{-1} \right]$ : magnetization;
- $\mu_0$   $\left[ \text{H m}^{-1} \right]$ : vacuum permeability.

### 9.11.2 Model Derivation

In this section the physical-model of the magnetic bearing is presented. For physical model is intended a model which is closed to the physical mathematical representation of the original system, and it will be implemented as plant emulator. The physical-model will be implemented in *Simscape* ambient.

Denoting by  $y$  (see Figure 9.56) the difference between the position of the center of the rotor

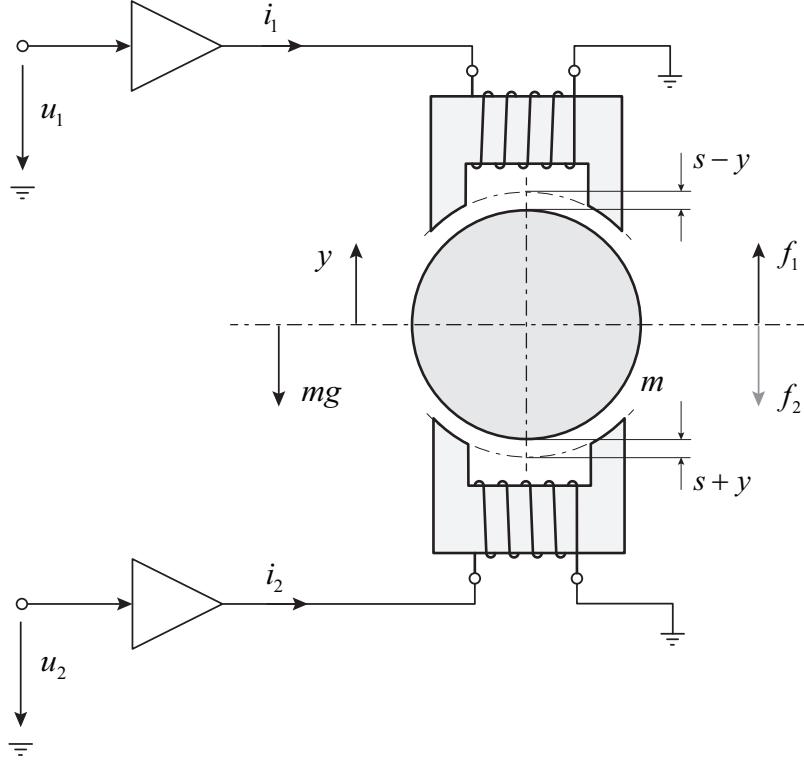


Figure 9.56: Two poles pairs bearing magnets.

and its nominal position;  $y$ -axis is being oriented upwards;  $s$  is the nominal air gap,  $f_1$  and  $f_2$  are the forces created by the two electromagnets;  $i_1$  and  $i_2$  are the electromagnet currents.

The magnetic bearing model, for the open domain  $|y| < y_{\max} < s$  will be represented as follows

$$\dot{y} = v \quad (9.11.1)$$

$$\dot{v} = \frac{1}{m}(f_1 - f_2 - mg - \sigma v) \quad (9.11.2)$$

with<sup>‡‡</sup>

$$f_1 = \frac{\lambda_1(i_1)^2}{(s - y)^2}, \quad f_2 = \frac{\lambda_2(i_2)^2}{(s + y)^2} \quad (9.11.4)$$

The term  $\sigma$  represents the viscosity coefficient for translational displacement of the rotor shaft, and  $mg$  the weight force.

<sup>‡‡</sup>Electromagnet forces are parabolic-depending with the current and hyperbolic-depending with the effective air-gap displacement. Moreover, the equivalent force applied to the shaft results as follows

$$f_y = f_1 - f_2 - mg - f_{load} \quad (9.11.3)$$

The currents  $i_1$  and  $i_2$  must satisfy the *Kirchhoff's Voltage Law*; it follows that

$$u_1 = R_1 i_1 + L_1 \frac{di_1}{dt} + i_1 \frac{dL_1}{dt} \quad (9.11.5)$$

$$u_2 = R_2 i_2 + L_2 \frac{di_2}{dt} + i_2 \frac{dL_2}{dt} \quad (9.11.6)$$

with  $u_i$  the input voltage and  $R_i$  the resistance of the circuit  $i$ ,  $i = 1, 2$ , and with inductances<sup>§§</sup>

$$L_1(y) = \frac{\lambda_1}{s - y} \quad (9.11.7)$$

$$L_2(y) = \frac{\lambda_2}{s + y} \quad (9.11.8)$$

where their dynamic can be expressed as follows

$$\frac{dL_1}{dt}(t) = \frac{\partial L_1}{\partial y} \frac{dy}{dt} = \frac{\lambda_1}{(s - y)^2} v, \quad L_1(0) = \frac{\lambda_1}{s} \quad (9.11.9)$$

$$\frac{dL_2}{dt}(t) = \frac{\partial L_2}{\partial y} \frac{dy}{dt} = -\frac{\lambda_2}{(s + y)^2} v, \quad L_2(0) = \frac{\lambda_2}{s} \quad (9.11.10)$$

Considering the following operative rotor position domain

$$-s < -y_{\max} \leq y \leq y_{\max} < s \quad (9.11.11)$$

Coupling the above equations, the physical-model of the magnetic bearing is determined.

**The full physical-model implementation results as follows**

**for**  $|y| \leq y_{\max} < s$

$$\dot{y}(t) = v(t), \quad y(0) = 0 \quad (9.11.12)$$

$$\dot{v}(t) = \frac{1}{m} [f_1(t) - f_2(t) - mg - \sigma v(t) - f_{load}(t)], \quad v(0) = 0 \quad (9.11.13)$$

$$u_1(t) = R_1 i_1(t) + L_1(t) \frac{di_1}{dt}(t) + i_1(t) \frac{dL_1}{dt}(t), \quad i_1(0) = i_{10} \quad (9.11.14)$$

$$u_2(t) = R_2 i_2(t) + L_2(t) \frac{di_2}{dt}(t) + i_2(t) \frac{dL_2}{dt}(t), \quad i_2(0) = 0 \quad (9.11.15)$$

$$\frac{dL_1}{dt}(t) = \frac{\lambda_1}{[s - y(t)]^2} v(t), \quad L_1(0) = \frac{\lambda_1}{s} \quad (9.11.16)$$

---

<sup>§§</sup>Electromagnet inductances are directly linked to the geometry of the magnetic bearing. Increasing of the effective air-gap in one pole (decreased for the opposite) reduces (increases for the opposite) its nominal value.

$$\frac{dL_2}{dt}(t) = -\frac{\lambda_2}{[s + y(t)]^2}v(t), \quad L_2(0) = \frac{\lambda_2}{s} \quad (9.11.17)$$

$$f_1(t) = \frac{\lambda_1 [i_1(t)]^2}{[s - y(t)]^2} \quad (9.11.18)$$

$$f_2(t) = \frac{\lambda_2 [i_2(t)]^2}{[s + y(t)]^2} \quad (9.11.19)$$

$f_{load}(t)$  is an unknown external force (disturbance).

**for**  $|y| > y_{\max} < s$

$$\dot{y}(t) = v(t), \quad y(0) = 0 \quad (9.11.20)$$

$$\dot{v}(t) = \frac{1}{m} \left[ f_1(t) - f_2(t) - mg - \sigma_{rw}v(t) - f_{load}(t) \right], \quad v(0) = 0 \quad (9.11.21)$$

$$u_1(t) = R_1 i_1(t) + L_1(t) \frac{di_1}{dt}(t) + i_1(t) \frac{dL_1}{dt}(t), \quad i_1(0) = i_{10} \quad (9.11.22)$$

$$u_2(t) = R_2 i_2(t) + L_2(t) \frac{di_2}{dt}(t) + i_2(t) \frac{dL_2}{dt}(t), \quad i_2(0) = 0 \quad (9.11.23)$$

$$\frac{dL_1}{dt}(t) = \frac{\lambda_1}{[s - y(t)]^2}v(t), \quad L_1(0) = \frac{\lambda_1}{s} \quad (9.11.24)$$

$$\frac{dL_2}{dt}(t) = -\frac{\lambda_2}{[s + y(t)]^2}v(t), \quad L_2(0) = \frac{\lambda_2}{s} \quad (9.11.25)$$

$$f_1(t) = \frac{\lambda_1 [i_1(t)]^2}{[s - y(t)]^2} \quad (9.11.26)$$

$$f_2(t) = \frac{\lambda_2 [i_2(t)]^2}{[s + y(t)]^2} \quad (9.11.27)$$

where  $\sigma_{rw} \gg \sigma$  represents an artificial modelization of the rotor hard-stop, and  $f_{load}(t)$  is an unknown external force (disturbance).

### 9.11.3 Model Linearization

For control purpose a linearization of the physical-model (which results in a nonlinear system) around the nominal position of the rotor is, in the following, performed.

From Eqs. (9.11.12)–(9.11.19) it is possible to re-write the physical-model in the form (as-

suming  $f_{load} = 0$ )

$$\dot{y} = v \quad (9.11.28)$$

$$\dot{v} = \frac{1}{m} \left[ \frac{\lambda(i_1)^2}{(s-y)^2} - \frac{\lambda(i_2)^2}{(s+y)^2} - mg - \sigma v \right] \quad (9.11.29)$$

$$u_1 = R i_1 + \frac{\lambda}{s-y} \frac{di_1}{dt} + i_1 \frac{\lambda}{(s-y)^2} v \quad (9.11.30)$$

$$u_2 = R i_2 + \frac{\lambda}{s+y} \frac{di_2}{dt} - i_2 \frac{\lambda}{(s+y)^2} v \quad (9.11.31)$$

where, it is assumed,  $\lambda_1 = \lambda_2 = \lambda$ , and  $R_1 = R_2 = R$ . Rearranging, for canonical form, Eqs. (9.11.28) – (9.11.31), they can be rewritten as follows

$$\dot{y} = v \quad (9.11.32)$$

$$\dot{v} = \frac{1}{m} \left[ \frac{\lambda(i_1)^2}{(s-y)^2} - \frac{\lambda(i_2)^2}{(s+y)^2} - mg - \sigma v \right] \quad (9.11.33)$$

$$\frac{di_1}{dt} = -\frac{R}{\lambda}(s-y)i_1 - v \frac{1}{s-y}i_1 + \frac{s-y}{\lambda}u_1 \quad (9.11.34)$$

$$\frac{di_2}{dt} = -\frac{R}{\lambda}(s+y)i_2 + v \frac{1}{s+y}i_2 + \frac{s+y}{\lambda}u_2 \quad (9.11.35)$$

Considering the state vector  $\vec{x} = [y \ v \ i_1 \ i_2]^T$  and the input vector  $\vec{u} = [u_1 \ u_2]^T$ , the system of Eqs. (9.11.32) – (9.11.35) is a nonlinear system of the form

$$\dot{\vec{x}} = \vec{g}(\vec{x}) + \vec{h}(\vec{x}, \vec{u}) \quad (9.11.36)$$

$$\vec{y} = \mathbf{C}\vec{x} \quad (9.11.37)$$

where

$$g_1(\vec{x}) = v \quad (9.11.38)$$

$$g_2(\vec{x}) = \frac{1}{m} \left[ \frac{\lambda(i_1)^2}{(s-y)^2} - \frac{\lambda(i_2)^2}{(s+y)^2} - mg - \sigma v \right] \quad (9.11.39)$$

$$g_3(\vec{x}) = -\frac{R}{\lambda}(s-y)i_1 - v \frac{1}{s-y}i_1 \quad (9.11.40)$$

$$g_4(\vec{x}) = -\frac{R}{\lambda}(s+y)i_2 + v \frac{1}{s+y}i_2 \quad (9.11.41)$$

and

$$h_1(\vec{x}, \vec{u}) = 0 \quad (9.11.42)$$

$$h_2(\vec{x}, \vec{u}) = 0 \quad (9.11.43)$$

$$h_3(\vec{x}, \vec{u}) = \frac{s-y}{\lambda} u_1 \quad (9.11.44)$$

$$h_4(\vec{x}, \vec{u}) = \frac{s+y}{\lambda} u_2 \quad (9.11.45)$$

The linearization problem is boiled out linearising the model around the operative point  $i_{20} = 0, y_0 = 0, v_0 = 0, u_{20}$  and

$$i_{10} = \sqrt{\frac{mgs^2}{\lambda}} \Rightarrow u_{10} = R_1 \sqrt{\frac{mgs^2}{\lambda}} \quad (9.11.46)$$

Consider the linearization of the model of Eqs. (9.11.32) – (9.11.35) around the operative point

$$\vec{x}_0 = \begin{bmatrix} y_0 \\ v_0 \\ i_{10} \\ i_{20} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sqrt{\frac{mgs^2}{\lambda}} \\ 0 \end{bmatrix}, \quad \vec{u}_0 = \begin{bmatrix} u_{10} \\ u_{20} \end{bmatrix} = \begin{bmatrix} R_1 \sqrt{\frac{mgs^2}{\lambda}} \\ 0 \end{bmatrix} \quad (9.11.47)$$

it results that the transition matrix of the corresponding LTI system becomes

$$a_{11} = \frac{\partial g_1}{\partial y} \Big|_{\vec{x}_0} + \frac{\partial h_1}{\partial y} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.48)$$

$$a_{12} = \frac{\partial g_1}{\partial v} \Big|_{\vec{x}_0} + \frac{\partial h_1}{\partial v} \Big|_{\vec{x}_0, \vec{u}_0} = 1 \quad (9.11.49)$$

$$a_{13} = \frac{\partial g_1}{\partial i_1} \Big|_{\vec{x}_0} + \frac{\partial h_1}{\partial i_1} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.50)$$

$$a_{14} = \frac{\partial g_1}{\partial i_2} \Big|_{\vec{x}_0} + \frac{\partial h_1}{\partial i_2} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.51)$$

$$a_{21} = \frac{\partial g_2}{\partial y} \Big|_{\vec{x}_0} + \frac{\partial h_2}{\partial y} \Big|_{\vec{x}_0, \vec{u}_0} = \frac{2\lambda}{m} \frac{(i_{10})^2}{s^3} = \frac{2g}{s} \quad (9.11.52)$$

$$a_{22} = \frac{\partial g_2}{\partial v} \Big|_{\vec{x}_0} + \frac{\partial h_2}{\partial v} \Big|_{\vec{x}_0, \vec{u}_0} = -\frac{\sigma}{m} \quad (9.11.53)$$

$$a_{23} = \frac{\partial g_2}{\partial i_1} \Big|_{\vec{x}_0} + \frac{\partial h_2}{\partial i_1} \Big|_{\vec{x}_0, \vec{u}_0} = \frac{2}{s} \sqrt{\frac{\lambda g}{m}} \quad (9.11.54)$$

$$a_{24} = \frac{\partial g_2}{\partial i_2} \Big|_{\vec{x}_0} + \frac{\partial h_2}{\partial i_2} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.55)$$

$$a_{31} = \frac{\partial g_3}{\partial y} \Big|_{\vec{x}_0} + \frac{\partial h_3}{\partial y} \Big|_{\vec{x}_0, \vec{u}_0} = R(1-R) \sqrt{\frac{mgs^2}{\lambda^3}} \quad (9.11.56)$$

$$a_{32} = \frac{\partial g_3}{\partial v} \Big|_{\vec{x}_0} + \frac{\partial h_3}{\partial v} \Big|_{\vec{x}_0, \vec{u}_0} = -\sqrt{\frac{mg}{\lambda}} \quad (9.11.57)$$

$$a_{33} = \frac{\partial g_3}{\partial i_1} \Big|_{\vec{x}_0} + \frac{\partial h_3}{\partial i_1} \Big|_{\vec{x}_0, \vec{u}_0} = -s \frac{R}{\lambda} \quad (9.11.58)$$

$$a_{34} = \frac{\partial g_3}{\partial i_2} \Big|_{\vec{x}_0} + \frac{\partial h_3}{\partial i_2} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.59)$$

$$a_{41} = \frac{\partial g_4}{\partial y} \Big|_{\vec{x}_0} + \frac{\partial h_4}{\partial y} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.60)$$

$$a_{42} = \frac{\partial g_4}{\partial v} \Big|_{\vec{x}_0} + \frac{\partial h_4}{\partial v} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.61)$$

$$a_{43} = \frac{\partial g_4}{\partial i_1} \Big|_{\vec{x}_0} + \frac{\partial h_4}{\partial i_1} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.62)$$

$$a_{44} = \frac{\partial g_4}{\partial i_2} \Big|_{\vec{x}_0} + \frac{\partial h_4}{\partial i_2} \Big|_{\vec{x}_0, \vec{u}_0} = -s \frac{R}{\lambda} \quad (9.11.63)$$

$$h_{11} = h_{21} = 0 \quad (9.11.64)$$

$$h_{12} = h_{22} = 0 \quad (9.11.65)$$

$$h_{31} = \frac{\partial h_3}{\partial u_1} \Big|_{\vec{x}_0, \vec{u}_0} = \frac{s}{\lambda} \quad (9.11.66)$$

$$h_{32} = \frac{\partial h_3}{\partial u_2} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.67)$$

$$h_{41} = \frac{\partial h_4}{\partial i_1} \Big|_{\vec{x}_0, \vec{u}_0} = 0 \quad (9.11.68)$$

$$h_{42} = \frac{\partial h_4}{\partial i_2} \Big|_{\vec{x}_0, \vec{u}_0} = \frac{s}{\lambda} \quad (9.11.69)$$

Writing in matrix form, we obtain the following matrices

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{2g}{s} & -\frac{\sigma}{m} & \frac{2}{s}\sqrt{\frac{\lambda g}{m}} & 0 \\ R(1-R)\sqrt{\frac{\lambda g}{m}} & -\sqrt{\frac{mg}{\lambda}} & -s\frac{R}{\lambda} & 0 \\ 0 & 0 & 0 & -s\frac{R}{\lambda} \end{bmatrix} \quad (9.11.70)$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{s}{\lambda} & 0 \\ 0 & \frac{s}{\lambda} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.11.71)$$

which gives the matrix representation of the linearized physical-model of the magnetic bearing.

The linearized system responds to the following dynamic

$$\Delta \dot{\vec{x}} = \mathbf{A} \Delta \vec{x} + \mathbf{B} \Delta \vec{u} \quad (9.11.72)$$

$$\Delta \vec{y} = \mathbf{C} \Delta \vec{x} \quad (9.11.73)$$

where

$$\Delta \vec{x} = \vec{x} - \vec{x}_0 = \begin{bmatrix} y \\ v \\ i_1 - i_{10} \\ i_2 \end{bmatrix}, \quad \Delta \vec{u} = \vec{u} - \vec{u}_0 = \begin{bmatrix} 0 \\ 0 \\ u_1 - u_{10} \\ u_2 \end{bmatrix} \quad (9.11.74)$$

The voltage inputs  $u_1$  and  $u_2$  can be assumed mutually exclusive<sup>¶¶</sup>, then, the vector input  $\vec{u}$  can be written in a way to transform the model from a multi-input to a single-input system, as follows

$$u_1 = \begin{cases} u & \text{if } u \geq 0 \\ 0 & \text{if } u \leq 0 \end{cases} \quad u_2 = \begin{cases} 0 & \text{if } u \geq 0 \\ -u & \text{if } u \leq 0 \end{cases} \quad (9.11.75)$$

By Eq. (9.11.75) the voltage input  $\vec{u}$  can be written as function of a single voltage input  $u$ .

---

<sup>¶¶</sup>when  $u_1 > 0 \Rightarrow u_2 = 0$ , and when  $u_2 > 0 \Rightarrow u_1 = 0$

A generic representation of a possible control structure based on the linearized magnetic bearing model is shown in Figure 9.57.

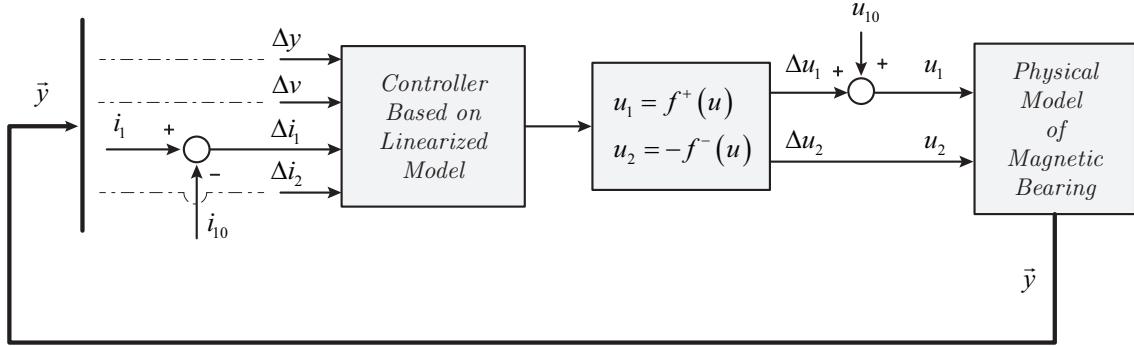


Figure 9.57: Functional graph for the implementation of a control based on the linearized model.

#### 9.11.4 State-Feedback with Integrator applied to the linearized magnetic bearing model.

In this section the design of the full state feedback with integrator control applied to the magnetic bearing is reported.

**The design of the state feedback with integrator, for the linear magnetic bearing model, reduces to the following pole-placement problem:**

Find

$$\mathbf{K}_e = \begin{bmatrix} \begin{bmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \end{bmatrix} & \begin{bmatrix} -k_i \\ 0 \end{bmatrix} \end{bmatrix} \quad (9.11.76)$$

such that

$$\left| s\mathbf{I} - (\mathbf{A}_e - \mathbf{B}_e \mathbf{K}_e) \right| = \sum_{k=1}^n (s - \mu_k) \quad (9.11.77)$$

is satisfied, and where

$$\mathbf{A}_e = \begin{bmatrix} \mathbf{A} & \vec{0} \\ -\vec{C}_1 & 0 \end{bmatrix}, \quad \mathbf{B}_e = \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix} \quad (9.11.78)$$

$\mu_k$ ,  $i = 1, \dots, n$ , are the desired poles.

Where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{2g}{s} & -\frac{\sigma}{m} & \frac{2}{s}\sqrt{\frac{\lambda g}{m}} & 0 \\ R(1-R)\sqrt{\frac{\lambda g}{m}} & -\sqrt{\frac{mg}{\lambda}} & -s\frac{R}{\lambda} & 0 \\ 0 & 0 & 0 & -s\frac{R}{\lambda} \end{bmatrix} \quad (9.11.79)$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{s}{\lambda} & 0 \\ 0 & \frac{s}{\lambda} \end{bmatrix}, \quad \vec{C}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \quad (9.11.80)$$

The use of the vector  $\vec{C}_1$  follows from the fact that the integral compensation acts on the only state variable  $y(t)$  of the state vector  $\vec{x}(t)$ .

Figure 9.58 shows the layout of the proposed control system, where Eqs. (9.11.75) are replaced by the functions

$$f^+(u) = \frac{1}{2}(u + |u|) \quad (9.11.81)$$

$$f^-(u) = -\frac{1}{2}(u - |u|) \quad (9.11.82)$$

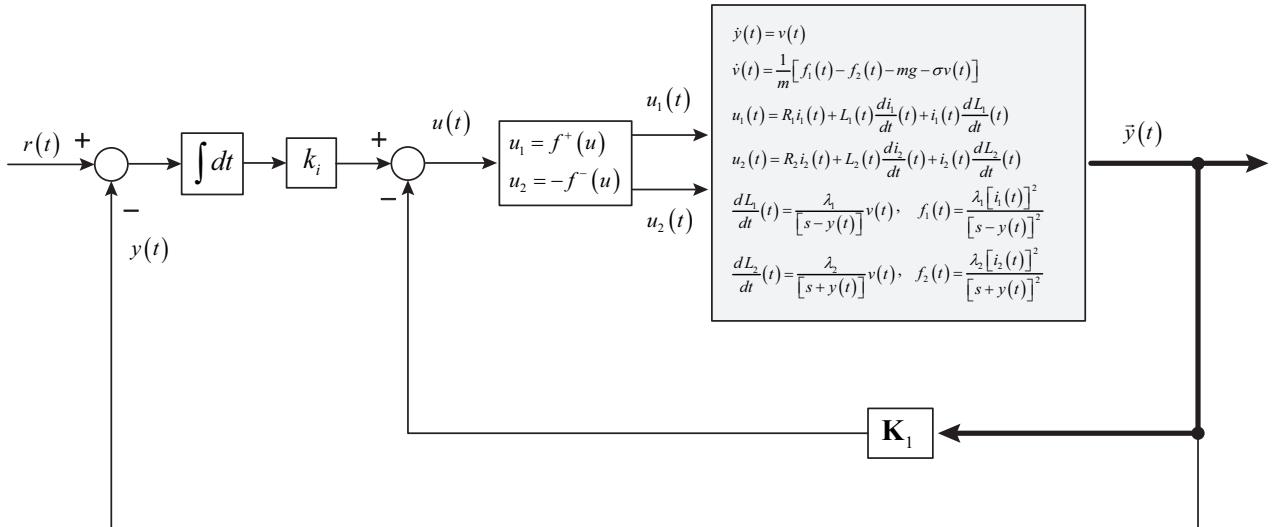


Figure 9.58: Magnetic bearing control architecture.

### 9.11.5 Simulation Results

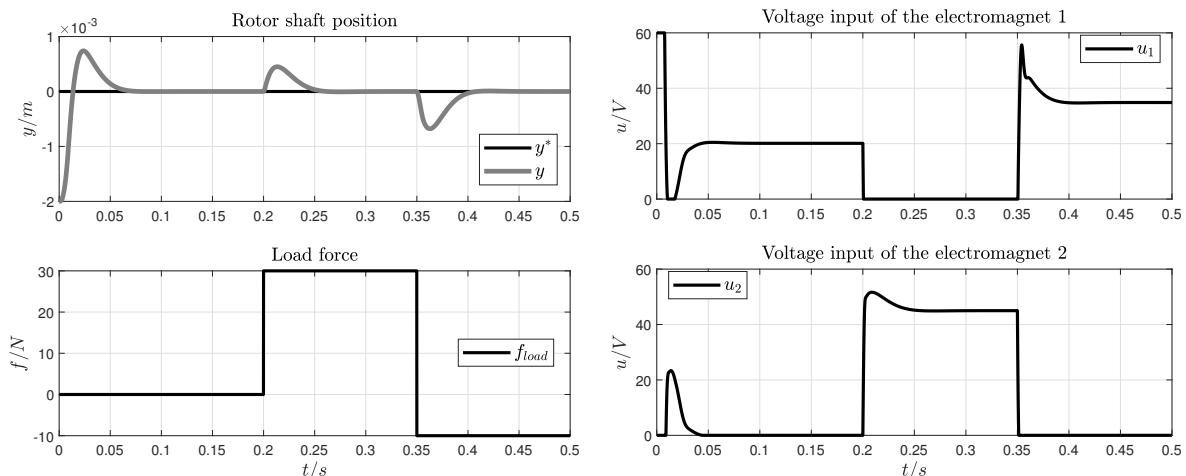
In this section simulation results are shown. The data of the plant are listed in the Table 9.5

Parameter	Value
$\lambda_1$	$1 \times 10^{-4} \text{ H m}$
$\lambda_2$	$1 \times 10^{-4} \text{ H m}$
$R_1$	$10 \Omega$
$R_2$	$10 \Omega$
$s$	$4.5 \times 10^{-3} \text{ m}$
$y_{\max}$	$2 \times 10^{-3} \text{ m}$
$m$	$0.5 \text{ kg}$
$\sigma$	$1 \text{ N s m}^{-1}$
$\sigma_{rw}$	$1 \times 10^6 \text{ N s m}^{-1}$
$g$	$10 \text{ m s}^{-2}$

Table 9.5: Magnetic bearing data.

The simulation has been performed in the following way:

- the initial position of the shaft is taken equal to the maximum negative displacement of the air-gap;
- during the control of the position, a load curve force is applied to the shaft bearing.



(a) Rotor shaft position (upper picture) and external load force applied to the shaft (bottom picture).

(b) Coil 1 voltage input  $u_1(t)$  (upper picture) and coil 2 voltage input  $u_2(t)$  (bottom picture).

Figure 9.59: Simulation results - case study of linear control based on the linearized model of the magnetic bearing.

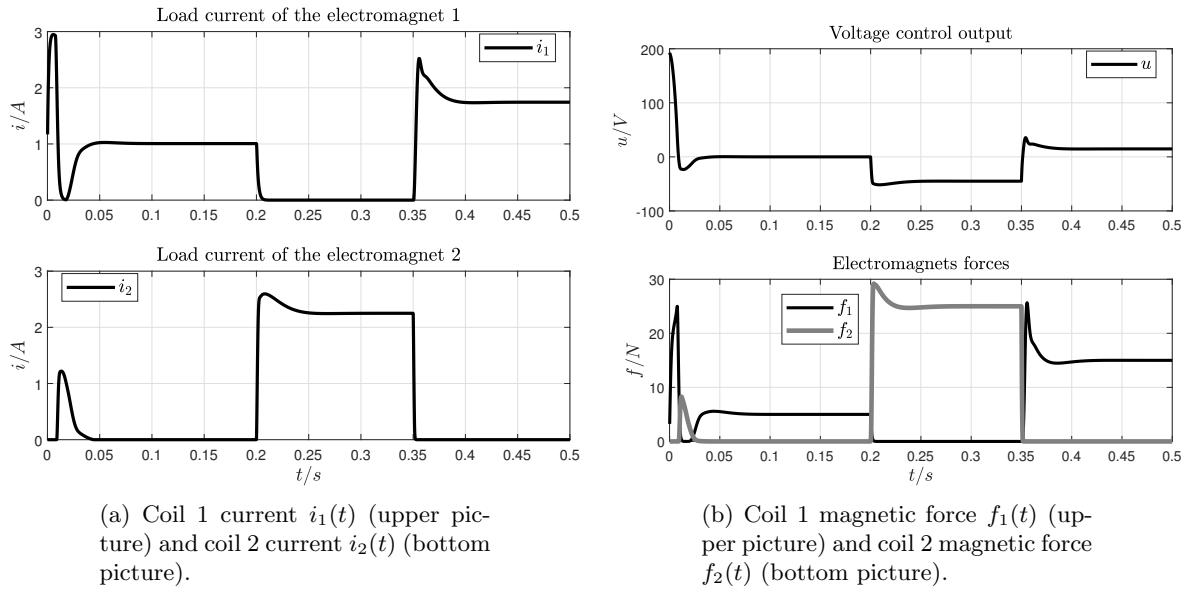


Figure 9.60: Simulation results - case study of linear control based on the linearized model of the magnetic bearing.

# Chapter 10

## Appendices

### 10.1 Moving Reference Frames

In our study of electromechanical devices we will need to determine the voltage induced in a coil as it moves through a stationary magnetic field. This requires an understanding of quasi-static field theory as observed in two different inertial reference frames (in motion relative to one another). For our purposes, an inertial reference frame is one that moves with constant velocity. In this section, we derive the transformation laws for quasi-static fields as observed in two different inertial reference frames.

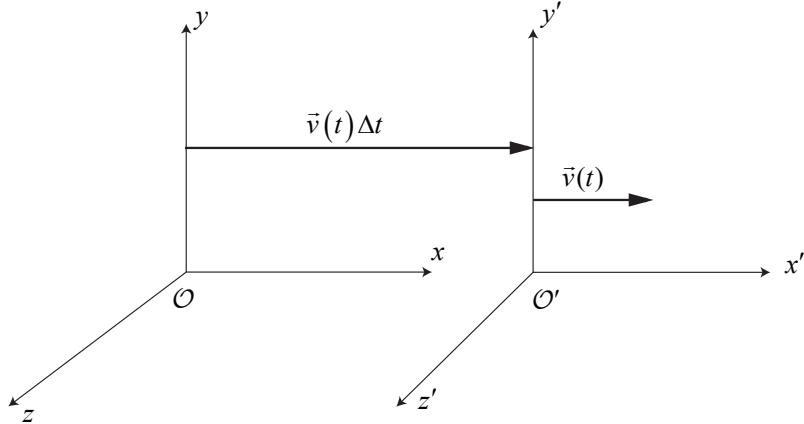


Figure 10.1: Inertial reference frames in relative motion along one axis.

Consider two inertial reference frames, a stationary “laboratory” reference frame  $\mathcal{O}$  with coordinates  $(x, y, z, t)$  and a moving reference frame  $\mathcal{O}'$  with coordinates  $(x', y', z', t')$ . Frame  $\mathcal{O}'$  moves with velocity  $\vec{v}$  with respect to  $\mathcal{O}$  as shown in Figure 10.1. Observers in  $\mathcal{O}$  and  $\mathcal{O}'$  measure different values for the electromagnetic fields. Rigorous comparison of these measurements can be made if one knows how fields transform from  $\mathcal{O}$  to  $\mathcal{O}'$  or vice versa. The transformation laws follow from the theory of special relativity.

The basic postulates of special relativity are as follows:

1. The law of physics are the same in all inertial reference frames.

2. Observer in inertial reference frames measure the same value for the speed of light.

The first postulate implies that the quasi-static field equations must have the same form in both  $\mathcal{O}$  and  $\mathcal{O}'$  as shown in the following equations

### Quasi-Static Equations

#### Reference frame $\mathcal{O}$

$$\begin{aligned}\vec{\nabla} \times \vec{H} &= \vec{J} \\ \vec{\nabla} \cdot \vec{B} &= 0 \\ \vec{\nabla} \times \vec{E} &= -\frac{\partial \vec{B}}{\partial t} \\ \vec{\nabla} \cdot \vec{J} &= 0\end{aligned}\tag{10.1.1}$$

#### Reference frame $\mathcal{O}'$

$$\begin{aligned}\vec{\nabla}' \times \vec{H}' &= \vec{J}' \\ \vec{\nabla}' \cdot \vec{B}' &= 0 \\ \vec{\nabla}' \times \vec{E}' &= -\frac{\partial \vec{B}'}{\partial t} \\ \vec{\nabla}' \cdot \vec{J}' &= 0\end{aligned}\tag{10.1.2}$$

Here  $\vec{\nabla}'$  is the gradient operator with respect to the  $(x', y', z', t')$  coordinate

$$\vec{\nabla}' = \frac{\partial}{\partial x'} \vec{i}' + \frac{\partial}{\partial y'} \vec{j}' + \frac{\partial}{\partial z'} \vec{k}'$$

our goal is to determine the transformation laws for the field going from  $\mathcal{O}$  to  $\mathcal{O}'$  and vice versa. We can determine these if we know the transformation relations for the coordinate variables. These follow from the second postulate of the special relativity and are given by

$$\begin{aligned}t' &= \gamma(t - \beta x/c) \\ x' &= \gamma(x - \beta ct) \\ y' &= y \\ z' &= z\end{aligned}\tag{10.1.3}$$

where

$$\gamma = \frac{1}{\sqrt{1 + \beta^2}}$$

and

$$\beta = \frac{v}{c} -$$

Here,  $c$  is the speed of the light in a vacuum. We are interested in low-velocity electromechanical devices for which  $u \ll c$ . Therefore

$$\beta \approx 0$$

and

$$\gamma \approx 1.$$

Substituting the above approximations Eq. 10.1.3 reduce to the Galilean transform relation

$$\begin{aligned} t' &= t \\ x' &= x - vt \\ y' &= y \\ z' &= z \end{aligned} \tag{10.1.4}$$

This analysis generalized to the case of an arbitrary velocity as shown in Figure 10.2.

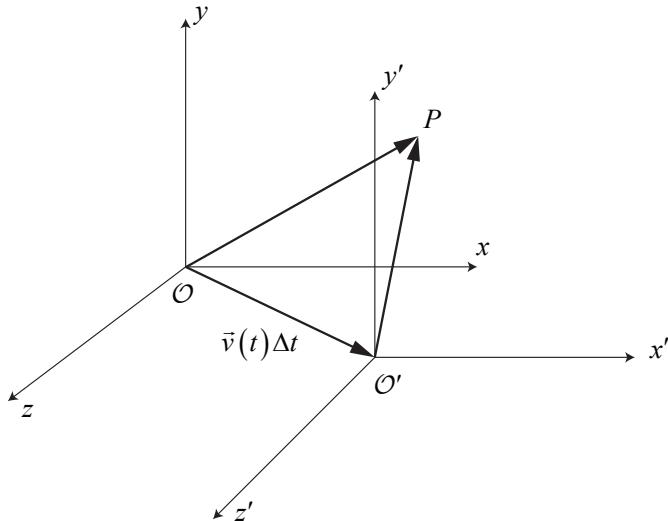


Figure 10.2: Inertial reference frames in relative motion.

For this case we find that

$$\begin{aligned} t' &= t \\ x' &= x - v_x t \\ y' &= y - v_y t \\ z' &= z - v_z t \end{aligned} \tag{10.1.5}$$

Now that we know the transform relations for the coordinates, we can determine the corresponding relations for the field equations. These will ultimately enable us to compare the fields in the two reference frames. We start by considering a scalar-valued function  $f'$  of the **primed** coordinates,  $f'(x', y', z')$ . By virtue of Eq. 10.1.5,  $f'$  can be considered to be a function of the **unprimed** coordinates,

$$f'(x', y', z') = f'(x'(x, t), y'(y, t), z'(z, t)),$$

We use the chain rule to determine the derivatives of  $f'$  with respect to the **unprimed** coordinates. We consider spatial derivatives first and find that

$$\frac{\partial f'}{\partial x} = \frac{\partial f'}{\partial x'} \frac{\partial x'}{\partial x} = \frac{\partial f'}{\partial x'}.$$

It follows that

$$\vec{\nabla} f' = \vec{\nabla}' f'.$$

Next, we consider the time derivative. From the chain rule we have

$$\begin{aligned} \frac{\partial f'}{\partial t} &= \frac{\partial f'}{\partial t'} \frac{\partial t'}{\partial t} + \frac{\partial f'}{\partial x'} \frac{\partial x'}{\partial t} + \frac{\partial f'}{\partial y'} \frac{\partial y'}{\partial t} + \frac{\partial f'}{\partial z'} \frac{\partial z'}{\partial t} = \\ &= \frac{\partial f'}{\partial t'} - \frac{\partial f'}{\partial x'} v_x - \frac{\partial f'}{\partial y'} v_y - \frac{\partial f'}{\partial z'} v_z \end{aligned} \quad (10.1.6)$$

where

$$\begin{aligned} \frac{\partial t'}{\partial t} &= 1; & \frac{\partial x'}{\partial t} &= -v_x; \\ \frac{\partial y'}{\partial t} &= -v_y; & \frac{\partial z'}{\partial t} &= -v_z; \end{aligned} \quad (10.1.7)$$

Eq. 10.1.6 can be rewritten as

$$\frac{\partial f'}{\partial t} = \frac{\partial f'}{\partial t'} - \vec{v} \cdot (\vec{\nabla}' f') \quad (10.1.8)$$

or

$$\frac{\partial f'}{\partial t'} = \frac{\partial f'}{\partial t} + \vec{v} \cdot (\vec{\nabla} f') \quad (10.1.9)$$

The relations 10.1.8 and 10.1.9 relate the derivatives in  $\mathcal{O}$  and in  $\mathcal{O}'$  for scalar-valued function. A similar analysis applies to vector-valued function. Let  $\vec{A}'(x', y', z', t')$  be a vector-valued function of the **primed** coordinate variables.

This can also be considered to be a vector-valued function of the **unprimed** coordinates

$$\vec{A}'(x', y', z', t') = \vec{A}'(x'(x, t), y'(y, t), z'(z, t))$$

It is easy to check that

$$\vec{\nabla}' \cdot \vec{A}' = \vec{\nabla} \cdot \vec{A}' \quad (10.1.10)$$

and that

$$\vec{\nabla}' \times \vec{A}' = \vec{\nabla} \times \vec{A}'. \quad (10.1.11)$$

Similarly, by taking the time derivative we find that

$$\frac{\partial \vec{A}'}{\partial t'} = \frac{\partial \vec{A}'}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{A}'. \quad (10.1.12)$$

Suppose that the **unprimed** frame is the fixed or laboratory frame. Then, from the left-hand side of Eq. 10.1.12 it is clear that the right-hand side is *the rate of change with respect to time of  $\vec{A}'$  for an observer moving with velocity  $\vec{v}$* . This derivative, written in terms of the coordinates  $(x, y, z, t)$  of the fixed reference frame, is called *substantial derivative*

$$\boxed{\frac{D\vec{A}'}{Dt'} = \frac{\partial \vec{A}'}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{A}'.} \quad (\text{respect to } \mathcal{O}') \quad (10.1.13)$$

Eq. 10.1.12 can be written in different form. Because  $\vec{v}$  is constant and by the identity

$$\vec{\nabla} \times (\vec{A} \times \vec{B}) = (\vec{B} \cdot \vec{\nabla}) \vec{A} - (\vec{A} \cdot \vec{\nabla}) \vec{B} + \vec{A}(\vec{\nabla} \cdot \vec{B}) - \vec{B}(\vec{\nabla} \cdot \vec{A}) \quad (10.1.14)$$

makes it possible to write Eq. 10.1.12 in the form

$$\frac{\partial \vec{A}'}{\partial t'} = \frac{\partial \vec{A}'}{\partial t} + \vec{v}(\vec{\nabla} \cdot \vec{A}') - \vec{\nabla} \times (\vec{v} \times \vec{A}'), \quad (10.1.15)$$

Finally, substitute Eqs. 10.1.10, 10.1.11 and 10.1.15 into the quasi-static equations for  $\mathcal{O}'$  in 10.1.2 we obtain

$$\vec{\nabla}' \times \vec{H}' = \vec{J}' \Rightarrow \vec{\nabla} \times \vec{H}' = \vec{J}' \quad (\text{respect to } \mathcal{O}') \quad (10.1.16)$$

$$\vec{\nabla}' \cdot \vec{B}' \Rightarrow \vec{\nabla} \cdot \vec{B}' = 0 \quad (\text{respect to } \mathcal{O}') \quad (10.1.17)$$

$$\vec{\nabla}' \times \vec{E}' = -\frac{\partial \vec{B}'}{\partial t'} \Rightarrow \vec{\nabla} \times (\vec{E}' - \vec{v} \times \vec{B}') = -\frac{\partial \vec{B}'}{\partial t} \quad (\text{respect to } \mathcal{O}') \quad (10.1.18)$$

$$\vec{\nabla}' \cdot \vec{J}' \Rightarrow \vec{\nabla} \cdot \vec{J}' = 0 \quad (\text{respect to } \mathcal{O}') \quad (10.1.19)$$

We have used  $\vec{\nabla} \cdot \vec{B}' = 0$  in the equation 10.1.18.

It has been postulated that equations 10.1.16-10.1.19 describe the same physical laws as

$$\vec{\nabla} \times \vec{H} = \vec{J} \quad (\text{respect to } \mathcal{O}) \quad (10.1.20)$$

$$\vec{\nabla} \cdot \vec{B} = 0 \quad (\text{respect to } \mathcal{O}) \quad (10.1.21)$$

$$\vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \quad (\text{respect to } \mathcal{O}) \quad (10.1.22)$$

$$\vec{\nabla} \cdot \vec{J} = 0 \quad (\text{respect to } \mathcal{O}) \quad (10.1.23)$$

A comparison of Eqs. 10.1.16-10.1.19 and Eq. 10.1.20-10.1.23 shows that a consistent set of

transformations which satisfies this requirement is:

### Transformation Relations

$$\vec{H} = \vec{H}' \quad (10.1.24)$$

$$\vec{B} = \vec{B}' \quad (10.1.25)$$

$$\vec{E} = \vec{E}' - \vec{v} \times \vec{B} \quad (10.1.26)$$

$$\vec{J} = \vec{J}' \quad (10.1.27)$$

In Eqs. 10.1.24 - 10.1.27 the variables  $(\vec{H}, \vec{B}, \vec{E}, \vec{v}, \vec{J})$  are measured in the stationary reference frame  $\mathcal{O}$  and the variables  $(\vec{H}', \vec{B}', \vec{E}', \vec{J}')$  are measured in the moving frame  $\mathcal{O}'$ . For these relations to hold, the variables must be compared **at the same physical point in space**. For example, from Eqs. 10.1.5 and 10.1.26 we have

$$\begin{aligned} \vec{E}'(x', y', z', t') &= \vec{E}(x - v_x t, y - v_y t, z - v_z t, t) \\ &\quad + \vec{v} \times \vec{B}(x - v_x t, y - v_y t, z - v_z t, t). \end{aligned}$$

Notice that we have assumed that  $\vec{v}$  is a constant (independent of position).

The relations 10.1.24 - 10.1.27 give the transformations for the fields and the source. However, we still need to know the constitutive relations in  $\mathcal{O}$  and  $\mathcal{O}'$ . We apply a general principle that says that the constitutive relations for moving media are the same as for stationary media when they are written in term of the fields defined in an inertial reference frame moving with the media. For example for a linear, homogeneous and isotropic media at rest in  $\mathcal{O}'$  we have  $\vec{J}' = \sigma \vec{E}'$  and  $\vec{B}' = \mu \vec{H}'$ . However, in  $\mathcal{O}$  (which is moving with respect to the media) the relations are modified by virtue of Eqs. 10.1.24 - 10.1.27. Specifically, we find that

### Constitutive Relations

#### Reference frame $\mathcal{O}$

$$\vec{J} = \sigma(\vec{E} + \vec{v} \times \vec{B})$$

$$\vec{B} = \mu \vec{H}$$

#### Reference frame $\mathcal{O}'$

$$\begin{aligned} \vec{J}' &= \sigma \vec{E}' \\ \vec{B}' &= \mu \vec{H}' \end{aligned} \quad (10.1.28)$$

## 10.2 Propaedeutic Fundamentals for Magnetic Bearings

As illustrated in Figure 10.3, active magnetic bearing are created by combining electromagnets, power supplies, position sensors, and a control system. The control system uses the signal from the position sensor and determine the voltage or the current which must be impressed to the coil. The resulting coil current generates a force which acts on the suspended rotor.

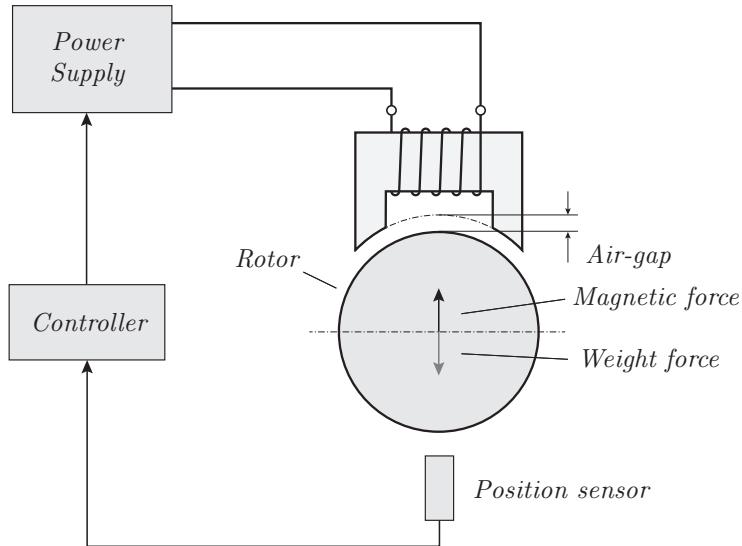


Figure 10.3: Basic magnetic bearing structure, showing the fundamental components.

In the following sections, some of fundamental laws of the magnetic systems are depicted.

### 10.2.1 Magnetic Field and Induction

The magnetic field in a space is specified by mechanical forces and electrical induction. Both effects can be used to define a measure of the intensity of the magnetic field. In a stationary magnetic field the *Lorentz force* acts perpendicular to the velocity of a charge  $q$ . The magnetic field vector  $\vec{B}$  (magnetic induction or flux density) is perpendicular to the force  $\vec{f}$  and speed  $\vec{v}$ :

$$\vec{f} = q(\vec{v} \times \vec{B}) \quad (10.2.1)$$

This vector product means that the force is determined only by the component  $\vec{B}_w$  of  $\vec{B}$  which is perpendicular to the velocity  $\vec{v}$  (Figure 10.4).

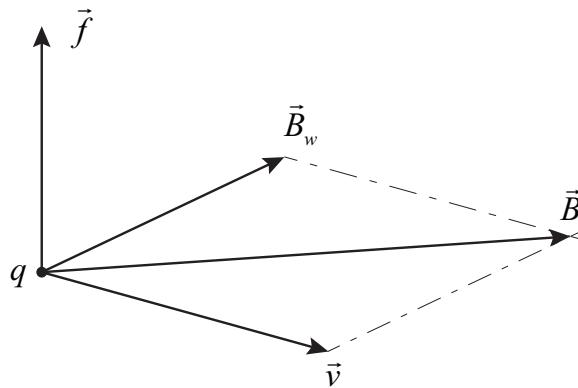


Figure 10.4: Lorentz force.

The measuring unit of the magnetic flux density is Tesla [ $T = N A^{-1} m^{-1}$ ], and may be defined as the flux density of magnetic field where a force of 1 N acts on a conductor with a current of 1 A and a length of 1 m. The conductor is perpendicular to the flux.

The magnetic flux can be visualized by magnetic field lines. The density of these lines indicates the flux density modulus, and the direction of the lines indicates the direction of the field vector of the flux density. Each field line is always closed. The total magnetic flux  $\phi$  passing through a surface  $A$  is the integral of the flux density  $\vec{B}$  over the surface:

$$\phi = \iint_A \vec{B} \cdot d\vec{A} \quad (10.2.2)$$

A magnetic field can be generated by moving charges, alternating electric fields, and permanent magnet. A rotation-symmetrical magnetic field is generated around a straight conductor with a constant current  $\vec{i}$ . The magnetic field  $\vec{H}$  is inversely proportional to the distance  $r$  from the conductor, and its direction is tangential to concentric circles around the conductor (Figure 10.5a)

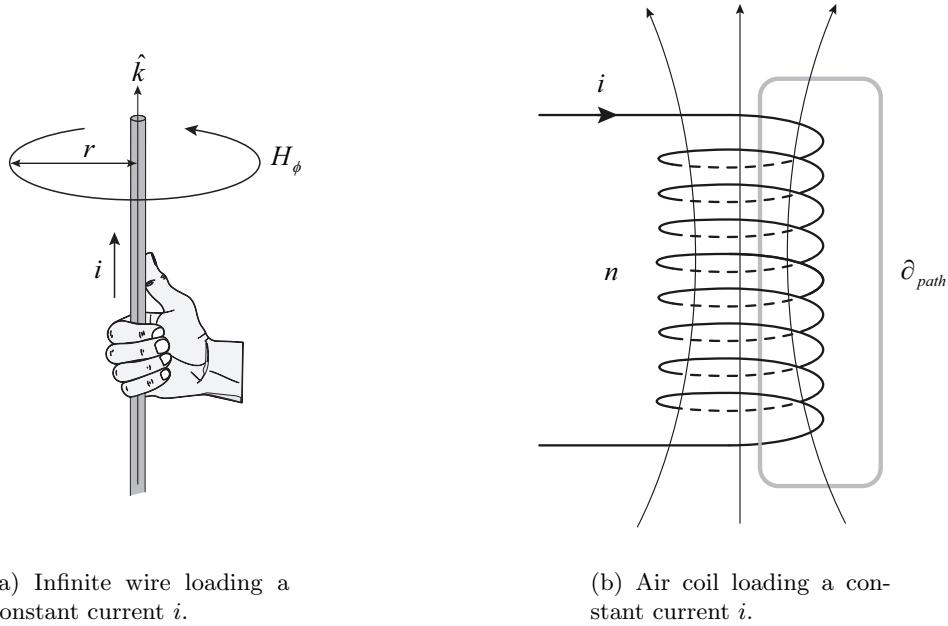


Figure 10.5: Magnetic fields around conductors.

Here, the magnetic field intensity is determined by the current density independently of the medium. A contour integral has either a fixed value or else it vanishes, depending on whether the path leads around the conductor or not

$$\oint \vec{H} \cdot d\vec{s} = i \quad (10.2.3)$$

The magnitude of the magnetic field vector in the case of Figure 10.5a is therefore

$$H = |\vec{H}| = \frac{i}{2\pi r} \quad (10.2.4)$$

If the integration path ( $\partial_{path}$ ) encompasses several current loops, as is the case with the air coil in Figure 10.5b, then the integral of the current density  $\vec{J}$  through the surface  $A$  enclosed by the integration path  $\partial_{path}$  yields  $i_{en}$ , the enclosed current,

$$\int_{\partial_{path}} \vec{H} \cdot d\vec{s} = \iint_A \vec{J} \cdot d\vec{a} = ni = i_{en} \quad (10.2.5)$$

The formulation of Eq. (10.2.5) is called the Ampere's circuital law. The magnetic field  $\vec{H}$  and the magnetic induction (flux density)  $\vec{B}$  are linked by the *constitutive law*

$$\vec{B} = \mu_0 \mu_r \vec{H} \quad (10.2.6)$$

where  $\mu_r$  represents a tensor. For given geometries and isotropic media, Eq. (10.2.6) reduces to

$$B = \mu_0 \mu_r H \quad (10.2.7)$$

The term  $\mu_0 = 4\pi \times 10^{-7} \text{ H m}^{-1}$  stands for the *magnetic permeability* of the vacuum. The *relative permeability*  $\mu_r$  depends on the medium upon which the magnetic field acts. For the vacuum,  $\mu_r$  equals 1 and is also approximately unity in air. The SI unit of the magnetic field  $\vec{H}$  is  $[\text{A m}^{-1}]$ .

When a conductor is moved in a magnetic field, all of its charged particles are moved, too. According to Eq. (10.2.1), a Lorentz force acts on these particles, and they move perpendicularly to the field and the direction of the conductor. The motion of the charged particles in the conductor corresponds to the electric current.

A current is always caused by an electric field. The electric field generating the induction current is produced by a change in the magnetic flux which passes through the surface surrounded by a conductor as in Figure 10.6. The resulting potential difference is called *induction voltage*.

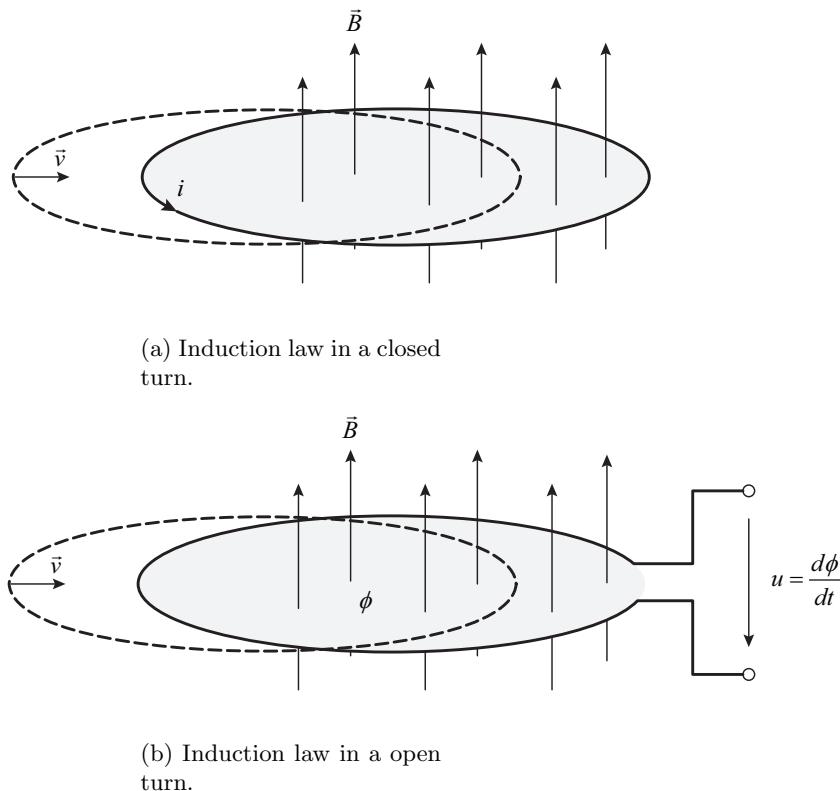


Figure 10.6: Induction law in a turn..

The voltage  $u$  induced in a coil with  $n$  windings equals the product of the winding number and the derivative of the flux with respect to time  $t$  (induction law):

$$u = n \frac{d\phi}{dt} = \frac{d\psi}{dt} \quad (10.2.8)$$

where  $\psi$  represents the flux linkage to the coil.

### 10.2.2 Properties of Ferromagnetic Material

When a magnetic field with a density  $\vec{H}$  [ $\text{A m}^{-1}$ ] acts on a material, the magnetic flux density  $\vec{B}$  [T] generated will be either higher or lower than the flux density  $\mu_0 \vec{H}$  generated in the vacuum, depending on material properties. The part of  $\vec{B}$  originating from the material itself is called magnetic polarization  $\vec{M}$ :

$$\vec{B} = \mu_0 \vec{H} + \vec{M} \quad (10.2.9)$$

Comparing (10.2.9) with  $\vec{B} = \mu_0 \mu_r \vec{H}$  (for isotropic media) yields

$$\vec{M} = (\mu_r - 1) \mu_0 \vec{H} = \chi_m \mu_0 \vec{H} \quad (10.2.10)$$

in which  $\chi_m = \mu_r - 1$  is called the *magnetic susceptibility*. This describes the relationship between the magnetic polarization and the flux density of the vacuum.

Material with  $\chi_m < 0$  ( $\mu_r < 1$ ) are called *diamagnetic*. They reduce the flux density. Materials where  $\chi_m > 0$  ( $\mu_r > 1$ ) are called *paramagnetic*. In some paramagnetic materials, coupling of the resulting atomic magnetic moments can occur. When they are parallel, the material is called *ferromagnetic*. In this kind of material  $\mu_r$  is generally  $\gg 1$  and depends both on the size of the magnetic field and the magnetic “history” of the material.

In general, the parallel arrangement of the atomic magnetic dipoles applies only to a limited space, referred to as *Weiss' domains*. The transition zones between these domains, where the atomic moments change from one privileged direction to another, are called *Bloch walls*.

The behaviour of magnetic material is usually visualized in a  $B - H$  diagram, see Figure 10.7. Thus, when an unmagnetized ferromagnetic sample is placed within a homogenous magnetic field with a value  $H$  of increasing intensity, the magnitude of the flux density  $B$  increases rapidly along the new curve in the first quadrant, due to Bloch wall shifts. Meanwhile, the domains having their privileged direction parallel to the field direction expand - at the expense of the others.

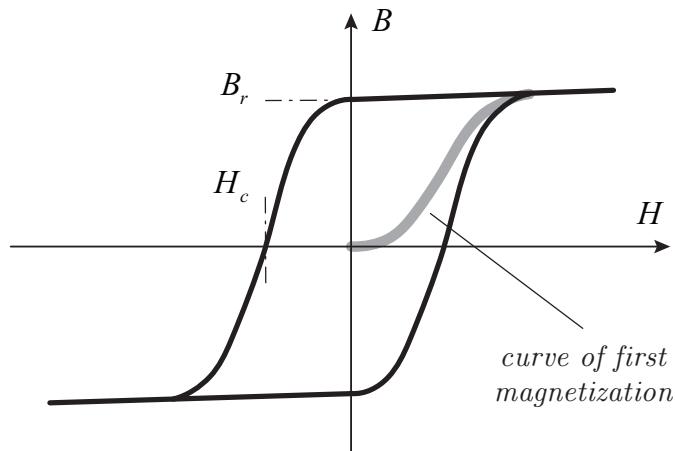


Figure 10.7:  $B - H$  diagram, hysteresis loop.

If  $H$  keeps increasing, flux density increases, but slowly. Now, the so-called turning processes take place, where the dipoles of the Weiss's domains that are left after the wall shifts change from their privileged direction in the field direction. Once all magnetic dipoles are parallel to the outer magnetic field, saturation is achieved, and  $B$  only increase with slope  $\mu_0$ .

When the outer field is reduced to  $H = 0$  flux density does not run reversibly along the original curve, but irreversibly along a *hysteresis loop*. Before reaching value  $H = 0$ , only part of the turning processes are reversed. The remaining flux density is referred to  $B_r$ . If  $H$  is increased in the opposite direction, further turning processes will take place first. Then, remagnetization through wall shifts occurs, and  $B$  drops. The field intensity necessary to attain  $B = 0$  is called *coercive field* intensity  $H_c$ . With increasing intensity of the counter field the sample is magnetized until saturation in the opposite direction is achieved. By resetting the field to zero and by increasing it subsequently in the original direction, saturation is achieved again, and the hysteresis loop has thus been run through once.

### 10.2.3 Magnetic Circuit

In the magnetic bearing technology, electromagnets or permanent magnets cause the flux to circulate in a magnetic loop. When analysing such magnetic loops, an exact theoretical computation of the field is rarely possible and seldom required. One usually works with analytic methods of approximation, based on the simplifying assumption that the flux, except for in the air gap, runs entirely through the iron (no leakage flux). Since the permeability  $\mu = \mu_0\mu_r$  of iron is considerably larger than of air, the magnetic field lines leave the iron almost perpendicularly to its surface. Both for constant and alternating fields the computational methods used for the static fields are applied, which is admissible as long as the alternating fields have a very large wavelength, compared with the geometry of the field.

For the computation of flux density  $B$ , the following simplifying assumptions are made: flux  $\phi$  is located entirely within the magnetic loop with iron cross section  $A_{fe}$  which is assumed to be constant along the entire loop equal to cross-section  $A_a$  in the air gap. From

$$\phi = B_{fe}A_{fe} = B_aA_a \quad (10.2.11)$$

and

$$A_{fe} = A_a \quad (10.2.12)$$

follows

$$B_{fe} = B_a = B \quad (10.2.13)$$

The field within the magnetic loop is assumed to be homogeneous both in the iron and in the air gap. Therefore, we base our calculation on a mean length  $l_{fe}$  of the magnetic path and air gap length of  $2s$ .

For the magnetic circuit of Figure 10.8 is follows

$$\oint \vec{H} \cdot d\vec{s} = l_{fe} H_{fe} + 2sH_a = ni \quad (10.2.14)$$

The term  $ni$  in (10.2.14) is called the *magnetomotive force* (mmf)

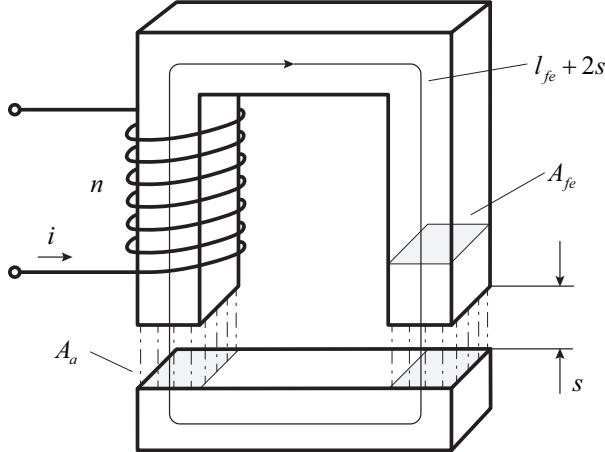


Figure 10.8: Magnetic circuit.

that “forces” flux through a magnetic circuit. Since according to Eq. (10.2.13), the flux density  $B$  in the iron and in the air gap is identical, field intensity  $H_{fe}$  and  $H_a$  from Eq. (10.2.14) can be replaced by relation  $B = \mu_0 \mu_r H$ , which follows

$$l_{fe} \frac{B}{\mu_0 \mu_r} + 2s \frac{B}{\mu_0} = ni \quad (10.2.15)$$

which yields to

$$B = \mu_0 \frac{ni}{l_{fe}/\mu_r + 2s} \quad (10.2.16)$$

In the iron,  $\mu_r \gg 1$ . so the magnetization of the iron is often neglected. In this case, Eq. (10.2.16) results as follows

$$B = \mu_0 \frac{ni}{2s} \quad (10.2.17)$$

Equation (10.2.16) is a good approximation as long as the iron is kept far below the saturation flux density, since relative permeability has but little effect on the steep rise of the magnetization curve. However, if the iron is kept close to the saturation with high flux densities, we must consider the characteristics of the magnetization curve, and the flux density  $B$  can no longer be calculated directly from the enclosed current  $i_{en} = ni$ .

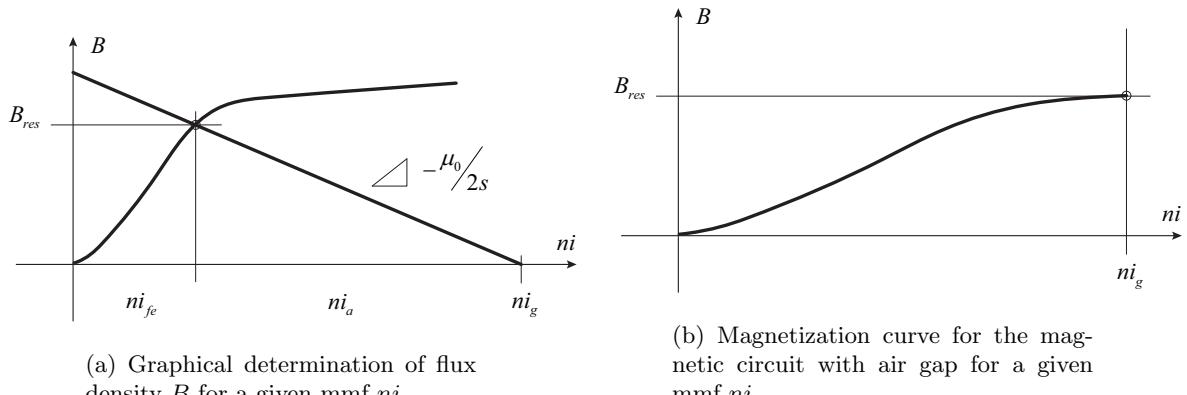


Figure 10.9: Graphical determination of the magnetization.

In the simple case of a magnetic circuit with a constant cross section, the flux can be determined graphically using the magnetization curve. To do this, the first step is to scale the  $H$  axis of the  $B - H$  diagram using the relation  $H_{fe} = ni$  so that the graph shows the flux density as a function of the magnetomotive force  $ni$ . In the next step, intersect the magnetization curve of the iron with the air gap curve (straight line with slope  $-\mu_0/(2s)$ ) drawn from the given mmf  $ni_g$  to the left, Figure 10.9a. The intersection point shows the resulting flux density  $B_{res}$ . Figure 10.9a shows two portions of the given mmf, a portion  $ni_{fe}$  needed to “force” flux in the iron and a portion  $ni_a$  needed to “force” flux in the air gap. With flux density  $B_{res}$  a magnetization curve for the magnetic circuit with air gap can be drawn, Figure 10.9b.

Inductance  $L$  is the ratio of the linkage flux  $n\phi$  (for a coil with  $n$  turns) to the generating current  $i$ :

$$L = \frac{n\phi}{i} = \frac{\psi}{i} \quad (10.2.18)$$

If the iron is neglected, the flux density  $B$  from (10.2.17) and the cross section of the air gap  $A_a$  can be inserted in (10.2.18), and the inductance  $L$  of a magnetic circuit can thus calculated approximately:

$$L = \frac{\mu_0 n^2 A_a}{2s} \quad (10.2.19)$$

This approximation of  $L$  generally overestimates the actual value because it neglects iron reluctance, flux leakage, and the geometric size of the coil. Since the relationship between  $B$  and  $H$ , and between  $\phi$  and  $i$ , are nonlinear,  $L$  will also depend on the operating point of the  $B - H$  diagram. Therefore, we can also define a differential inductance

$$L_d = n \frac{d\phi}{di} = \frac{d\psi}{di} \quad (10.2.20)$$

which corresponds to the gradient in a  $\phi - i$  diagram.

The inductance of a bearing magnet is also of importance to the design of the power supply, which embodies a current regulator. According to the law of inductance, the induced voltage  $u$  in coil with  $n$  turns equals

$$u = \frac{d\psi}{dt} = L_d \frac{di}{dt} \quad (10.2.21)$$

#### 10.2.4 Magnetic Force

In contrast to the forces acting on conductors in a magnetic field (Lorentz force), the attraction force of magnets is generated at the boundaries between differing permeability  $\mu$ . The calculation of these forces is based on the field energy.

We consider the energy  $W_a$  stored in the volume of the air gap,  $V_a = 2sA_a$ . In the case of the homogeneous field in the air gap of the magnetic loop, as represented in Figure 10.10a, the stored energy  $W_a$  is given as follows

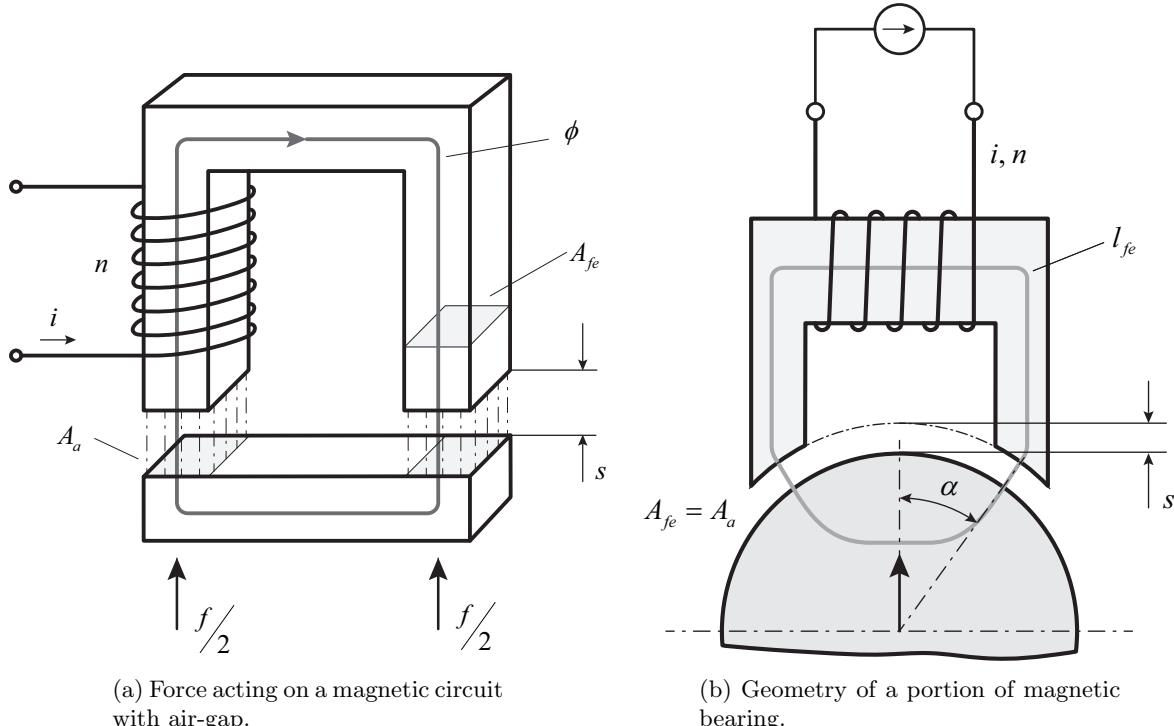


Figure 10.10: Force and geometry of a magnetic bearing portion of circuit.

$$W_a = \frac{1}{2} B_a H_a V_a = \frac{1}{2} B_a H_a A_a (2s) \quad (10.2.22)$$

The force acting on the ferromagnetic body ( $\mu_r \gg 1$ ) is generated by a change of the field energy in the air gap, as a function of the body displacement. For small displacements  $ds$  the magnetic flux  $B_a A_a$  remains constant.

When the air gap  $s$  increases by  $ds$ , the volume  $V_a = 2sA_a$  increases, and the energy  $W_a$  in the field increases by  $dW_a$ . This energy has to be provided mechanically, i.e. an attractive

force has to be overcome. Thus, force  $f$  equals the partial derivative of the field energy  $W_a$  with respect to the air gap  $s$  (principle of virtual displacement):

$$f = -\frac{\partial W_a}{\partial s} = B_a H_a A_a = \frac{B_a^2 A_a}{\mu_0} \quad (10.2.23)$$

In the case of a closed system, the force  $f$  can be derived from the principle of virtual displacement. For electromagnets, (Figure 10.10), electric energy is introduced into the system through the coil terminals to set up the magnetic field. In order for the Eq. (10.2.23) to remain valid, the differentiation has to be carried out as if there is no electric energy exchange between the coil and its power supply, i.e. when flux density  $B$  remains constant. To derive force  $f$  as a function of coil current and the air gap,  $B(i, s)$  is inserted into (10.2.23) after differentiating. In the simplest of cases where the iron is neglected,  $B_l$  is replaced by (10.2.17). The resulting force  $f$  will be

$$f = \mu_0 A_a \left( \frac{ni}{2s} \right)^2 = \frac{1}{4} \mu_0 n^2 A_a \frac{i^2}{s^2} = \lambda \frac{i^2}{s^2} \quad (10.2.24)$$

in which the area  $A_a$  is assumed to be the *projected* area of the pole face, rather than the curved surface area.

Equation (10.2.24) shows the quadratic dependence of the force on the current and the inversely quadratic dependence on the air gap, as illustrated in both magnetic poles affect the rotor with an angle  $\alpha$  (Figure 10.10b), as opposed to the model of the U-shaped electromagnet circuit shown in Figure 10.10a. In the case of a radial bearing with four pole pairs  $\alpha$  equals, e.g. 22.5 deg. Considering  $\alpha$  Eq. (10.2.24) results as follows

$$f = \frac{1}{4} \mu_0 n^2 A_a \frac{i^2}{s^2} \cos \alpha = \lambda \frac{i^2}{s^2} \cos \alpha \quad (10.2.25)$$

### 10.2.5 Linearization of the Force-Current Relation

The force of a electromagnet at an operating point can be written in the linearized form

$$f_y(y, i) = k_i i_y - k_s y \quad (10.2.26)$$

Here, the force  $f_y$  is given by a tangent (slope  $k_i$ ) to a parabola at the operating point. This operating point is given by the bias current  $i_0$  and the nominal air gap  $s$ , Figure 10.10. Furthermore,  $k_s$  is the slope of the curve  $1/(s - y)^2$  at the operating point, Figure 10.11.

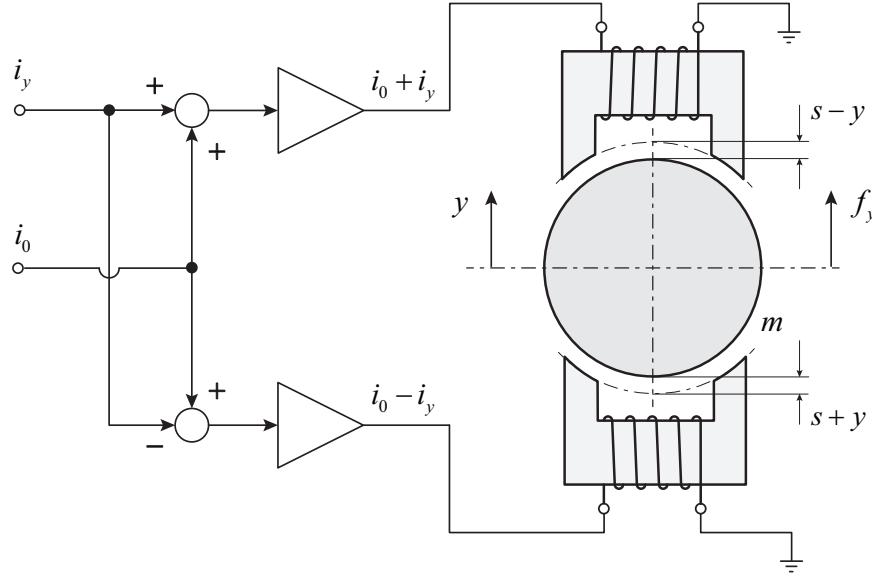


Figure 10.11: Differential driving mode of the bearing magnets.

Usually, two counteracting electromagnets are operated in a bearing magnet, Figure 10.11. This configuration makes it possible to generate both positive and negative forces. In the case of the so-called differential driving mode, one magnet is driven with the sum of bias current  $i_0$  and control current  $i_y$ , and the other one with the difference  $(i_0 - i_y)$ . Consequently, if the magnetization of the iron is neglected, this scheme produces a linear force-current relation.

Force  $f_y$  in Figure 10.11 represent the difference of force between both electromagnets. Both forces are obtained by inserting the sum  $(i_0 + i_y)$  and the difference  $(i_0 - i_y)$  for current  $i$  in Eq. (10.2.25). For the air gaps, the terms  $(s - y)$  and  $(s + y)$  are inserted:

$$f_y(y, i) = f_+(y, i) - f_-(y, i) = k \left[ \frac{(i_0 + i_y)^2}{(s - y)^2} - \frac{(i_0 - i_y)^2}{(s + y)^2} \right] \cos \alpha \quad (10.2.27)$$

with

$$k = \frac{1}{4} \mu_0 n^2 A_a \quad (10.2.28)$$

# Bibliography

- [1] A. Mehrle, *Advanced Control Engineering*. Management Center Innsbruck (MCI) 2021.
- [2] K. Ogata, *Modern Control Engineering*. Pearson 2009.
- [3] K. Ogata, *Discrete-time Control Systems*. Prentice-Hall 1987.
- [4] R. Zanasi, L. Biagiotti, *Lecture Note on Control and Linear Systems*. University of Modena.
- [5] R. Isermann, M. Munchhof, *Identification of Dynamic Systems: An Introduction With Applications*. Springer 2011.
- [6] B. P. Gibb, *Advanced Kalman Filtering, Least-Squares and Modeling*. J. Wiley 2011.
- [7] L. Ljung, *System Identification. Theory for the User*. Pearson 1999.
- [8] F. Lewis, *Optimal Control*. J. Wiley 2012.
- [9] N. Mohan, T.M. Undeland, W.P. Robbins, *Power Electronics*. Second edition J. Wiley 1995.
- [10] S. D. Umans, *Fitzgerald & Kingsley's Electric Machinery*. McGraw Hill 2014.
- [11] R. Marino, P. Tomei, C. M. Verrelli, *Induction Motor Control Design*. Springer 2010.
- [12] T. Geyer, *Model Predictive Control of High Power Converters and Industrial Drives*. Wiley 2017.
- [13] J. Rodriguez, P. Cortes, *Predictive Control of Power Converters and Electrical Drives*. Wiley 2012.
- [14] P. Krause, O. Wasynchuk, S. Sudhoff, S. Pekarek, *Analysis of Electric Machinery and Drive Systems*. Third Edition Wiley 2013.
- [15] D.D. Hanselman, *Brushless Motor. Magnetic Design, Performance and Control*. E.Man Press 2012.
- [16] J.R. Hendershot, T.J.E. Miller, *Design of Brushless Permanent-Magnet Machines*. Motor Design Books LLC 2010.

---

## Bibliography

---

- [17] J. Pyrhönen, T. Jokinen, V. Hrabovcova, H. Niemelä, *Design of rotating electrical machines*. Wiley 2008.
  - [18] T. Lipo, *Introduction to AC Machine Design*. Wiley 2017.
  - [19] N. Bianchi, T. Jahns, *Design, analysis, and control of interior PM synchronous machines*. CLEUP 2004.
  - [20] F. Giri, *AC Electric Motors Control: Advanced Design Techniques and Applications*. Wiley 2013.
  - [21] Edward P. Furlani, *Permanent magnet and electromagnetical devices*. Academic Press 2001.
  - [22] H.H. Woodson, J.R. Melcher, *Electromechanical Dynamics, Part I: Discrete Systems*. John Wiley 1968.
  - [23] R. Feynmann, R. Leighton, M. Sands, *Lecture on Physics*. Basic Book 1964.
  - [24] J. D. Jackson, *Classical Electrodynamics*. John Wiley 1962.
  - [25] W.K.H. Panofsky, M. Phillips, *Classical Electricity and Magnetism*. Addison-Wesley 1962.
  - [26] D. J. Griffiths, *Introduction to Electrodynamics*. Fourth Edition Cambridge University Press 2017.
  - [27] F. Fujikawa, Z. Q. Yang, T. Koga, *Robust and Fast Speed Control for Torsional System Based on State-Space Method*. Proceedings IECON 1991.
  - [28] L. Wang, *Model Predictive Control System Design and Implementation Using Matlab<sup>®</sup>*. Springer 2010.
  - [29] G. Tao, *Adaptive Control Design and Analysis*. Wiley-Interscience 2003.
  - [30] M. Sunwoo, Ka C. Cheok, N. J. Huang, *Model Reference Adaptive Control for Vehicle Active Suspension Systems*. IEEE Transaction on industrial electronics 1991.
  - [31] M. Verhaegen, V. Verdult, *Filtering and System Identification. A Least Square Approach*. Cambridge University Press 2007.
  - [32] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Pearson 1993.
  - [33] H. K. Khalil, *Nonlinear Systems*. Third edition, Prentice Hall 2002.
  - [34] H.K. Khalil, *Nonlinear Control*. Pearson 2015.
  - [35] J.J.E. Slotine, W. Li, *Applied Nonlinear Control*. Prentice Hall 1991.
  - [36] D.R. Merkin, *Introduction to the Theory of Stability*. Springer Hall 1993.
-

---

## Bibliography

---

- [37] P.A. Ioannou, J. Sun, *Robust Adaptive Control*. Dover 2012.
- [38] G. Strang, *Linear Algebra*. Fourth Edition Brooks-Cole 2006.
- [39] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. SIAM 2000.
- [40] E. D. Sontag, *Mathematical Control Theory*. Second Edition Springer 1998.
- [41] G. L. Plett, *Battery Modeling. Volume I*. Artech House 2015.
- [42] G. L. Plett, *Battery Modeling. Volume II*. Artech House 2015.
- [43] G. L. Plett, *Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs. Part 1. Background*. Journal of Power Sources 134 (2004).
- [44] G. L. Plett, *Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs. Part 2. Modeling and identification*. Journal of Power Sources 134 (2004).
- [45] G. L. Plett, *Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs. Part 3. State and parameter estimation*. Journal of Power Sources 134 (2004)
- [46] H. Merritt, *Hydraulic Control Systems*. Wiley 1967.
- [47] G. K. Costa, N. Sepehri, *Hydrostatic Transmission and Actuators*. Wiley 2015
- [48] N. D. Manring, *Hydraulic Control System*. Wiley 2005
- [49] A. Vacca, G. Franzoni, *Hydraulic Fluid Power: Fundamentals, Applications, and Circuit Design*. Wiley 2021
- [50] G. Schweitzer, E.H. Maslen, *Magnetic Bearings*. Springer 2009.
- [51] J. Lévine, J. Lottin, JC. Ponsart, A nonlinear approach to the control of magnetic bearing Bearings. *IEEE Transactions on Control Systems Technology*. Vol. 4, pp. 524-544, September 1996.