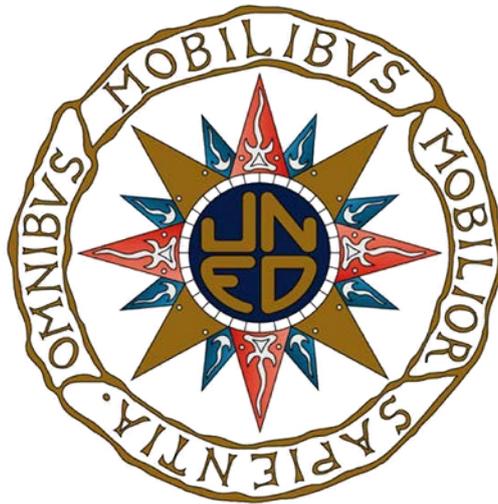


Tesis Doctoral



**Diseño de controladores
H-infinito para el control de
plantas mal condicionadas
en presencia de perturbaciones**

***Design of H-Infinity controllers
for ill-conditioned plants
in presence of disturbances***

**David Sánchez de la Llana
Licenciado en Ciencias Físicas**

Departamento de Informática y Automática
Escuela Técnica Superior de Ingeniería Informática
Universidad Nacional de Educación a Distancia

**18/05/2015
Issue 1.0.1**

Departamento: Informática y Automática ETS. Ingeniería Informática UNED

Título: Contribuciones al diseño de controladores H-infinito para el control de plantas mal condicionadas en presencia de perturbaciones

Autor: David Sánchez de la Llana

Título: Licenciado en Física por la Universidad Complutense de Madrid

Director: Joaquín Aranda Almansa

Agradecimientos:

Gracias a Joaquín Aranda por su paciencia “H-infinity” conmigo. No es fácil compaginar una tesis y un trabajo que requiere una altísima dedicación.

Grazie ai miei colleghi di ELV, Irene Cruciani, Christophe Roux and Gregorij Marteens per il suo aiuto. E stata una grande avventura questo VEGA e un grande piacere lavorare con voi... E c'è l'abbiamo fatta con questo cavolo di lanciatore!

Thanks to my ESA colleague Samir Bennani for its help and guidance.

Finalmente, gracias a mis padres.

La tesis va dedicada a Sergio y Goyito que desgraciadamente ya no están con nosotros.

LIST OF CONTENTS

1	Introduction, Objectives, Contributions and Structure	17
1.1	Introduction.....	17
1.2	Objectives	19
1.3	Structure of the Thesis.....	20
2	Brief Review of Classical Robust Control.....	21
2.1	Review of concepts of feedback systems.....	21
2.1.1	Definition of the transfer functions.....	21
2.1.1.1	Output Sensitivity (S_o).....	21
2.1.1.2	Complementary of the Output Sensitivity (T_o)	23
2.1.1.3	Control Sensitivity (S_iK or KS_o)	24
2.1.1.4	Sensitivity at the plant input (S_oP).....	24
2.1.2	Summary of tradeoffs in feedback control	25
2.1.3	Controllability and observability.....	27
2.1.4	State feedback and output injection.....	29
2.1.5	Observers and Observers-Based Controllers.....	29
2.1.6	Fundamental limitations on feedback systems.....	31
2.1.6.1	Fundamental limitations on S and T	31
2.1.6.2	Bode integral formula or the ‘waterbed effect’	31
2.1.6.3	Fundamental limitations on S and T due to RHP poles and zeros	32
2.2	The formulation of the Robust Control Problem	34
2.2.1	The size of a transfer function: singular values and norms.....	34
2.2.1.1	Singular values	34
2.2.1.2	Norms	36
2.2.2	Introduction of weights	38
2.2.2.1	Weights for S_o	39
2.2.2.2	Weights for T_o	40
2.2.2.3	Weights for KS_o	41
2.2.2.4	Weights for S_oP	42
2.2.2.5	Some practical notes for weights.....	42
2.2.3	The importance of scaling in MIMO	43
2.2.4	Robust stability and robust performance	45
2.2.4.1	Internal Stability and Small Gain theorem.....	45
2.2.4.2	Robust Stability.....	47
2.2.4.3	Robust Performance	48
2.2.4.4	Summary.....	49
3	The H-Infinity Theory	50
3.1	General formulation of the control problem: the augmented plant	50
3.2	Mathematical solutions to the H-infinity problem.....	51
3.2.1	The statement on the problem.....	51
3.2.2	Some key concepts.....	52
3.2.2.1	Hamiltonian function of a system.....	53
3.2.2.2	The Bounded Real Lemma	55
3.2.2.3	Connection between the Hamiltonian and a Riccati equation	57

3.2.1	The DGKF solution	58
3.2.1.1	Preconditions	58
3.2.1.2	The suboptimal solution	59
3.2.1.3	The family of suboptimal solutions	60
3.2.1.4	Comments on the DGKF solution	60
3.2.1	The LMI solution	62
3.2.1.1	Specific case: static state feedback	63
3.2.1.2	General Case	64
3.2.2	Properties of the H-Infinity controllers	70
3.3	History of H-infinity	71
3.4	H-infinity Mixed Sensitivity	73
3.4.1	H-infinity Mixed Sensitivity S-KS: reference tracking	74
3.4.2	H-infinity Mixed Sensitivity S-T	76
3.4.1	Problems of H-infinity Mixed Sensitivity	78
3.5	Current state of the art for H-infinity	78
3.5.1	Structured H-Infinity	79
3.5.1.1	Practical problems with “classical” H-Infinity	79
3.5.1.2	Fundamentals of H-Infinity Structured	80
3.5.2	Linear Parameter Variant (LPV) systems	85
4	The problems of the H-Infinity method when applied to ill-conditioned plants	87
4.1	The CDC distillation process benchmark	87
4.1.1	Plant Model	87
4.1.2	Control Specifications of the CDC problem	90
4.2	Problems of H-Infinity Sensitivity with ill conditioned plants and disturbances	91
4.2.1	The cause: the controller trends to invert the plant	91
4.2.2	Demonstration of the problem with the CDC benchmark	92
4.3	Alternative H-Infinity methods for control of ill conditioned plants	98
4.3.1	Scaling of the plant	98
4.3.2	Kwakernaak optimization for H-Infinity	99
4.3.3	Christen – Geering approach	101
4.3.4	Apkarian structured approach	106
4.4	CDC Benchmarks results	107
4.4.1	Reference μ controller (SKO96)	108
4.4.2	Second reference μ controller (LUND99)	112
4.4.3	Third reference μ controller (Balas 2012)	116
4.4.4	H-Infinity Kwakeernaak (KWA) controller	121
4.4.5	H-Infinity Christen-Geering (CHGE) controller	126
4.4.6	H-Infinity Structured (Apkarian) controller	131
4.5	Conclusions of applying H-Infinity to the CDC benchmark	142
5	Application to a real world application: the VEGA Launcher	144
5.1	Introduction to the VEGA Launcher	144
5.1.1	The Avionics of the VEGA Launcher	147
5.1.2	The GNC algorithms of the VEGA Launcher	150
5.1.3	The original TVC control of the VEGA Launcher	151
5.2	VEGA atmospheric flight in presence of roll	153
5.2.1	Background of the roll problem	153

5.2.2	The updated TVC control with roll compensation	157
5.3	Physical Equations of the VEGA Launcher	158
5.3.1	Reference Frames and Conventions.....	158
5.3.1.1	Reference Frames	158
5.3.1.2	Simplified Reference Frames.....	162
5.3.1.3	Angle conventions.....	164
5.3.2	Description of the forces and momentums acting on the LV.....	164
5.3.2.1	Aerodynamics forces and momentums	164
5.3.2.2	Propulsive forces and momentums.....	165
5.3.2.3	Nozzle torque	167
5.3.2.4	Gravity force	167
5.3.3	Non-linear equations (6DoF rigid body).....	167
5.3.3.1	Need of the transport theorem.....	168
5.3.3.2	Rotational equation	169
5.3.3.3	Translational equation (Tsiolkovsky formula)	170
5.3.3.4	Translational equation.....	171
5.3.4	Derivation and solution of the Linear equations for a rocket	173
5.3.4.1	Solution of the Rotational equation	173
5.3.4.2	Solution to the Translational equation.....	180
5.3.5	State space model of the LV	184
5.3.5.1	Study of the open loop plant in the frequency domain	186
5.3.5.2	Study of the model uncertainty for the plant.....	187
5.4	miniVEGA: a non Linear Simulator for the VEGA Launcher	190
5.4.1	Introduction to the VEGA Launcher simulators	190
5.4.2	Structure of mini-VEGA.....	191
5.4.2.1	Overall structure.....	191
5.4.2.2	The Measurement subsystem.....	192
5.4.2.3	The GNC subsystem.....	193
5.4.2.4	The TVC subsystem.....	196
5.4.2.5	The Wind model	197
5.4.2.6	The Launch Vehicle model	199
5.4.2.7	The mathematical library	208
5.4.2.8	Non modeled parts in miniVEGA	208
5.5	Comparison of controllers for the VEGA Launcher	209
5.5.1	The reference ELV controllers	209
5.5.1.1	The original ELV PID controller.....	209
5.5.1.2	The ELV PID controller with gyroscopic compensation	209
5.5.2	The proposed H-infinity controllers	211
5.5.2.1	The proposed H-infinity controller (CHGE).....	211
5.5.2.2	The proposed H-infinity controller (Structured)	216
5.5.3	Simulation scenarios.....	222
5.5.4	Simulation analysis	224
5.5.4.1	Scenario 1: response to ramps in absence of winds	224
5.5.4.2	Scenario 2: response to ramps in presence of winds.....	226
5.5.4.3	Scenario 3: disturbance rejections with wind gusts	229
5.5.5	Comparison of controllers	233

6	Contributions, Overall Conclusions and Future Work.....	234
6.1	Contributions.....	234
6.2	Overall Conclusions.....	235
6.3	Future work.....	235
7	Bibliography.....	237
7.1	List of references.....	237
7.2	Comments on bibliography.....	240
8	Appendix: Demonstrations.....	242
8.1	Mathematical background.....	242
8.1.1	Some notes on complex functions.....	242
8.1.1.1	Cauchy Riemann equations.....	242
8.1.1.2	Cauchy Goursat theorem.....	244
8.1.1.3	Laplace transform.....	248
8.1.2	Introduction to norms.....	249
8.1.3	Mathematical introduction to LFT.....	252
8.1.4	Kernel and null space.....	254
8.1.5	LMIs.....	254
8.1.6	Schur complement.....	255
8.1.7	The Congruence Transformation.....	258
8.2	Demonstrations for chapter 3.....	258
8.2.1	Maximum modulus theorem.....	258
8.2.2	Youla Parametrization of stabilizing controllers.....	258
8.2.3	Stability of a system with Lyapunov method.....	259
8.2.4	Hamiltonian and LMI equivalence for a linear system.....	260
8.2.5	Hamiltonian matrix, Riccati Equation and H_∞ norm.....	261
8.2.5.1	Hamiltonian matrices.....	261
8.2.5.2	Mathematical equivalence of Hamiltonian matrix and Riccati equation.....	262
8.2.5.3	Hamiltonian matrix and H_∞ norm of a system.....	263
8.2.5.4	Computing the H_∞ norm of a system.....	264
8.2.6	LMI of the closed loop in plant and controller variables.....	264
8.2.7	The Elimination Lemma.....	268
8.2.8	Final step of LMI demonstration.....	269
8.3	Demonstrations for chapter 4.....	273
8.3.1	State space formulas for augmented plants.....	273
8.3.1.1	General formulas.....	273
8.3.1.2	Demonstration of Kwakernaak state space representation.....	274
8.3.1.3	Demonstration of Christen-Geering state space representation.....	275
8.4	Demonstrations for chapter 5.....	278
8.4.1	No effect of gravity in LV acceleration (gravity turn).....	278
8.4.2	Deduction of the Transport Theorem.....	280
8.4.3	Expression of angular velocities in function of inertial variables.....	282
8.4.4	Expression of angle of attack in inertial frame.....	283
8.4.5	Check of rotational and translational formulas under rotation of 90 degrees.....	286
8.4.6	Quaternions.....	288
9	Appendix: Developed Software.....	291

9.1	Software for the CDC benchmark.....	291
9.1.1	Software User Manual Notes.....	291
9.1.1.1	Execution.....	291
9.1.1.2	List of important files.....	291
9.2	Software for the VEGA Launcher.....	292
9.2.1	Software User Manual Notes.....	292
9.2.1.1	Execution.....	292
9.2.1.2	List of important files.....	292
9.2.2	Function details.....	293
9.2.2.1	Function designK().....	293
9.2.2.2	Function runSim().....	294
9.2.2.3	Function runBatchSim.....	295
9.2.2.4	Function plotSim().....	296

LIST OF MATHEMATICAL SYMBOLS AND ACRONYMS

Symbol	Meaning
A^H	Transpose conjugate of matrix A
A^T	Transpose of matrix A
A^\perp	Orthogonal of matrix A
A^{-1}	Inverse of matrix A
A_6	Aerodynamic stability parameter (in VEGA)
(A,B,C,D)	State space matrices of a system
CDC	Control of a Distillation Column (benchmark)
β_ψ, β_θ	Angles of actuators on psi and theta planes
α	Angle of attack
$CHGE$	Christeen Geering (H-Infinity variant)
COG	Center of Gravity
CP	Center of aerodynamic Pressure (in VEGA)
C_x, C_N	Axial and normal aerodynamic coefficients (in VEGA)
$DGKF$	Doyle, Glover, P. Khargonekar, and B. Francis (state space solution of H-Infinity controllers)
EMA	Electro Mechanical Actuator
F, F_{Aero}	Force, Aerodynamic force
$F(s)$	Laplace transform
$F_l(P, K)$ $F_u(P, K)$	Lower and upper Linear Fractional Transformation of P and K
$G(s)$	Plant (usually in open loop)
γ	Gamma value of H-infinity optimization
$\ \cdot \ _2, \ \cdot \ _\infty$	H ₂ , H-infinity norms
$I, I_{xx}, I_{yy}, I_{zz}$	Moment of Inertia (along principal axis)
IMU	Inertial Measurement Unit
K	Controller
K_1	Controllability Parameter (in VEGA)
LFT	Linear Functional Transformation
LHP	Left half Plane (complex plane)
LMI	Linear Matrix Inequality
M	Mach number (in VEGA)
m	Mass of the launcher (in VEGA)
$M_{Aero} M_{Prop}$	Aerodynamic Torque, Propulsion Torque (in VEGA)
μ	(mu). Control theory
$\ \cdot \ $	Norm operator
ω	Frequency
$P(s)$	Plant
P_{dyn}	Dynamic Pressure

Symbol	Meaning
PP	Pivot Point (of the nozzle, in VEGA)
p, q, r	Angular rates (roll, yaw, pitch) (in VEGA)
p_0	(constant) Roll rate
q	Quaternion (in VEGA)
RACS	Roll and Attitude Control System (in VEGA)
RCT	Reaction Control Thrusters
RHP	Right half Plane (complex plane)
ρ	Air density (in VEGA)
S_R	Transversal reference surface (in VEGA)
SRM	Solid Rocket Motor
S_i, S_o	Input and Output Sensitivity
$\bar{\sigma}, \underline{\sigma}, \underline{\underline{\sigma}}$	Singular value, maximum singular value, minimum singular value
T_i, T_o	Input and Output Complementary Sensitivity
T	Thrust (in VEGA)
T_{wz}	Transfer function w to z. (Usually of the augmented plant)
TVC	Thrust Vector Control (in VEGA)
ϕ, φ, ϑ	Attitude angles (roll, yaw, pitch)
V	Modulus of the relative velocity (taking into account wind effects)
$V(x)$	Storage function (or potential function)
v_x, v_y, v_z u, v, w	Velocities in X, Y, Z axis in body frame
$W(s)$	Weight
$w(s)$	(Signal). External signals of the augmented plant
X, Y, Z	Position in X, Y, Z axis in inertial frame
$\dot{X}, \dot{Y}, \dot{Z}$	Derivative of position in X, Y, Z axis in inertial frame
x_{CP}, x_{COG}	Position in X axis of CP and COG
$x(s), \hat{x}(s)$	(States). State and estimated state of a system.
$y(s), \hat{y}(s)$	(Signal). Output and estimated output of a system.
$u(s)$	(Signal). Control signal in an augmented plant
z	Complex variable
$z(s)$	(Signal). Weighted signals in an augmented plant

LIST OF TABLES

Table 4-1: S2 criteria for Mixed Sensitivity controller	93
Table 4-2: S2 criteria for Mixed Sensitivity controller with no delay	97
Table 4-3: S2 criteria for SKO96 controller	109
Table 4-4: S2 criteria for the LUND99 controller	112
Table 4-5: S2 criteria for the Balas 2012 controller	118
Table 4-6: S2 criteria for the H-inf KWA controller	123
Table 4-7: S2 criteria for CHGE controllers	128
Table 4-8: S2 criteria for H-inf Structured controller	136
Table 4-9: Summary of CDC controllers	142

LIST OF FIGURES

Figure 2-1: Standard feedback system.....	21
Figure 2-2: Output Sensitivity as output disturbance	22
Figure 2-3: Output Sensitivity as tracking error	22
Figure 2-4: Multiplicative disturbance at output	23
Figure 2-5: Control Sensitivity	24
Figure 2-6: Sensitivity at plant input.....	25
Figure 2-7: S and T trade-offs	26
Figure 2-8: Open loop gain and weights	27
Figure 2-9: Luenberger observer	30
Figure 2-10: The waterbed effect.....	32
Figure 2-11: Visualization of a SVD in a 2x2 system	35
Figure 2-12: Typical shapes for S_o weights	39
Figure 2-13: Typical shapes for T_o weights	40
Figure 2-14: Typical form for KS_o weights.....	42
Figure 2-15: Augmented plant and scaling.....	44
Figure 2-16: Internal stability.....	45
Figure 2-17: Set-up for the Small Gain theorem	46
Figure 2-18: Multiplicative disturbance at output and stability	47
Figure 2-19: Robust stability	48
Figure 2-20: Robust performance	49
Figure 3-1: Robust control problem as LFT	50
Figure 3-2: The H-infinity problem as LFT.....	51
Figure 3-3: Separation structure of H-infinity controllers.....	61
Figure 3-4: The model matching problem	72
Figure 3-5: H-inf Mixed Sensitivity for reference tracking.....	74
Figure 3-6: H-inf Mixed Sensitivity for reference tracking as LFT.....	75
Figure 3-7: H-inf Mixed Sensitivity S-T	76
Figure 3-8: H-inf Mixed Sensitivity S-T a LFT	77
Figure 3-9: Robust control problem	80
Figure 3-10: Sub-differential	82
Figure 4-1: Distillation Plant.....	89
Figure 4-2: Singular values of the Open Loop Plant	89
Figure 4-3: H-inf Mixed Sensitivity controller response to sce1	94
Figure 4-4: H-inf Mixed Sensitivity controller response to sce2	94
Figure 4-5: H-inf Mixed Sensitivity controller response to sce3	95
Figure 4-6: H-inf Mixed Sensitivity SoP and To	96
Figure 4-7: Pole zero cancellation in H-infinity Mixed Sensitivity controller	97
Figure 4-8: Scaling for the plant for the design.....	99
Figure 4-9: Integrating the scaling on the controller.....	99
Figure 4-10: H-infinity controller with pre-filter.....	99
Figure 4-11: CHGE problem	102
Figure 4-12: A typical example for the H-infinity Structured.....	107
Figure 4-13: The Structured controller in diagonal view.....	107
Figure 4-14: SKO96 mu controller response to sce1	109

Figure 4-15: SKO96 mu controller response to sce2	110
Figure 4-16: SKO96 mu controller response to sce3	110
Figure 4-17: SKO96 mu controller S_oP and T_o	111
Figure 4-18: SKO96 mu controller K and KS_o	111
Figure 4-19 LUND99 mu controller response to sce1	113
Figure 4-20 LUND99 mu controller response to sce2	114
Figure 4-21: LUND99 mu controller response to sce3	114
Figure 4-22: LUND99 mu controller K and KS_o	115
Figure 4-23: LUND99 mu controller S_oP and T_o	116
Figure 4-24: block diagram with controller Balas 2012	117
Figure 4-25 Balas 2012 mu controller response to sce1	119
Figure 4-26 Balas 2012 mu controller response to sce2	119
Figure 4-27: Balas 2012 mu controller response to sce3	120
Figure 4-28: Balas 2012 mu controller KSo	120
Figure 4-29: Balas 2012 mu controller S_oP and T_o	121
Figure 4-30: KWA Controller response to sce1	123
Figure 4-31: KWA Controller response to sce2	124
Figure 4-32: KWA Controller response to sce3	124
Figure 4-33 KWA controller KS_o	125
Figure 4-34: KWA SoP and To Perturbed	125
Figure 4-35: CHGE controller response to sce1	129
Figure 4-36: CHGE controller response to sce2	129
Figure 4-37: CHGE controller response to sce3	130
Figure 4-38: CHGE controller KS_o	130
Figure 4-39: CHGE S_oP and T_o	131
Figure 4-40: Block diagram for the H-infinity Structured controller	132
Figure 4-41: The Structured controller in diagonal view	133
Figure 4-42: H-inf structured controller response to sce1	137
Figure 4-43: H-inf structured controller response to sce2	137
Figure 4-44: H-inf structured controller response to sce3	138
Figure 4-45: H-inf structured controller S_oP and T_o	138
Figure 4-46: H-inf structured controller K and KS_o	139
Figure 4-47: Block diagram for the H-infinity Structured controller with roll-off term	140
Figure 4-48: H-inf structured controller K and KS_o with roll-off filter	140
Figure 4-49: H-inf structured controller with roll-off response to sce 1	141
Figure 4-50: H-inf structured controller with roll-off S_oP and T_o	141
Figure 5-1: Schema of the VEGA Launcher	146
Figure 5-2: The closed loop of the VEGA Launcher	148
Figure 5-3: Simplified schema of the algorithms	150
Figure 5-4: The original TVC control	152
Figure 5-5: No coupling in absence of roll	154
Figure 5-6: TVC control with gyroscopic compensation	157
Figure 5-7: LV Body frame	159
Figure 5-8: View of the RTCs	159
Figure 5-9: Position of the EMA actuators	160
Figure 5-10: LV Initial Starting reference frame	161

Figure 5-11: Navigation, Guidance and Body RFs	163
Figure 5-12: Navigation, Guidance and Body RFs (2).....	164
Figure 5-13: Propulsive and aerodynamics forces in XZ (yaw) plane	166
Figure 5-14: Propulsive and aerodynamics forces in XY (pitch plane)	166
Figure 5-15: Thrust profile for first stage	166
Figure 5-16: Inertial and rotating frames.....	168
Figure 5-17: Deduction of Tsiolkovsky formula.....	171
Figure 5-18: Evolution of K_1 and A_6 along the first stage	187
Figure 5-19: Scattering of open loop (except roll)	188
Figure 5-20: Influence of roll rate in open loop plant.....	189
Figure: 5-21: Sigma values variation with roll rate.....	190
Figure: 5-22: Top level of miniVEGA	192
Figure 5-23: Model of the GNC in miniVEGA.....	195
Figure 5-24: Model of the TVC	197
Figure 5-25: Second order model with position and rate limiters	197
Figure 5-26: The Launch Vehicle model	199
Figure 5-27: The interpolation variables model	201
Figure 5-28: The aerodynamics model.....	203
Figure 5-29: The rotational dynamics.....	205
Figure 5-30: The translational dynamics.....	207
Figure 5-31: Schema of the ELV controller with Gyroscopic compensation	210
Figure 5-32: K and KS_0 for the CHGE controller	213
Figure 5-33: S_0P and T_0 for the CHGE controller	214
Figure 5-34: S_0 and S_0P for the CHGE controller	215
Figure 5-35: T_0 , T_i and weights for the CHGE controller	216
Figure 5-36: Setup for the K-infinity Structured controller	217
Figure 5-37: K and KS_0 for the structured controller	220
Figure 5-38: S_0P and T_0 for the structured controller	221
Figure 5-39: T_0 , T_i and weights for the structured controller	222
Figure 5-40: Scenario 1 roll rate $p = 0$ deg/s.....	224
Figure 5-41: Scenario 1 roll rate $p = 45$ deg/s.....	225
Figure 5-42: Scenario 2 roll rate $p = 0$ deg/s.....	226
Figure 5-43: Scenario 2 roll rate $p = 10$ deg/s.....	227
Figure 5-44: Scenario 2 roll rate $p = 45$ deg/s.....	228
Figure 5-45: Scenario 2 roll rate $p = 45$ deg/s (zoom).....	228
Figure 5-46: Scenario 3 roll rate $p = 0$ deg/s.....	229
Figure 5-47: Scenario 3 roll rate $p = 10$ deg/s.....	230
Figure 5-48: Scenario 3 roll rate at $p = 45$ deg/s (ELV controllers).....	231
Figure 5-49: Scenario 3 roll rate at $p = 45$ deg/s (H-Infinity controllers).....	232
Figure 8-1: The complex function z^2	243
Figure 8-2: Complex integral along a rectangular path.....	246
Figure 8-3: Complex integral along an arbitrary closed path.....	247
Figure 8-4: Path independence of complex integrals	248
Figure 8-5: Visualization of norms.....	250
Figure 8-6: Conservatism of p-norms.....	251
Figure 8-7: Lower LFT.....	252

Figure 8-8: Upper LFT.....	253
Figure 8-9: Internal model control	259
Figure 8-10: Setup for Hamiltonian calculus	263
Figure 8-11: Inertial and rotating frame	280
Figure 8-12: Inertial and rotating frame with same origin	280
Figure 8-13: Angle of attack (theta)	284
Figure 8-14: Angle of attack (psi)	285
Figure 8-15: Rotation of LV around x axis (-90 deg)	286

1 Introduction, Objectives and Structure

1.1 Introduction

New control theories have been developed along the last two decades of the 20th century. These new theories are oriented to multivariable control problems, where the classic methods either do not provide satisfactory results or require high level experienced control engineers and a lot of design effort.

One of these theories is the H-Infinity theory. It is an evolution of the LQR (Linear Quadratic Regulation) and LQG (Linear Quadratic Gaussian) theories. In turn, these theories were originated in the sixties in the aerospace world. LQG and LQR achieved good results in the paper, but not so good results in real MIMO applications (a well-known case is the recovery of stall of the F-8C Crusader fighter, see (Byrnes, 1991)). The main reason is that LQR and LQG do not manage the uncertainties of the plants.

The H-infinity addresses this problem of model uncertainty in a very general and elegant schema and provides a highly automated synthesis method, where the designer expresses the closed loop objectives as weights in the frequency domain and just executes the algorithms for finding the controller.

As drawbacks of the H-infinity method, we can mention three. First, some control objectives are often expressed in the time domain and not in the frequency domain. Second, the designed controller is usually of higher order than with classic methods (but the recent H-Infinity Structured technique can address this issue). The third one is the following: it has been known from time ago that the most commonly used variant of the H-Infinity method (the Mixed Sensitivity variant) has problems for the control of ill-conditioned plants. Some influential authors have arrived to the conclusion that this was a fundamental limitation of the most popular variant of H-infinity theory, the Mixed Sensitivity variant.

This thesis addresses mostly this third problem: the control with H-Infinity of ill-conditioned plants in presence of disturbances and evaluates systematically the available options, demonstrating that some of them are perfectly valid for such type of control.

The motivation for addressing this third problem was originated during the “Diploma de Estudios Avanzados” where a control of the RCAM aircraft (see (Aranda, 2000)) was attempted. It was found that in fact the H-Infinity Mixed Sensitivity had problems to control the longitudinal dynamics (that has a strong coupling between the throttle and elevator inputs, the so called “phugoid mode”).

The aerospace industry is adopting the new control techniques as μ , H-infinity, etc., if well very slowly: reliability, safety considerations, cultural inertia... places an important role here. Aerospace industry is by definition very conservative and has a big amount of know-how with classical methods (basically tuned PIDs). This know-how is perceived as “industrial property” not to be lost. However, these classic methods require very skilled and experienced designers and extensive time domain simulations that are very costly in time and effort.

A second motivation for this thesis is to demonstrate that new methods can be used for a typical aerospace problem obtaining similar performances to the classical method but with reduced development cost and robustness guaranteed “by design”.

1.2 Objectives

The objective of this thesis is to study the **practical** applicability of the H-Infinity to the control of **ill-conditioned** plants in presence of disturbances. The following sub-objectives are identified:

- To identify interesting examples of ill conditioned plants control problems. Two examples are identified: one academic problem (CDC distillation process) and one real world problem (Control of the first stage of the VEGA Launcher in presence of roll).
- To confirm the problems of the Mixed Sensitivity variant of the H-Infinity method for controlling these plants.
- To study improvements of the H-Infinity theory to manage the problem of controlling ill conditioned plants. To use these new optimized methods in one of the examples of ill conditioned plants identified. To compare exhaustively these methods in terms of performances, design effort and robustness.
- To study and propose solutions for dealing with the 'practical problems' of the H-Infinity method: expression of time specifications on the frequency domain, expressions of parametric uncertainties on the frequency domain, to provide rules for the selection of weights, etc.
- To apply the obtained results to another of the examples identified: the control of the VEGA Launcher Vehicle in presence of roll during the first stage and to compare with the controller used in the real Launcher. To compare the H-Infinity optimized methods with the real controllers (classical PIDs plus tuning filters) in terms of performances, design effort and robustness.

1.3 Structure of the Thesis

The structure of this thesis is as follows:

This first chapter is the introduction containing the Objectives of the thesis.

Chapter 2 contains a review of “classical” Robust Control notions. It should be considered only as a reminder if well an important effort has been done to describe traditional concepts from a modern point of view.

Chapter 3 focuses on the H-infinity theory. It contains the formulation of the problem, provides a list of the available mathematical solutions and fully describes one of them (the LMI solution). Then a review of the story and the current state of the art of the H-Infinity theory are presented (with focus in the new H-Infinity Structured theory). The Chapter 3 has good tutorial value and could serve as nucleus for lectures or presentations in an H-Infinity course.

Chapter 4 describes the problems of the H-infinity theory for control of “ill conditioned” plants in presence of disturbances that is the main goal of this thesis. An academic example (the CDC benchmark) demonstrates the unsuitability of the widely used H-infinity “Mixed Sensitivity” techniques for controlling the plant. Then, alternative H-infinity techniques to cope with the problem are systematically studied. Finally the CDC test bench is executed on these H-infinity variants and confronted with some reference μ controllers.

Chapter 5 provides the application of the previous result to a “real” control problem: the control of the VEGA Launcher during the first stage in presence of roll rate. First the VEGA Launcher is described. Then a detailed mathematical 6DoF model of the Launcher dynamics is developed. The simulator is implemented in Matlab/Simulink™. A set of scenarios is run in this simulator confronting the H-infinity controllers developed using techniques described in chapter 4. These controllers are compared against the real controllers implemented in the Launcher.

Chapter 6 contains the overall conclusions and intended future work.

Chapter 7 contains the commented bibliography.

Chapter 8 contains an appendix with demonstrations and mathematical background used along the thesis. We have opted to locate long demonstrations into appendices to make easier the reading of the relevant chapters.

Chapter 9 contains notes about the software developed on the frame of this thesis.

2 Brief Review of Classical Robust Control

The classical robust control notions and concepts are reviewed in this chapter.

2.1 Review of concepts of feedback systems

Given the standard feedback system Figure 2-1:

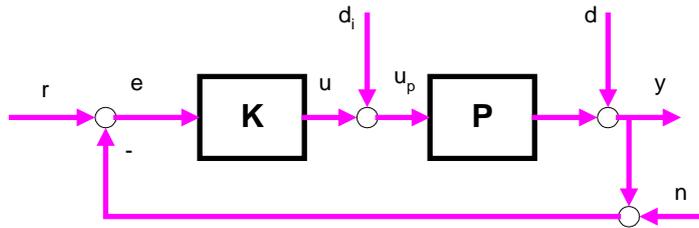


Figure 2-1: Standard feedback system

Where:

- r : reference signal
- e : error
- u : control signal
- u_p : input to the plant
- d_i : disturbances at plant input
- d : disturbances at plant outputs
- y : measured signals
- n : noise

There are eight independent functions from the inputs (r, d_i, d, n) to the outputs (u, y). From this, point, we assume that there is no measurement noise so having only four independent transfer functions.

It is shown hereafter that the performances and robustness of the system are defined by the shape of these transfer functions.

2.1.1 Definition of the transfer functions

2.1.1.1 Output Sensitivity (S_o)

The output sensitivity, S_o , is defined as the TF from the disturbances at plant output to the outputs, or also was the TF from the references to the error signal.

Removing the signals (d_i, r, n), we have the following system (Figure 2-2):

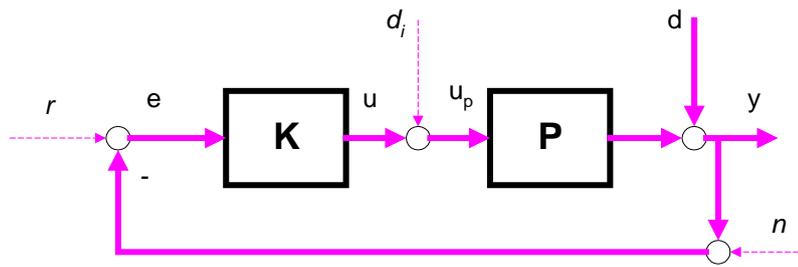


Figure 2-2: Output Sensitivity as output disturbance

Being the transfer function from d to y :

$$y = (1 + PK)^{-1} d = S_o d$$

Instead, if we consider the TF from r to e (Figure 2-3):

$$e = (1 + PK)^{-1} r = S_o r$$

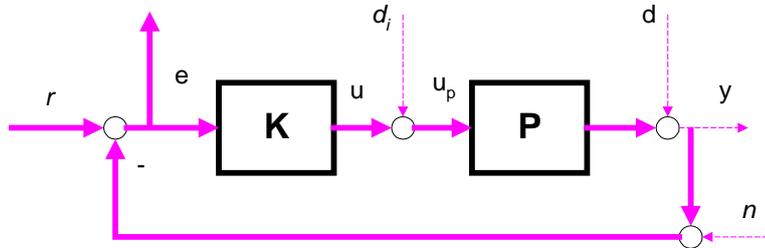


Figure 2-3: Output Sensitivity as tracking error

We want $|S_o|$ to be as small as possible for achieving the following objectives:

1. Decrease the effect of disturbances at plant output: if the gain of S_o is small, the influence of a disturbance will be decreased by that factor.
2. Good reference tracking: if S_o is small, the error e , defined as the difference between the reference and the measured output will be small. If the error is small, we are following well the reference.

In principle, it is *not* possible to make S_o small all over the frequency range: in a real system, the product of PK is a proper transfer function (the order of the denominator is equal or higher than the order of the numerator). This implies that $PK \rightarrow 0$ when $s \rightarrow \infty$. So, $(1 + PK)^{-1} \rightarrow 1$ when $s \rightarrow \infty$. I.e. above a defined frequency, the sensitivity is equal to 1.

We need to formulate mathematically the exigency ‘ S_o shall be small at low frequencies’ by introducing weights. This point will be introduced later.

2.1.1.2 Complementary of the Output Sensitivity (T_o)

It is defined as the TF from the references to the measured outputs:

$$y = (1 + PK)^{-1} PK r = T_o r$$

The name is due to the following **fundamental relation**:

$$S + T = (1 + PK)^{-1} + (1 + PK)^{-1} PK = 1$$

It is very important to note that $S + T = 1$, not that the modulus $|S| + |T| = 1$.

In principle, in order to have a good reference tracking, the objective is to achieve T_o equal to 1 in all the frequencies, so $y \approx r$.

This is not possible by principle because $(1 + PK)^{-1} PK \rightarrow 0$ when $s \rightarrow \infty$. I.e. at high frequencies T_o is zero in real physical systems (the input changes so fast that the system cannot follow it). This is in agreement of the fact that $S_o \rightarrow 1$ when $s \rightarrow \infty$.

We are interested in that T_o is ≈ 1 up to a given frequency and then decreases for higher frequencies.

The shape of T_o determines the robustness of the plant in **presence of noise and modeling errors**.

For demonstrating previous fact, let's suppose that the stable plant P is affected by a multiplicative disturbance (Figure 2-4).

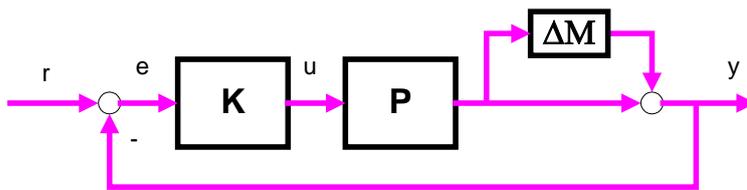


Figure 2-4: Multiplicative disturbance at output

The system will be stable if the characteristic equation has no zeros on the right semi plane, i.e.

$$\begin{aligned} \det [I + (I + \Delta_M) PK] &= \det [I + PK + \Delta_M PK] = \dots \quad \text{extracting the term } (I + PK) \\ \det [(I + PK) (I + \Delta_M PK (I + PK)^{-1})] &= \\ \det [I + PK] \det [I + \Delta_M T_o] & \end{aligned}$$

But $(I + PK)$ is stable because the nominal plant P is stable so the first determinant is always different from zero.

For the second determinant, if $\Delta_M T$ is minor than 1 (the concept of “norm” will be introduced later), the whole determinant is always different from zero and the closed loop system is stable.

We see that T_o determines the stability margin against multiplicative disturbances at the output. Any disturbance with $\Delta_M < T_o^{-1}$ does not de-stabilizes the system.

2.1.1.3 Control Sensitivity ($S_i K$ or $K S_o$)

It is defined as the TF from the references r to the control signal u generated by the controller (Figure 2-5):

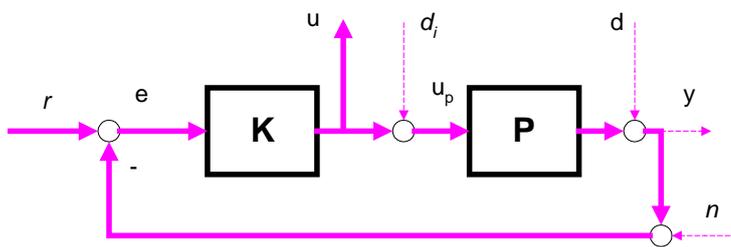


Figure 2-5: Control Sensitivity

Being:

$$\begin{aligned}
 u &= K (r - P u) \rightarrow \\
 (1 + KP) u &= K r \rightarrow \\
 u &= (1 + KP)^{-1} K r \rightarrow \\
 u &= S_i K r
 \end{aligned}$$

Or also having into account that mathematically $S_i K = K S_o$,

$$u = K S_o r$$

It is interesting that the control signal is as small as possible for avoiding actuators saturation and saving energy. This means that $K S_o$ shall be small at low frequencies and big a high frequencies to limit the actuating effort.

2.1.1.4 Sensitivity at the plant input ($S_o P$)

It is the TF from the perturbations at plant input d_i to the outputs y (Figure 2-6).

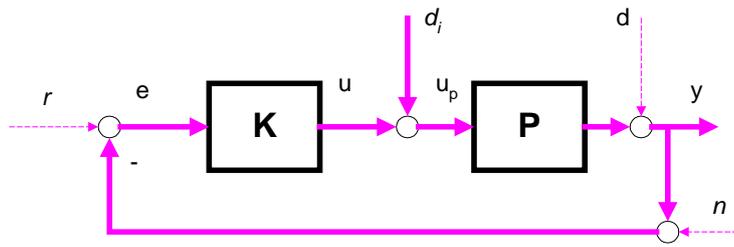


Figure 2-6: Sensitivity at plant input

This TF is equal to:

$$y = (1 + PK)^{-1} P d_i = S_o P d_i$$

Note that mathematically:

$$S_o P = P S_i$$

Traditionally in the literature this transfer function has not given the appropriate attention, supposing that the perturbations at plant inputs are negligible. Specifically for ill conditioned plants this approach is erroneous and the consideration of this transfer function is of paramount importance, as will be demonstrated in chapter 4 of this thesis.

2.1.2 Summary of tradeoffs in feedback control

The following relations hold in the feedback configuration introduced previously. It has been supposed that there is no measurement noise n .

- (1) $y = T_o r + S_o P d_i + S_o d$
- (2) $e = r - y = S_o(r - d) - S_o P d_i$
- (3) $u = K S_o r - K S_o d - T_i d_i$
- (4) $u_p = K S_o r - K S_o d + S_i d_i$

Equations (1) and (2) say that for obtaining a good tracking (i.e. $y \approx r$) is needed that at low frequency (Figure 2-7):

- $T_o \approx 1$
- S_o shall be small in order that perturbations at plant output have little influence in the output and the error is small (disturbance rejection at plant output)
- $S_o P$ shall be small in order that perturbations at plant input have little influence in the output and the error is small

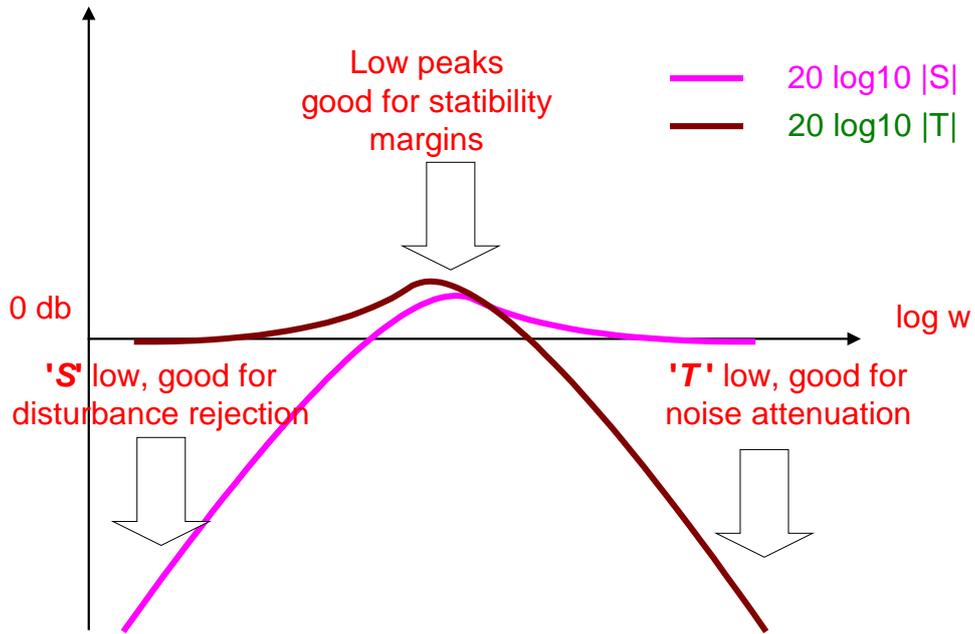


Figure 2-7: S and T trade-offs

Equations (3) y (4) says that in order the control signal to be small

- S_i shall be small
- KS_o shall be small

We have already seen that as:

$$S_o = 1 / (1 + PK)$$

S_o small at low frequencies implies $PK \gg 1$ at such low frequencies. Also, as

$$T_o = PK / (1 + PK)$$

T_o small at high frequencies implies $PK \ll 0$ at such high frequencies.

Let's define as the loop gain, L_o or its complementary L_i as (Figure 2-8):

$$L_o = PK$$

$$L_i = KP$$

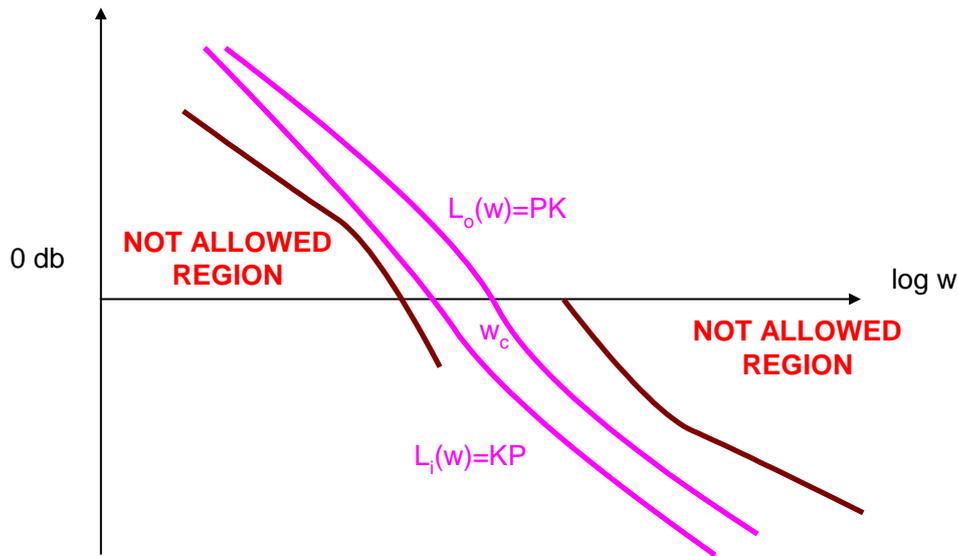


Figure 2-8: Open loop gain and weights

The gain crossover frequency, ω_c , is defined as the frequency where $|L(j\omega)|$ first crosses 1 from above. This is one definition of the open-loop bandwidth.

The phase crossover frequency, ω_{180} , is defined as the frequency where the Nyquist curve of $L(j\omega)$ first crosses the negative real axis.

2.1.3 Controllability and observability

Controllability

A system is controllable if starting from any initial state x_0 , the system can be driven to any final state x_F in finite time by applying the appropriate input u . The pair (A, B) is said to be controllable.

The following statements are equivalent. Refer to (Zhou, 1995) or (Datta, 2004) for a proof.

- (A, B) is controllable
- The controllability matrix $C = \begin{bmatrix} B & AB & A^2B & A^{n-1}B \end{bmatrix}$ has full row rank
- The matrix $[A - \lambda I, B]$ has full row rank for all λ in \mathcal{C} .
- The eigenvalues of $A + BF$ can be freely assigned

The controllability matrix is built as follows:

$$x' = Ax + Bu$$

taken $x(0) = 0$

$$x(t) = A^0 x(0) + \int_0^t A^{t-\tau} B u(\tau) d\tau = \int_0^t A^{t-\tau} B u(\tau) d\tau$$

$$x(1) = A x(0) + Bu = ABu + Bu$$
$$x(2) = A x(1) + Bu = A^2Bu + ABu + Bu$$

So is clear that the state is a combination of the inputs of the previous instants. If one of the steps is null, the following steps will remain on that state and not all the states will be reachable.

Stable systems and Stability

An unforced system $\dot{x} = Ax$ is stable if all the eigenvalues of A are in the left half plane. A is said to be Hurwitz.

The pair $[A, B]$ is said to be stabilizable if exist a feedback state law $u = Fx$ such that the system $[A + BF]$ is stable.

Observability

A system is said to be observable if the initial state x_0 , can be uniquely determined from the knowledge of $u(t)$ and $y(t)$ between t_0 and t_f . In other words, it is possible to determine all the states only from the inputs and outputs.

The concepts of observability and controllability are dual.

The following statements are equivalent. Refer to (Zhou, 1995) or (Datta, 2004) for a proof.

- (C, A) is observable
- The observability matrix $O = [C \quad CB \quad C^2B \quad \dots \quad C^{n-1}B]^T$ has full column rank
- The matrix $[A - \lambda I, C]^T$ has full row rank for all λ in \mathcal{C} .
- The eigenvalues of $A + LC$ can be freely assigned

Detectability

The pair $[C, A]$ is said to be detectable if $[A + LC]$ is stable for some L .

The following statements are equivalent. Refer to (Zhou, 1995) or (Datta, 2004) for a proof.

- (C, A) is detectable
- The matrix $[A - \lambda I, C]^T$ has full row rank for all λ in \mathcal{C} .
- Exist a L such that $A + LC$ is Hurwitz
- $[A^T, C^T]$ is stabilizable

2.1.4 State feedback and output injection

Given the system

$$\begin{aligned}x' &= Ax + Bu \\y &= Cx + Du\end{aligned}$$

Imposing a “state feedback” control law

$$u = Fx + v$$

The closed loop system becomes:

$$\begin{aligned}x' &= (A + BF)x + Bv \\y &= (Cx + DF)x + Dv\end{aligned}$$

The state feedback does not change the controllability, however it changes the observability. By state feedback the poles of the closed loop system can be assigned freely (but the original system shall be controllable).

The dual operation of the “state feedback” is the “output injection”

$$u = Ly$$

The closed loop system becomes:

$$\begin{aligned}x' &= Ax + BLy + Bv \\y &= Cx + DLy + Dv\end{aligned}$$

By duality the “output injection” does not change the observability, however it changes the controllability, i.e. we can make an uncontrollable system controllable with “output feedback”.

2.1.5 Observers and Observers-Based Controllers

In real control problems not all the states are available for feedback. The designer has available only some states plus the inputs and the outputs.

Given a system:

$$\begin{aligned}x' &= Ax + Bu \\y &= Cx + Du\end{aligned}$$

An observer system estimates the state of the previous system and is given by:

$$\begin{aligned}q' &= Mq + Nu + Hy \\ \hat{x} &= Qq + Ru + Sy\end{aligned}$$

i.e. the vector q is the state of the observer. The observer output is \hat{x} , i.e. the estimated state. The observer takes as inputs the control inputs and outputs of the estimated system.

A “Luenberger observer” (Figure 2-9) is a variation of previous described by the equations:

$$q' = Aq + Bu + L(Cq + Du - y)$$

$$\hat{x} = q$$

That can be interpreted as a mimic of the original system ($q' = Aq + Bu$) plus a weighted term that depends on the output error (difference between the output estimation $\hat{y} = Cq + Du$ and the real output y).

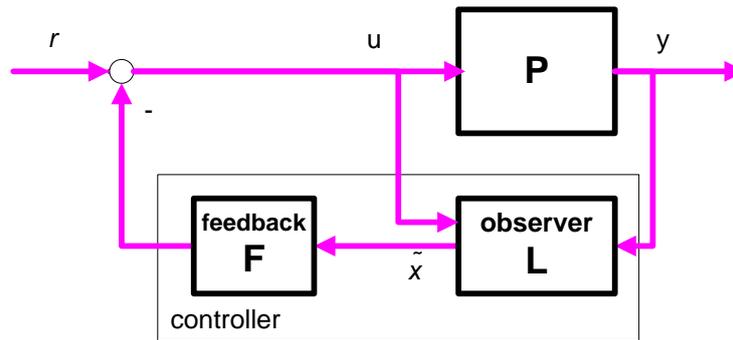


Figure 2-9: Luenberger observer

Let’s calculate the combined “observer –state feedback” for the system:

$$x' = Ax + Bu$$

$$y = Cx + Du$$

Defining the error e and the state feedback u :

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

$$u = F\hat{x}$$

We have:

$$x' = Ax + Bu = Ax + B(F\hat{x})$$

$$x' = Ax + BF\hat{x}$$

$$x' = Ax + BF(x - e)$$

$$x' = (A + BF)x - BFe$$

And

$$\begin{aligned}
 e &= x - \hat{x}; \\
 e' &= x' - \hat{x}'; \\
 e' &= (Ax + Bu) - (A\hat{x} + Bu + L(C\hat{x} + Du - y)) \\
 e' &= (Ax + Bu) - (A\hat{x} + Bu + L(C\hat{x} + Du - Cx - Du)) \\
 e' &= Ax - A\hat{x} - L(C\hat{x} - Cx) \\
 e' &= A(x - \hat{x}) + LC(x - \hat{x}) \\
 e' &= Ae + LCe
 \end{aligned}$$

The closed loop of the system with the “observer - state feedback” controller is given by:

$$\begin{bmatrix} x' \\ e' \end{bmatrix} = \begin{bmatrix} A + BF & -BF \\ 0 & A + LC \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix}$$

The poles of the closed loop are made by two parts: the poles coming from the state feedback ($A+BF$) and the poles coming from the observer ($A+LC$).

The “observer - state feedback” controller is given by:

$$\begin{bmatrix} \hat{x}' \\ u \end{bmatrix} = \left[\begin{array}{c|c} \frac{A + BF + LC + LDF}{F} & \frac{-L}{0} \end{array} \right] \begin{bmatrix} \hat{x} \\ y \end{bmatrix}$$

The “Luenberger observer” has been introduced here with some detail because it will be shown later that H-Infinity controller has a similar structure.

2.1.6 Fundamental limitations on feedback systems

2.1.6.1 Fundamental limitations on S and T

First limitation has been mentioned previously: as $PK \rightarrow 0$ when $\omega \rightarrow \infty$ for any physical system, from the definition of S and T we have:

- $S \rightarrow 0$ at low frequencies
- $T \rightarrow 0$ at high frequencies

As mathematically $S + T = 1$, we cannot do both S and T small at a given frequency.

2.1.6.2 Bode integral formula or the ‘waterbed effect’

The following formula is due to Bode:

$$\int_0^{\infty} \ln|S(j\omega)d\omega| = \pi \sum_{i=1}^m \text{Re}(p_i)$$

The area of the modulus of S is constant: it is zero for stable systems and it is a multiple of the sum of the real part of the unstable poles for unstable systems. This means that

increasing S in some frequency range motivates that S decreases in some other frequency range. This also means that unstable poles increase the area of $|S|$ to be greater than 1.

This is called the ‘waterbed effect’ (Figure 2-10). Decreasing S in a range of frequencies motivates S to increase in another range of frequencies.

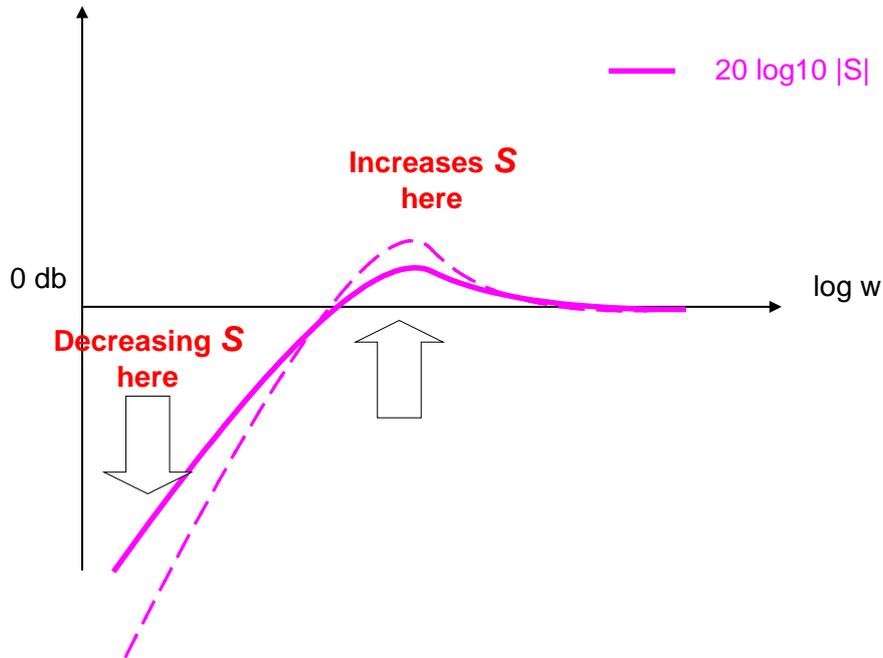


Figure 2-10: The waterbed effect

The previous restriction applies to any physical plant.

2.1.6.3 Fundamental limitations on S and T due to RHP poles and zeros

Maximum modulus theorem

For a stable function (i.e. it has no RHP poles), the maximum modulus over the imaginary axis is equal to the maximum modulus over the RHP. This follows from the special properties of complex functions (see demonstration in appendix 8.2.1).

Value of S and T at RHP

If P has a RHP zero (z_0) and K is a stabilizing controller (so no zero pole cancellations are present) we have that the sensitivity S is equal to 1 at the zero, because

$$S = 1 / (1+PK) \quad \text{and } P(z_0) = 0, \text{ so}$$

$$S(z_0) = 1 / (1+P(z_0) K) = 1$$

If P has a RHP pole (p_0), and K is a stabilizing controller (so no zero pole cancellations are present) we have that the complementary sensitivity T is equal to 1 at the pole, because:

$$S = 1 / (1+PK) \quad \text{and } P(p_0) = \infty, \text{ so}$$
$$S(p_0) = 1 / (1+P(p_0) K) = 0$$

But as $S + T = 1$, we shall have $T(p_0) = 1$

As summary, assuming a stabilizing controller K for a plant P that have a RHP zero (z_0) and pole (p_0):

$$S(z_0) = 1 \text{ and } S(p_0) = 0$$
$$T(p_0) = 1 \text{ and } T(z_0) = 0$$

Given a stable weight W_S , applying the maximum modulus theorem, we have that having $\max(W_S S)$ on the imaginary axis is equal to $\max(W_S S)$ on all the RHP.

In particular it will be greater or equal to the value of W_S in z_0 , but as $S(z_0) = 1$, we have $\max(W_S S) > W_S(z_0)$, i.e. the magnitude of $(W_S S)$ will be always greater than a fixed value $W_S(z_0)$. It is not possible to decrease the Sensitivity of the closed loop below this limit.

Given a stable weight W_T , applying the maximum modulus theorem, we have that having $\max(W_T T)$ on imaginary axis is equal to $\max(W_T T)$ on all the RHP.

In particular it will be greater or equal to the value of W_T in p_0 , but as $T(p_0) = 1$, we have $\max(W_T T) > W_T(p_0)$, i.e. the magnitude of $(W_T T)$ will be always greater than a fixed value $W_T(p_0)$. It is not possible to decrease the Complementary Sensitivity of the closed loop below this limit.

2.2 The formulation of the Robust Control Problem

The following goals are defined:

- **Nominal stability:** the nominal closed loop plant is stable
- **Nominal Performance:** the nominal closed loop plant fulfills the specifications
- **Robust stability:** all the closed loop plants (nominal and disturbed) are stable
- **Robust Performance:** all the closed loop plants (nominal and disturbed) fulfills the specifications

It is necessary to introduce before the concepts of singular values, norms, weights and scaling.

2.2.1 The size of a transfer function: singular values and norms

We have shown that in order to achieve good control performances some transfer functions have to be 'small' in a given frequency range. We have shown that there are some intrinsic limitations on S and T and their shape. In order to formulate mathematically this concept of 'small in a given frequency range' or 'minor than' the concepts of singular values and norm have to be introduced.

2.2.1.1 Singular values

Any matrix of dimensions ($l \times m$) can be factorized as:

$$A = U E V^H$$

Where U and V are unitary (i.e. its inverse is equal to its conjugate transposed). The matrix E is a diagonal matrix with the singular values of A , where

$$\begin{aligned} \sigma_1 > \sigma_2 \dots > \sigma_k, \\ k = \min(l, m) \end{aligned}$$

Intuitively, the major singular value is the greatest gain that can be obtained for any combination of the inputs. The minor singular value is the smallest gain that can be obtained for any combination of the inputs. The largest input direction is given by σ_1 . The smallest input direction is given by σ_k . If we plot the Bode diagram for the gain for a MIMO system, the gain are always bounded by the major singular value and minor singular value.

It is useful to study the particular case of a 2×2 matrix because it has a physical visualization in terms of operations on a vector.

The singular value decomposition of $A = U E V^H$ can be interpreted as the concatenation of the following basic operations:

- Take the input vector and rotate it with the matrix rotation V
- Scale the resulting vector (a different factor scale is applied to each dimension, e.g. a circle would be transformed to an ellipse)
- Rotate again with the matrix rotation U

Graphically (Figure 2-11), if we take as input vectors 2 orthogonal unit vectors, the steps of the SVD are:

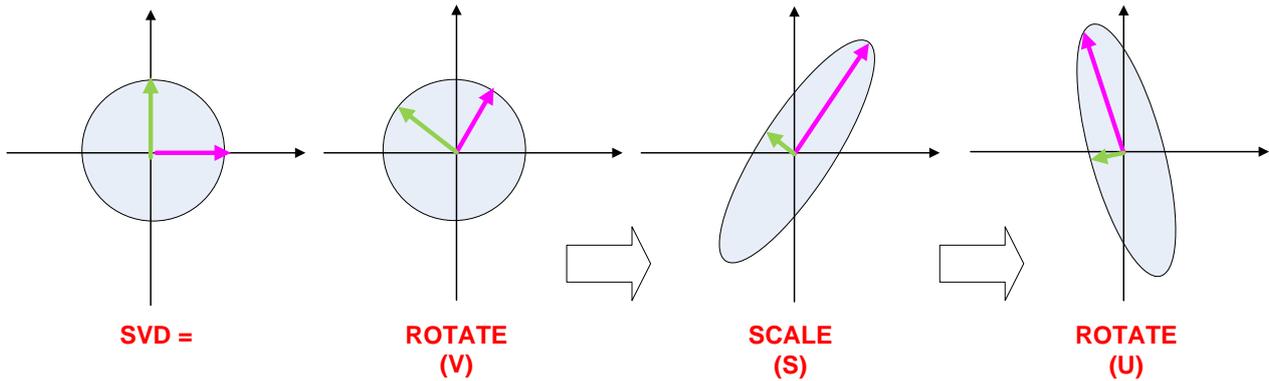


Figure 2-11: Visualization of a SVD in a 2x2 system

Connection of singular values with induced 2-norm of a matrix

The induced 2-norm (Euclidian norm) that will be introduced later is related to the maximum singular values.

$$\bar{\sigma}(A) = \max_{\|x\|=1} \|Ax\|_2$$

$$\underline{\sigma}(A) = \min_{\|x\|=1} \|Ax\|_2$$

Intuitively: just introduce a vector x such that its norm H_2 is 1 (i.e. lies in a unit circle). Apply the matrix A to it taking the SVD decomposition $A = U E V^H$.

$$\begin{aligned}
 Ax &= \\
 (U E V^H)x &= \\
 (U E) (V^Hx) &= \\
 (U E) y &\quad (\text{where 2-norm of } y \text{ is 1 as a rotation does not change the modulus})
 \end{aligned}$$

Now as y lies in the unit circle,

Let's suppose that we have taken the vector x such that $y = [1 \ 0]$

$$U \begin{bmatrix} \sigma_{\max} & 0 \\ 0 & \sigma_{\min} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = U \begin{bmatrix} \sigma_{\max} \\ 0 \end{bmatrix} \quad \text{but} \quad \left\| \begin{bmatrix} \sigma_{\max} \\ 0 \end{bmatrix} \right\|_2 = \sqrt{\sigma_{\max}^2 + 0^2} = \sigma_{\max}$$

The vector x is mapped to a vector y that lies on the major axis of the ellipse and suffers a scaling of σ_{max} . Any other input vector will lie in any other direction that the major axis of the ellipse so the scaling will be minor than σ_{max} (because in an ellipse the major axis is the one with maximum length). This demonstrates that $\bar{\sigma}(A) = \max_{\|x\|=1} \|Ax\|_2$ is true.

Connection of singular values with eigenvalues

Given the matrix M , we can perform the singular values decomposition:

$$M = UEV^T$$

And then multiply by M^T :

$$M^T M = (UEV^T)^T UEV^T = VEU^T UEV^T = VEEV^T = VE^2V^T$$

But this means

$$(M^T M)V = VE^2$$

I.e. the square of the singular values of M are the eigenvalues of $M^T M$.

2.2.1.2 Norms

A norm is a mathematical operation that assigns a scalar ('the norm') to each element of a space. In order to be a norm the operation shall observe the following properties:

- Non negative: $norm(v) \geq 0$
- Positive: $norm(v) = 0$ iff $v = 0$
- Homogeneity: $norm(a v) = a norm(v)$ for any scalar a
- Triangle inequality: $norm(e1 + e2) \leq norm(e1) + norm(e2)$

The appendix 8.1.2 defines in detail the different types of vector norms and how the concept of vector norm can be generalized to the concept of induced norm of a matrix, then to the concept of norm of a signal and finally to the concept norm of a system.

The H_∞ norm of a system $G(s)$ is defined as:

$$\| \cdot \|_\infty \equiv \sup_{w \in \mathbb{R}} \bar{\sigma}(G(iw)) \equiv \sup_{\text{Re}(s) > 0} \bar{\sigma}(G(s))$$

The following physical interpretations are useful:

- The H-Infinity norm is the maximum peak of the major singular value for all the frequencies or equivalently, the H-Infinity norm of a system is the maximum along frequency of the 2-norm at each frequency
- The H-Infinity norm is the maximum peak of the Bode magnitude for all the inputs directions and all the frequencies
- The H-Infinity norm is equal to the farthest point in the Nyquist diagram

Note that the definition of H-infinity norm for a system uses the singular values for defining the norm at each frequency value and then the H-infinity norm for taking the peak over all the frequencies.

A function belongs to the space RH_∞ if their H_∞ norm is finite. All the TF functions bounded in the RHP belongs to the RH_∞ space. The TF functions with poles in the RHP do not belong to the RH_∞ space. All the TF that are rational and proper (order of the denominator is higher or equal that order of the numerator) belong to the RH_∞ space.

It is very important to remark that the H_∞ norm is a 'worst case' norm along the frequency in opposition to the H_2 norm that is an 'average case norm'. This has important practical implications: let's suppose the transfer function of an actuator. The H_2 norm may be on average under the physical limits of actuation but we can have 'peaks' of actuation that goes over the allowed values so breaking the actuator. The H_∞ norm is a worst case norm.

The following property of the H_∞ norm is very useful:

- Being an induced norm, $\|AB\|_\infty \leq \|A\|_\infty \|B\|_\infty$

It is useful because we now that the norm of the overall composition (augmented system) will be bounded by the product of the norm of the parts.

2.2.2 Introduction of weights

Previously the H_∞ norm has been introduced as a tool for measuring the ‘size’ of a transfer function.

Let’s take for example S_o . Our requirement is that the magnitude of S_o is small (less than a value ‘ m ’) at low frequencies (e.g. frequencies minor than ω_o):

$$|S_o| < m, \text{ for } \omega < \omega_o$$

This is equivalent to require that the greater singular value of S_o is less than ‘ m ’ at such frequencies:

$$\bar{\sigma}(S_o) < m, \quad \omega < \omega_o$$

But is this the same that require:

$$\frac{1}{m} \bar{\sigma}(S_o) < 1, \quad \omega < \omega_o$$

We can substitute the scalar $1/m$ for a frequency dependent weight $W_S(s)$ so the limit is specific for each frequency, having

$$\bar{\sigma}(W_S S_o) \leq 1 \quad \omega \in R,$$

Or equivalently:

$$\|W_S S_o\|_\infty < 1$$

Following a similar strategy, require T_o to be small in a given frequency range is equivalent to require:

$$\|W_T T_o\|_\infty < 1$$

The literature e.g. (Skogestad, 1996) proposes as a typical weight the following:

$$W(s) = \frac{\frac{s}{M} + \omega_B}{s + \omega_B A}$$

Note that $W(0) \rightarrow 1/A$ and $W(\infty) \rightarrow 1/M$. The bandwidth frequency ω_B is chosen according to the specific problem. The values of M and A depends on the specific problem and the type of weight (see hereafter).

If more roll-off is needed the previous weight can be generalized as:

$$W(s) = \left(\frac{\frac{s}{(M)^{1/K} + \omega_B}}{s + \omega_B(A)^{1/K}} \right)^K$$

2.2.2.1 Weights for S_o

Particularized for weighting S_o , we need the weight to be a low pass filter to influence S_o at frequencies minor than ω_B . Supposing for example $\omega_B = 0.1$, $M=2$, $A=0.1$, the W_s weight should have the following shape (Figure 2-12):

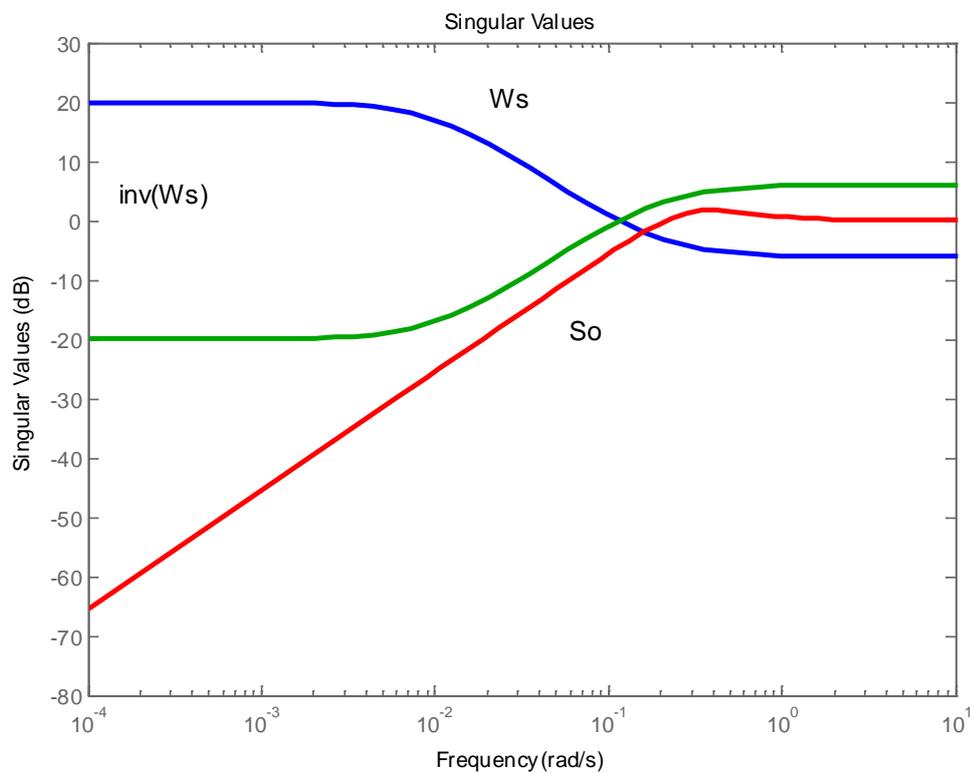


Figure 2-12: Typical shapes for S_o weights

If the control is satisfactory optimum we will see that the curves for S_o and $inv(W_s)$ are very similar at low frequencies.

Usually the performances are expressed on the time domain and not on the frequency domain. We introduce in our work a more easy approach for the W_s weight based on the shape of an ideal second order system used from time ago see for example (Aranda, 2000). Note: the use of a second ideal order system as “ideal model to match” is used often in generalized augmented plant schemas, but the direct use of the sensitivity of such ideal system as W_s was not found in the literature prior to the provided reference.

The proposal is:

- Choose a 2nd order system that meets the specifications (chose ω_n and δ)
 - o The TF of the system is: $T_{o_id} = \omega_n^2 / (s^2 + 2 \delta \omega_n s + \omega_n^2)$
- Calculate $S_{o_id} = 1 - T_{o_id}$
- Define the weight as the *inverse* of S_{o_id}

The weight defined is not proper. Depending the particular plant being controlled it can work or not. If a strictly proper weight is needed, we can add some zeros by simply calculate S_{o_id} as:

$$S_{o_id} = 1 + \epsilon - T_{o_id}, \text{ where } \epsilon \text{ is a small value}$$

2.2.2.2 Weights for T_o

Particularized for weighting T_o , we need the weight to be a high pass filter to limit T_o at frequencies greater than ω_B . Supposing for example $\omega_B = 2$, $M=0.1$, $A=2$, the W_S weight should have the following shape (Figure 2-13):

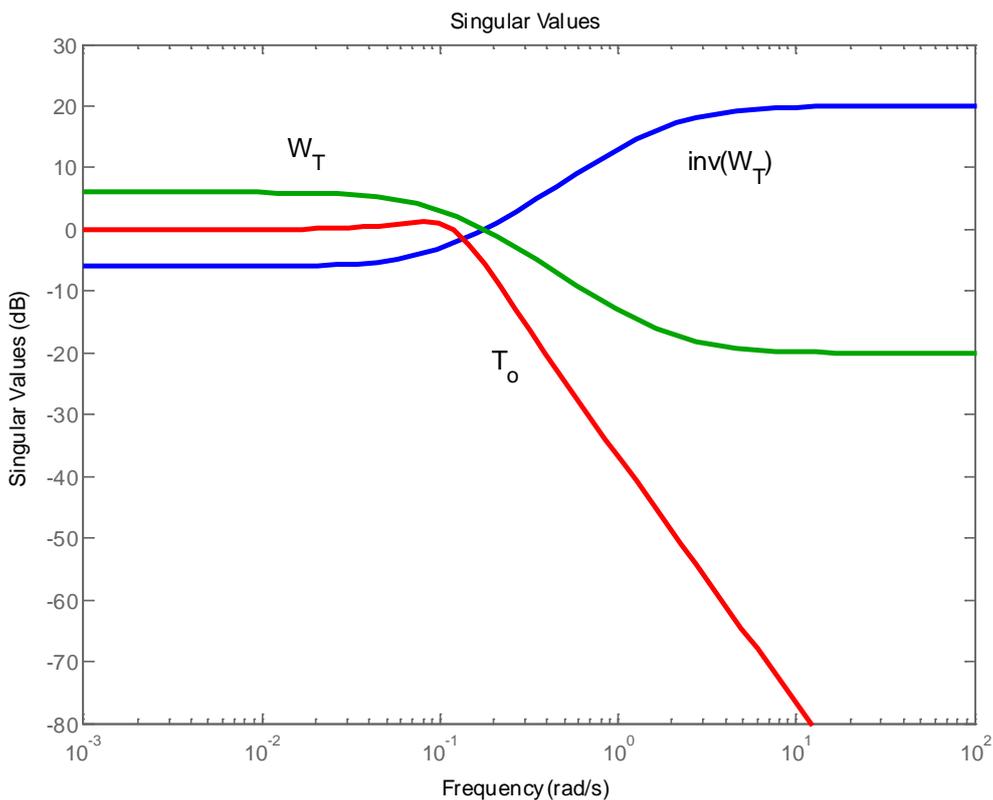


Figure 2-13: Typical shapes for T_o weights

The specifications for T_o are usually in the frequency domain, requiring roll-off at after a given frequency (for example for avoiding the coupling with bending modes of the plant). If the roll-off needs to be greater than 20 dB/decade we can use a higher order for the weight.

If the specifications for T_o are given in the temporal domain, we can use again the strategy of an “ideal 2nd order system”.

The proposal is:

- Choose a 2nd order system that meets the specifications (chose ω_n and delta)
 - o The TF of the system is: $T_{o_id} = \omega_n^2 / (s^2 + 2 \text{ delta } \omega_n s + \omega_n^2)$
- Define the weight as the *inverse* of T_{o_id}

The weight defined is not proper. Depending the particular plant being controlled it can work or not. If a strictly proper weight is needed, we can add some zeros.

It shall be noted that the ω_B frequency for W_S shall be minor than the ω_B frequency for W_T . This follows from the fact that the ‘not allowed regions’ on Figure 2-8: Open loop gain and weights shall not overlap.

2.2.2.3 Weights for K_S .

Particularized for weighting K_S , we need the weight to be a high pass filter limiting the actuation effort at frequencies greater than ω_B . Here the ω_B is defined by the physical capacities of the actuators. Usually the problem specifications define the actuator bandwidth and the maximum actuation gain allowed in the frequency domain.

Some other times the specifications for the actuators are defined in the time domain. For example, the thrust vector control for the nozzle of a rocket studied later has the following limits:

- Maximum deflection ± 7 degrees
- Maximum deflection rate 10 degrees/s

From previous, the actuator will complete a period (from zero position to -7 degrees, from -7 to +7 degrees and back to zero in $(7 \times 4)/10 = 2.8$ seconds. This means that the angular frequency is $\omega_{act} = 2\pi/T = 2.24 \text{ rad/s}$.

However this value cannot be used directly as bandwidth: we have to take into account the roll-off of the weight (20 dB/decade for a first order weight). If we want to conserve actuation up to this frequency, the gain shall be significant at least until ω_{act} . This means that the bandwidth shall be of the order of 1 decade higher, i.e. is $\omega_B = 10 * \omega_{act}$.

Then the maximum reference should require a deflection not greater than ± 7 degrees. This imposes the limit A for the weight at low frequencies. The value of M and the order of the weight depend on how much we want to limit the actuator effort at high frequencies. Practical experience shows that M should be some value near (and minor) than 1.

In our example $\omega_B = 22.4$, $A = 7 * \pi / 180$, $M = 0.9$ the W_U weight should have the following shape (Figure 2-14):

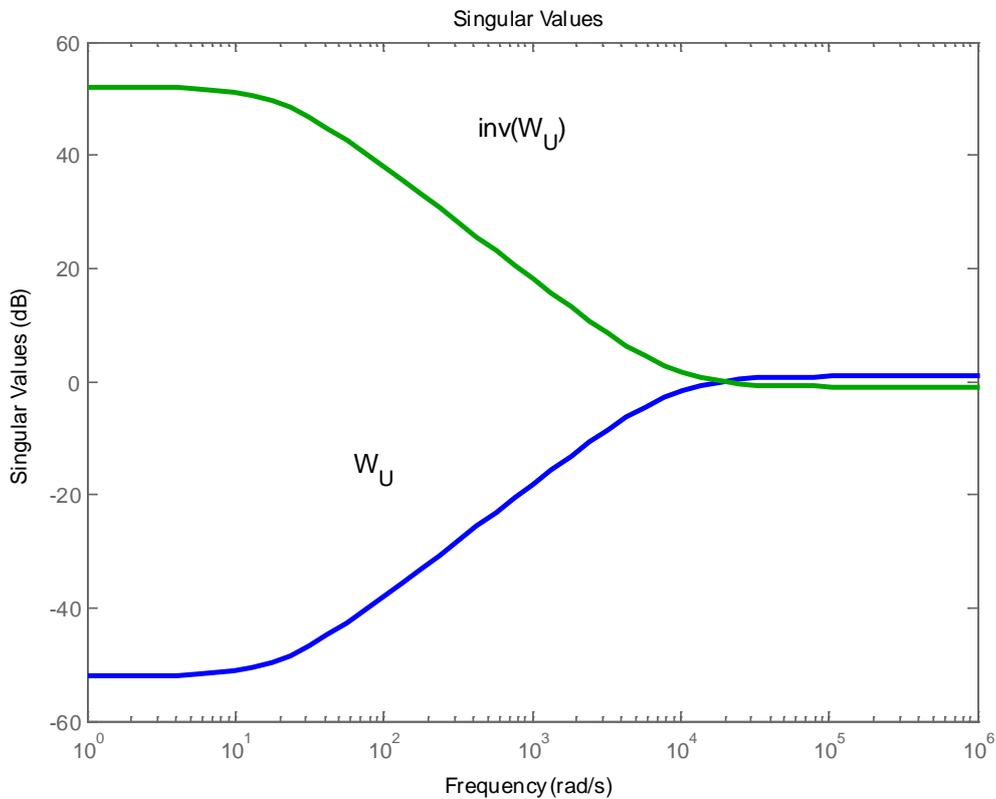


Figure 2-14: Typical form for KS_0 weights

2.2.2.4 Weights for S_oP

This transfer function (disturbances at plant input to the plant outputs) has been not given the appropriate importance in the literature. Weights for this transfer function will be studied in depth in chapter 4.

2.2.2.5 Some practical notes for weights

The experience of the author (in accordance with some cites in the literature) is that nothing more complex than a second or third order weight is useful. Second order weights are a good compromise because can be extracted directly of an equivalent ideal 2nd order system. Weights of higher order produce higher order controllers and often the H-infinity algorithm does not find solution.

To extend the selection of weights to MIMO problems the experience of the author shows that best way is to use diagonal weights. Each channel can have slight different weights but they should not be very different (the opposite would reveal poor scaling of the problem, see next paragraph).

2.2.3 The importance of scaling in MIMO

The H-Infinity theory (and a lot of other modern theories of control) is based in the minimization of the magnitude of some cost function. Usually this magnitude is the norm of a transfer function.

The concept behind scaling is to use reasonable and comparable inputs and outputs for all the transfer functions (i.e. use coherent units).

When we work with a MIMO problem an appropriate scaling is more important because not only the inputs and outputs but the different channels have to be comparable between them. The optimization algorithm will try to minimize (using the chosen norm) a cost function that is composed of the different terms. These terms shall be comparable between them: if one of the functions has much minor magnitude (norm) than the others, it simply does not contribute to the minimization process: the minimization would be driven just by the functions with the bigger magnitudes (norms).

The different types of scaling are analyzed hereafter.

Scaling 'input versus outputs':

This is the more easy to understand and just means to use equivalent units for inputs and outputs in a given channel. For example, in the SISO case of a voltage amplifier we should not use millivolts as inputs and volts as outputs.

It is always interesting to observe this scaling for coherence (using the same units in our problem). However in SISO the non-appropriate scaling is harmless (it simply changes the scale of our bode diagrams but does not influence the results). Things are different in MIMO.

Scaling between different channels:

Scaling between channels is equivalent to choose normalized units for all the channels. It is important in MIMO systems and even more important when we have coupling in the transfer matrix.

For example suppose we want to control the dynamics of an airplane. We have available as measures the angles of attack and the airspeed. The angles of attack will be expressed on radians and will have a magnitude of a few degrees (equivalent to about 0.1 rad). The airspeed will be expressed in meters and can have a magnitude of tenths of meters per second. If we have non diagonal terms in our transfer function we will be trying to optimize terms in AoA and terms in airspeed

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} AoA \\ airspeed \end{bmatrix}$$

The optimization algorithm will try to optimize the chosen norm (Norm-X) of something as:

$$Norm-X(G_{11} * AoA + G_{12} * airspeed) + NormX (G_{21} * AoA + G_{22} * airspeed)$$

As in non-scaled units:

$$airspeed \gg AoA,$$

The term to optimize will trend to be

$$NormX (G_{12} * airspeed) + NormX (G_{22} * airspeed)$$

... and as result the terms in AoA will be neglected.

We *shall* scale the units in order to apply any optimization method.

A general set of rules for scaling is explained in (Balas, 2012):

- All the inputs to the augmented system should have magnitude 1
- All the outputs of the augmented plant should have magnitude 1
- Each input scaling is such that a scaled input of magnitude 1 originates a typical (usually a maximum) not scaled input
- Each output scaling is such that a typical measured output originates a scaled measured output of 1
- See (Figure 2-15)

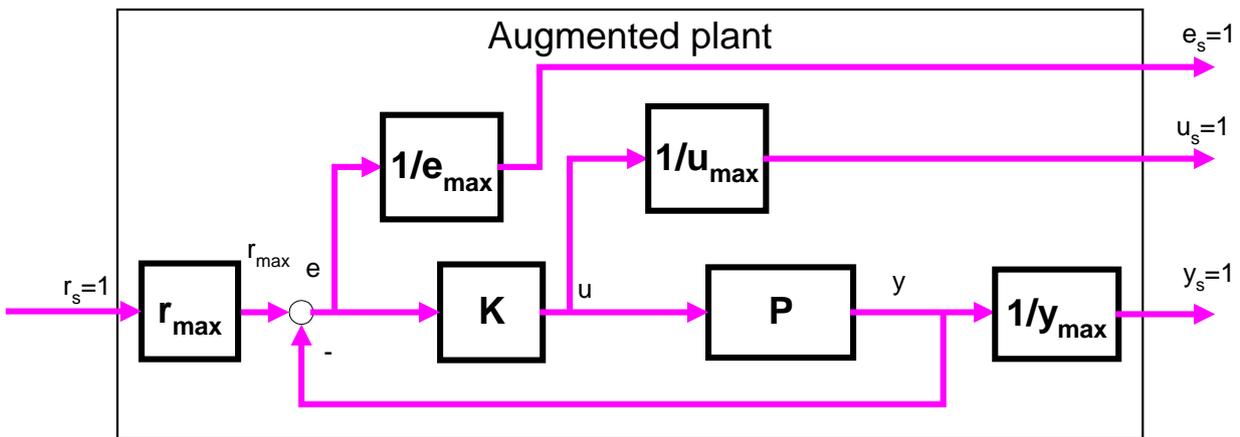


Figure 2-15: Augmented plant and scaling

Fig

Scaling of transfer functions

Another type of scaling is needed when the TF being optimized are a priori of different magnitude. This is not the case of the classical H-infinity mixed sensitivity S-T option

because in this case, S and T are by definition a magnitude similar to 1 in a range of frequencies and their maximums are also comparable.

We will see in chapter 4 that scaling S_oP is important when weighing it in H-Infinity problems together with S_o and T_o .

2.2.4 Robust stability and robust performance

The following goals were defined previously:

- **Nominal stability:** the nominal closed loop plant is stable
- **Nominal Performance:** the nominal closed loop plant fulfills the specifications
- **Robust stability:** all the closed loop plants (nominal and disturbed) are stable
- **Robust Performance:** all the closed loop plants (nominal and disturbed) fulfills the specifications

We already discussed that nominal performance is directly derived from S_o . Now we study the others objectives.

2.2.4.1 Internal Stability and Small Gain theorem

Internal stability

Let's suppose the following system (Figure 2-16)

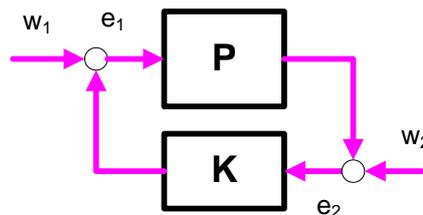


Figure 2-16: Internal stability

The equations are:

$$e_1 = w_1 + Ke_2$$

$$e_2 = w_2 + Pe_1$$

That can be written as:

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} (I - KP)^{-1} & (I - KP)^{-1}K \\ (I - PK)^{-1}P & (I - PK)^{-1} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

The system is well posed if all the transfer functions of the transfer matrix have inverse and are proper. The system is said to be internally stable if the closed loop system all the transfer matrices has a bounded H-infinity norm.

Internal stability guarantees that all the signals in the closed loop system are bounded. In other point of view: in presence of null external systems ($w = 0$), the state vector of the closed loop system goes to zero.

Small Gain Theorem

Let's assume M is internally stable and well-posed (Figure 2-17).

Then the system is stable for all disturbances $\|\Delta\|_\infty \leq 1$ *iff* $\|M\|_\infty \leq 1$.

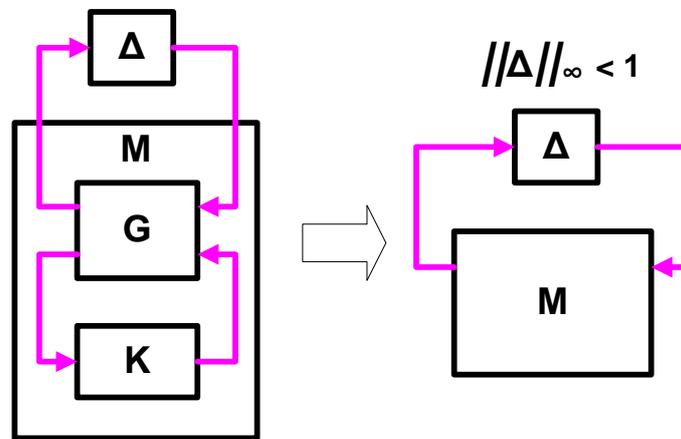


Figure 2-17: Set-up for the Small Gain theorem

This follows from the following: the LFT of the closed loop system is:

$$z = [M_{11} + M_{12} \Delta (I - M_{22} \Delta)^{-1} M_{21}] w$$

But we have required M is internally stable and well posed. This means that all the M_{ii} are stable. The only source of instability is $(I - M_{22} \Delta)$.

But

$$(I - M_{22} \Delta) \neq 0 \rightarrow \|M_{22} \Delta\|_\infty < 1$$

$$\|M_{22} \Delta\|_\infty < \|M_{22}\|_\infty \|\Delta\|_\infty$$

But we have required both $\|\Delta\|_\infty \leq 1$ and $\|M\|_\infty \leq 1$ that implies $\|M_{22}\|_\infty \leq 1$.

The interpretation of the small gain theorem is easy: if the norm of the closed loop is minor than 1, we stay out of the dangerous point $(-1,0)$ so no instability may occur.

2.2.4.2 Robust Stability

The objective of Robust Stability is defined as:

- All the closed loop plants (nominal and disturbed) are stable

The shape of T_o determines the robustness of the plant in presence of noise and modeling errors.

For demonstrating that, let's suppose that the stable plant P is affected by a multiplicative disturbance (Figure 2-18).

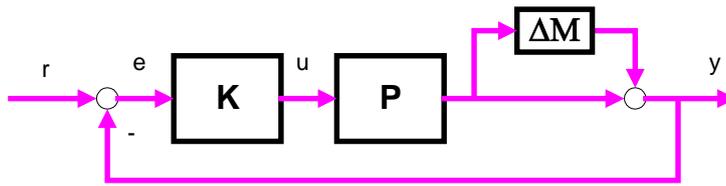


Figure 2-18: Multiplicative disturbance at output and stability

The system will be stable if the characteristic equation has no zeros on the right semi plane, i.e.

$$\det[I + (I + \Delta_M) PK] = \det[I + PK + \Delta_M PK] = \dots \quad \text{extracting the term } (I + PK)$$

$$\det[(I + PK)(I + \Delta_M PK(I + PK)^{-1})] =$$

$$\det[I + PK] \det[I + \Delta_M T_o]$$

But $(I + PK)$ is stable, as the nominal plant P is stable and the first determinant is always different from zero.

So, if $\|\Delta_M T_o\|_\infty < 1$, the whole determinant is always different from zero and the closed loop system is stable.

We see that T_o determines the stability margin against multiplicative disturbances at the output. Any disturbance with $\|\Delta_M\|_\infty < \|T_o^{-1}\|_\infty$ does not de-stabilizes the system.

We can reformulate the exigency ‘ T_o shall be small at high frequencies’ by introducing a frequency dependent weight and requiring

$$\|W_3 T_o\|_\infty < 1,$$

Where W_3 shall be a high pass filter for forcing T_o to be 1 at low frequencies and small at high frequencies.

The condition $\|W_3 T_o\|_\infty < 1$ can be expressed also as:

$$\|W_3PK(1+PK)^{-1}\|_\infty < 1$$

$$W_3PK \leq (1+PK)$$

That has a graphical interpretation in terms of a Nyquist diagram (Figure 2-19). The set of possible open loop gains (PK) at a given frequency is a disk centered in PK with radius W_3PK . In order to be nominally stable, the Nyquist plot shall encircle the point (-1) the correct number of times (zero if the plant as no RHP).

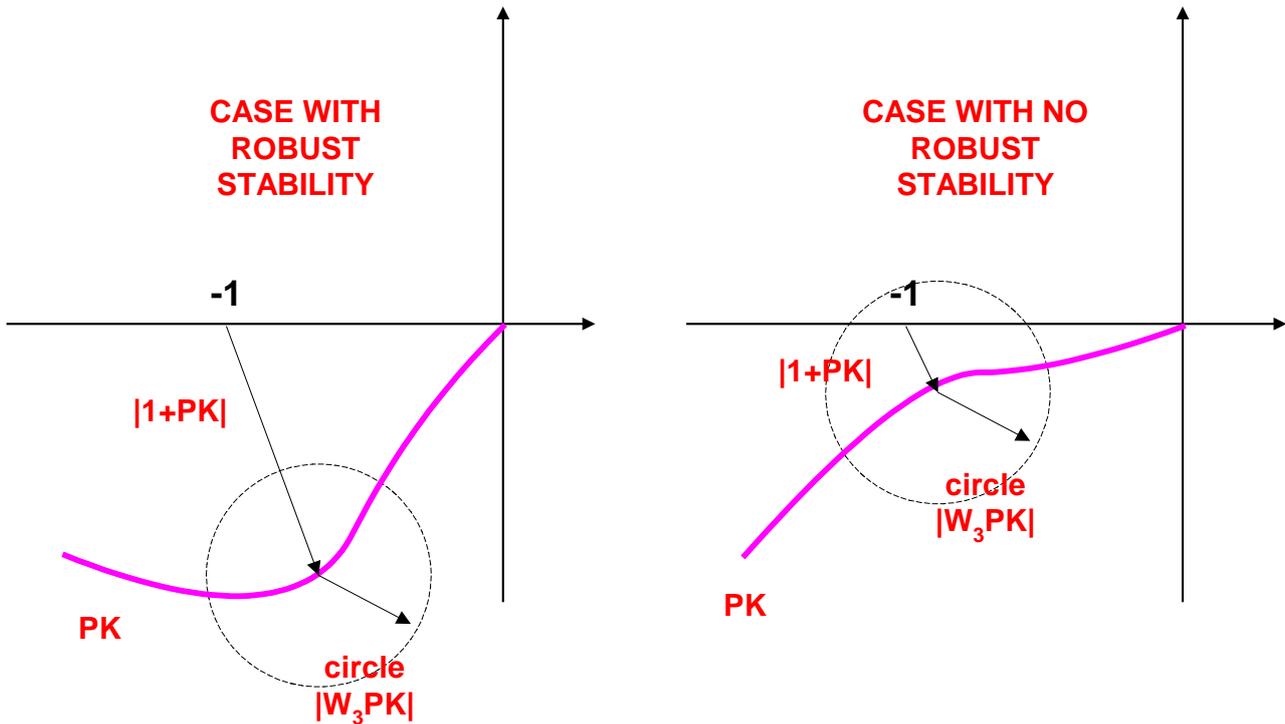


Figure 2-19: Robust stability

2.2.4.3 Robust Performance

The objective of Robust Performance is defined as:

- All the closed loop plants (nominal and disturbed) fulfills the specifications

The performance is expressed by the Output Sensitivity S_o . In presence of disturbances the robust performance criteria can be expressed as:

$$\|W_1(1+P(1+\Delta W_3)K)^{-1}\|_\infty < 1, \text{ with } |\Delta| < 1$$

Expressed in a Nyquist diagram (Figure 2-20) we have now 2 disks. One disk of radio W_1 centered at point (-1) measures the robustness. The other disk represents the uncertainty of the plant. The robust performance margin is the distance between both disks.

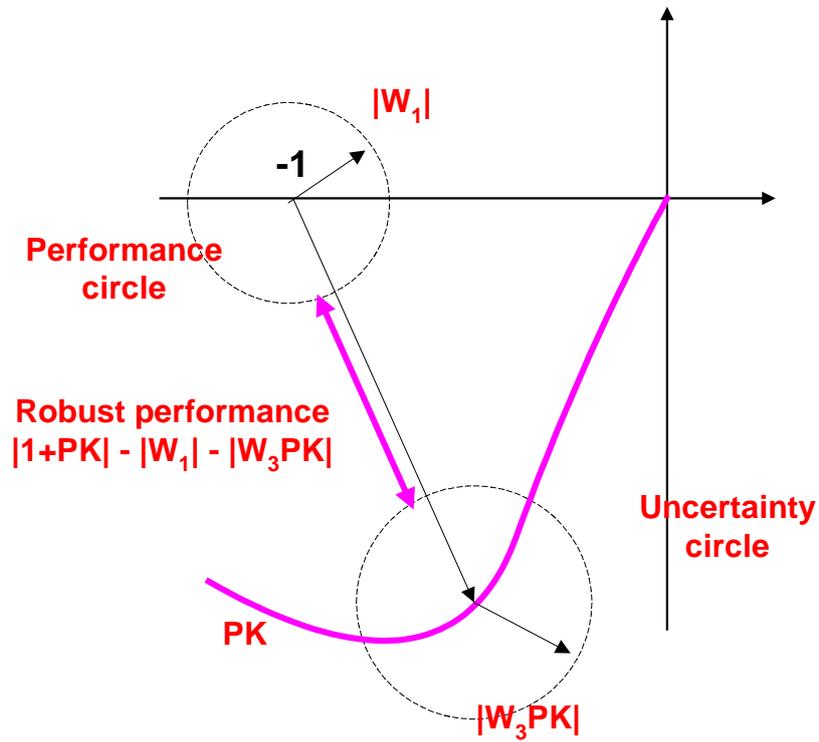


Figure 2-20: Robust performance

2.2.4.4 Summary

Nominal Performance: $\Leftrightarrow \|W_1 S_o\|_\infty < 1$

Robust Stability: $\Leftrightarrow \|W_3 T_o\|_\infty < 1$

Nominal Performance and Robust Stability:

$$\Leftrightarrow \max(\|W_1 S_o\|_\infty, \|W_3 T_o\|_\infty) < 1$$

Robust Performance: $\Leftrightarrow \max(\| |W_1 S_o| + |W_3 T_o| \|_\infty) < 1$

3 The H-Infinity Theory

3.1 General formulation of the control problem: the augmented plant

We are going to introduce a general formulation of the robust control problem by means of the concept of ‘augmented plant’. The ‘augmented plant’ contains into a single framework both the performance and robustness requirements of the robust control problem. We will show that both types of requirements can be expressed by means of weights and relations that the transfer functions of the augmented plant shall meet.

Let’s formulate the control problem in the following layout (Figure 3-1):

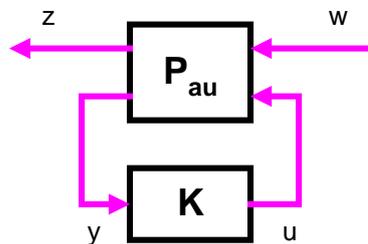


Figure 3-1: Robust control problem as LFT

Where

P_{au} : Augmented plant (including weights)

K : Controller

w : exogenous signals: references + disturbances

z : regulated (weighted) signals

y : measured signals (signals available to the controller)

u : control signals

Let’s partition P_{au} in dimensions compatible with w, u, z, y .

$p_1 = \dim(z)$ = number of weighted signals

$p_2 = \dim(y)$ = number of measured signals

$m_1 = \dim(w)$ = number of exogenous signals (references and disturbances)

$m_2 = \dim(u)$ = number of control signals (i.e. actuators)

So P_{11} ($p_1 \times m_1$), P_{12} ($p_1 \times m_2$), P_{21} ($p_2 \times m_1$), P_{22} ($p_2 \times m_2$)

The system equations are:

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$$

$$\begin{aligned} z &= P_{11} w + P_{12} u \\ y &= P_{21} u + P_{22} w \\ u &= K y \end{aligned}$$

From previous, we can express the transfer function from w to z as:

$$z = P_{11} + P_{12} K (I - P_{22} K)^{-1} P_{21} w$$

But this mathematical expression is a well mathematical expression known as a LFT (Linear Functional Transformation). Refer to the appendix 8.1.3 for a introducing to LFTs and it use in control.

The general control problem has been formulated as: find a controller that minimizes the norm of the TF from w to z .

3.2 Mathematical solutions to the H-infinity problem

3.2.1 The statement on the problem

From the previous discussions we deduce that the problem H-infinity is a general case of the general control problem, where we use the H-Infinity norm for minimization of the TF from w to z .

Let's take a representation in state space of the general control problem (Figure 3-2):

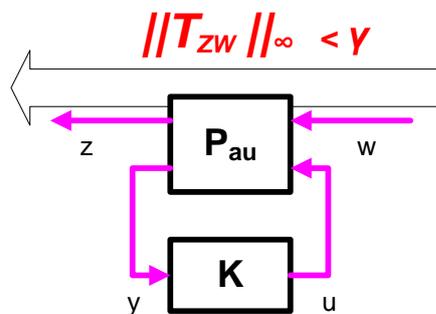


Figure 3-2: The H-infinity problem as LFT

$$\begin{bmatrix} x' \\ z \\ y \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}$$

Or alternatively, in 'input output' notation:

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \begin{bmatrix} A & B_w & B_u \\ C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}$$

The general problem of H-Infinity control is expressed as:

To find a controller that minimizes the H-infinity norm of the TF from w to z and stabilizes internally the system.

$$z = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}w$$

No exact solution has been found for this optimal control problem (i.e. to find the minimum). It is much easier to resolve the suboptimal problem:

To find a controller that guarantees that the H-infinity norm of the TF from w to z is minor that a given number and stabilizes internally the system.

$$\|T_{wz}\|_{\infty} < \gamma, \quad \gamma > 0$$

3.2.2 Some key concepts

Some important concepts have to be introduced before describing the solutions to the H-infinity problem. The methodology described in this work follows the LMI based formulation introduced by (Gahinet, 1994). This LMI formulation is clearer and more general than the “traditional” formulation based on Riccati equations.

The key concepts are the Hamiltonian function of a control system, the Bounded Real Lemma and the equivalence of Riccati and LMI formulations.

The strategy of this paragraph is as follows:

First, it is shown that the Hamiltonian function of a system describes the net energy flow of the system in terms of its system state equations and a cost function of its external inputs (w) and measured outputs (z).

Then it is demonstrated that the Hamiltonian function can be expressed as a LMI.

Then the Bounded Real lemma is introduced: it relates the H-infinity norm of a system with a LMI feasibility test on its Hamiltonian matrix (i.e. using the LMI formulation of the Hamiltonian). Finally, it is shown that the LMI test is equivalent to a Riccati equation test.

3.2.2.1 Hamiltonian function of a system

Let's be a system (not necessarily linear) described by:

$$\begin{aligned} x' &= f(x, w) \\ z &= g(x, w) \end{aligned}$$

Where x is the state, $z \in Z$ are the weighted signals, $w \in W$ are the exogenous signals.

The previous system is said to be *dissipative* with regard to a *supply function* $s(w, z)$ if exists a *storage function* $V(x)$ such that:

$$V(x(t_2)) \leq V(x(t_1)) + \int_{t_1}^{t_2} s(w(t), z(t)) dt$$

This 'storage function' is called the 'Hamiltonian' for analogy with the function of physics.

Note that:

$s : W \times Z \rightarrow R$ I.e. $s(w, z)$ assigns a real number (a cost) to each pair (w, z)

$V : X \rightarrow R$ I.e. $V(x)$ is a function that assigns a real number (a potential) to each state

The previous has a **very important physical interpretation**: $s(w, z)$ measures the energy provided to and extracted from the system. $V(x)$ measures the energy stored in the system. It is clear that the final storage $V(x(t_2))$ can be as maximum the initial energy $V(x(t_1))$ plus the net energy flow.

Then a step further: if the storage function V is differentiable, then the following inequality holds (simply by differentiating both terms in previous equation):

$$\frac{\partial V}{\partial t} = \frac{\partial V}{\partial x} \frac{\partial x}{\partial t} \leq s(w, z)$$

Substituting the expressions for x' and z :

$$\frac{\partial V}{\partial x} f(x, w) \leq s(w, g(x, w))$$

Now let's specialize the previous for a linear system:

$$\begin{aligned} x' &= f(x, w) = Ax + Bw \\ z &= g(x, w) = Cx + Dw \end{aligned}$$

And let's use a quadratic supply function:

$$s(w, z) = \begin{bmatrix} w & z \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix}$$

And a quadratic storage function:

$$V(x) = x^T Xx$$

Where X is some symmetric matrix (be aware that X is a matrix *not* the state x).

The general equation

$$\frac{\partial V}{\partial x} f(x, w) \leq s(w, g(x, w))$$

Is particularized as:

$$2x^T Xx' \leq \begin{bmatrix} w & z \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix}$$

And previous formula can be expressed as follows. (See demonstration in appendix 8.2.4).

$$\begin{bmatrix} x & w \end{bmatrix} \left(\begin{bmatrix} I & 0 \\ A & B \end{bmatrix}^T \begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ A & B \end{bmatrix} - \begin{bmatrix} 0 & I \\ C & D \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} 0 & I \\ C & D \end{bmatrix} \right) \begin{bmatrix} x \\ w \end{bmatrix} \prec 0$$

The previous expression is a Linear Matrix Inequality or LMI (see appendix 8.1.5 for an introduction to LMIs).

It is usual to express the previous LMI in the following 3 alternative forms:

By grouping terms:

$$F(X) = \begin{bmatrix} I & 0 \\ A & B \\ 0 & I \\ C & D \end{bmatrix}^T \begin{bmatrix} 0 & X & 0 & 0 \\ X & 0 & 0 & 0 \\ 0 & 0 & -Q & -S \\ 0 & 0 & -S^T & -R \end{bmatrix} \begin{bmatrix} I & 0 \\ A & B \\ 0 & I \\ C & D \end{bmatrix} \prec 0 \quad (\text{Eq 3-1})$$

By developing the product:

$$F(X) = \begin{bmatrix} A^T X + XA - C^T RC & XB - (SC)^T - C^T RD \\ B^T X - SC - D^T RC & -Q - SD - (SD)^T - D^T RD \end{bmatrix} \prec 0 \quad (\text{Eq 3-2})$$

That can be decomposed as:

$$F(X) = \begin{bmatrix} A^T X + XA & XB \\ B^T X & 0 \end{bmatrix} - \begin{bmatrix} 0 & I \\ C & D \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} 0 & I \\ C & D \end{bmatrix} < 0 \quad (\text{Eq 3-3})$$

3.2.2.2 The Bounded Real Lemma

The Bounded Real Lemma connects the Hamiltonian (expressed as an LMI) and the H-infinity norm of a system.

One of the ways of expressing the H-infinity norm is as the supreme of the induced 2-norm:

$$\|T_{zw}(s)\|_\infty \equiv \sup_{w \in R} \left\{ \frac{\|z\|_2}{\|w\|_2} : w \neq 0 \right\}$$

The minimization of the norm of T_{zw} can be expressed as achieving:

$$\|T_{zw}(s)\|_\infty < \gamma \quad \Rightarrow \sup \left\{ \frac{\|z\|_2}{\|w\|_2} \right\} < \gamma : w \neq 0$$

That can be expressed in term of a cost function:

$$s(w, z) = \|z\|_2^2 - \gamma^2 \|w\|_2^2 < 0$$

That can be expressed also as (dividing by minus gamma):

$$s(w, z) = -\frac{1}{\gamma} \|z\|_2^2 + \gamma \|w\|_2^2 < 0$$

But this cost function is simply a particularization of the general quadratic cost function:

$$s(w, z) = \begin{bmatrix} w & z \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix}$$

With:

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = \begin{bmatrix} \gamma I & 0 \\ 0 & -\frac{1}{\gamma} I \end{bmatrix}$$

So for this specific cost function we have an expression for calculating the H-infinity norm as a LMI:

$$F(X) = \begin{pmatrix} I & 0 \\ A & B \\ 0 & I \\ C & D \end{pmatrix}^T \left(\begin{array}{cc|cc} 0 & X & 0 & 0 \\ X & 0 & 0 & 0 \\ \hline 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & \gamma^{-1} I \end{array} \right) \begin{pmatrix} I & 0 \\ A & B \\ 0 & I \\ C & D \end{pmatrix} \prec 0 \quad (\text{Eq 3-4})$$

This is a crucial result: relates the H-infinity norm of a transfer function T_{zw} with a feasibility LMI test derived from its Hamiltonian matrix.

Alternative formulations of the LMI

Also using the particular cost function:

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = \begin{bmatrix} \gamma I & 0 \\ 0 & -\frac{1}{\gamma} I \end{bmatrix}$$

Substituting in the general formula for a Hamiltonian Matrix:

$$F(X) = \begin{bmatrix} A^T X + XA - C^T RC & XB - (SC)^T - C^T RD \\ B^T X - SC - D^T RC & -Q - SD - (SD)^T - D^T RD \end{bmatrix} \prec 0$$

We arrive to:

$$F(X) = \begin{bmatrix} A^T X + XA + C^T \gamma^{-1} C & XB + C^T \gamma^{-1} D \\ B^T X + D^T \gamma^{-1} C & + D^T \gamma^{-1} D - \gamma I \end{bmatrix} \prec 0$$

Can be also written as:

$$F(X) = \begin{bmatrix} A^T X + XA & XB \\ B^T X & -\gamma I \end{bmatrix} - (-\gamma^{-1}) \begin{bmatrix} C^T \\ D^T \end{bmatrix} \begin{bmatrix} C & D \end{bmatrix} \prec 0$$

Given the matrix M

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

The Schur complement of d in M is given by:

$$a - bd^{-1}c$$

(Refer to appendix 8.1.6 for an introduction to the Schur complement).

Applying the inverse of Schur complement to previous, i.e. identifying:

$$F(X) = \underbrace{\begin{bmatrix} A^T X + XA & XB \\ B^T X & -\mathcal{I} \end{bmatrix}}_a - \underbrace{\begin{bmatrix} C^T \\ D^T \end{bmatrix}}_b \underbrace{(-\gamma^{-1})}_{d^{-1}} \underbrace{\begin{bmatrix} C & D \end{bmatrix}}_c$$

A third way of formulating a LMI is found:

$$F(X) = \begin{bmatrix} A^T X + XA & XB & C^T \\ B^T X & -\mathcal{I} & D^T \\ C & D & -\mathcal{I} \end{bmatrix} \prec 0 \quad (\text{Eq 3-5})$$

3.2.2.3 Connection between the Hamiltonian and a Riccati equation

If we apply the Schur complement to the general formula

$$F(X) = \begin{pmatrix} A^T X + XA - C^T RC & XB - (SC)^T - C^T RD \\ B^T X - SC - D^T RC & -Q - SD - (SD)^T - D^T RD \end{pmatrix} \prec 0$$

We have:

$$d = -Q - SD - (SD)^T - D^T RD$$

$$F = a - bd^{-1}c \prec 0$$

$$F = (A^T X + XA - C^T RC) - (XB - (SC)^T - C^T RD) \left(-Q - SD - (SD)^T - D^T RD \right)^{-1} (B^T X - SC - D^T RC) \prec 0$$

If we particularize for the cost function:

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = \begin{bmatrix} \mathcal{I} & 0 \\ 0 & -\frac{1}{\gamma} I \end{bmatrix}$$

We have:

$$d = -\mathcal{I} - D^T \left(-\frac{1}{\gamma} \right) D$$

$$F = a - bd^{-1}c \prec 0$$

$$F = \left(A^T X + XA - C^T \left(-\frac{1}{\gamma} \right) C \right) - \left(XB - C^T \left(-\frac{1}{\gamma} \right) D \right) \left(-\mathcal{I} - D^T \left(-\frac{1}{\gamma} \right) D \right)^{-1} \left(B^T X - D^T \left(-\frac{1}{\gamma} \right) C \right) \prec 0$$

With $D = 0$ and $\gamma = 1$

$$F = (A^T X + XA + C^T C) - (XB)(-1)^{-1}(B^T X) \prec 0$$

$$F = (A^T X + XA + C^T C) + (XBB^T X) < 0$$

That is the Riccati equation used on the DGKF solution (discussed hereafter).

3.2.1 The DGKF solution

3.2.1.1 Preconditions

The following theorem was stated in (Doyle, 1989)

Let be the control problem formulated previously in paragraph 3.1. It is assumed that the following conditions hold:

(C1) (A, B_1) is stabilizable, and (C_1, A) is detectable. (A, B_2) is stabilizable, and (C_2, A) is detectable.

(This condition means that the plant does not contains uncontrollable of unobservable modes, i.e. that we can stabilize the system with state feedback and that an observer exists).

(C2) $\text{rank}(D_{12}) = n_u, \text{rank}(D_{21}) = n_y$

(rank(D₁₂) = n_u means that all the control signals are weighted and contribute to the calculated norm. rank(D₂₁) = n_y means and that all the external signals are observable for the controller).

(C3) The columns of $\begin{bmatrix} A - i\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix}$ are linearly independent for all ω , i.e. the column range of the matrix is $n_x + n_u$

(C4) The rows of $\begin{bmatrix} A - i\omega I & B_1 \\ C_2 & D_{21} \end{bmatrix}$ linearly independent for all ω , i.e. the row range of the matrix is $n_x + n_y$

(C3 and C4 mean that there are no $j\omega$ -axis zeros in T_{zu} or T_{yw} . The controller cannot cancel these zeros because the closed loop would not be internally stable).

For simplicity on the solution, the following conditions are also assumed:

(C5) $D_{11} = 0, D_{22} = 0$.

(C5 implies that T_{zw} and T_{yu} are strictly proper i.e. there is not direct feed through from $w \rightarrow z$ of from $u \rightarrow y$. This holds in almost all the real-life control problems because the effect of the inputs on the outputs is not instantaneous).

(C6) $D_{12} = \begin{bmatrix} 0 \\ I \end{bmatrix}, D_{21} = [0 \quad I]$

(Sometimes C5 and C6 are formulated together as $D_{12}^T D_{12} = D_{21}^T D_{21} = I$).

$$(C7) \quad D_{12}^T C_1 = 0 \quad \text{y} \quad B_1 D_{21}^T = 0$$

(It means that the weighted signal z shall be of the form $z = C_1 x + u$. I.e. a unit weight in the control input and there are no cross-term in x and u).

It shall be noted that there are different ways of expressing the previous conditions C1 to C4. Also, the conditions C5 to C7 can be removed at cost of complicating the expression of the resulting controller solution.

3.2.1.2 The suboptimal solution

If the previous conditions hold, then a controller K exists that stabilizes internally P and $\|T_{zw}\|_\infty < \gamma$ if the following conditions are hold:

(i) X_∞ is a semidefinite positive stabilizing solution of the Riccati equation:

$$A^T X_\infty + X_\infty A + C_1^T C_1 + X_\infty (\gamma^{-2} B_1 B_1^T - B_2 B_2^T) X_\infty = 0$$

Or equivalently, of the Hamiltonian:

$$H_\infty = \begin{bmatrix} A & \gamma^{-2} B_1 B_1^T - B_2 B_2^T \\ -C_1^T C_1 & -A^T \end{bmatrix}$$

(Refer to the appendix 8.2.5.2 for a demonstration of the equivalence between Riccati equations and Hamiltonian matrix).

(ii) Y_∞ is a semidefinite positive stabilizing solution of the Riccati equation:

$$A^T Y_\infty + Y_\infty A + B_1 B_1^T + Y_\infty (\gamma^{-2} C_1^T C_1 - C_2^T C_2) Y_\infty = 0$$

Or equivalently, of the Hamiltonian matrix:

$$J_\infty = \begin{bmatrix} A^T & \gamma^{-2} C_1^T C_1 - C_2^T C_2 \\ -B_1 B_1^T & -A \end{bmatrix}$$

(iii) $\rho(X_\infty Y_\infty) < \gamma^2$ (all the eigenvalues of $X_\infty Y_\infty$ have a value less than γ^2)

Then a suboptimal controller is given by:

$$K_{SUB} = \left[\begin{array}{c|c} A_\infty & -Z_\infty L_\infty \\ \hline F_\infty & 0 \end{array} \right]$$

Where:

$$\begin{aligned}
 F_\infty &= -B_2^T X_\infty \\
 L_\infty &= -Y_\infty C_2^T \\
 Z_\infty &= (I - \gamma^{-2} Y_\infty X_\infty)^{-1} \\
 A_\infty &= A + \gamma^{-2} B_1 B_1^T X_\infty + B_2 F_\infty + Z_\infty L_\infty C_2
 \end{aligned}$$

3.2.1.3 The family of suboptimal solutions

In fact, there is not only one controller but a set of controllers that satisfy the conditions. These are given by the following LFT expression:

$$K = F_l(K_c, Q)$$

I.e. an LFT of Q over a ‘central’ controller given by:

$$K_c = \left[\begin{array}{c|cc} A_\infty & -Z_\infty L_\infty & Z_\infty B_2 \\ \hline F_\infty & 0 & I \\ -C_2 & I & 0 \end{array} \right]$$

Where $Q(s)$ is any stable and proper FT such that $\|Q\|_\infty < \gamma$.

Generally the controller chosen is the “central” controller (i.e. $Q(s) = 0$).

A critic raised here, see for example (Gahinet, 1994) is that the choice of “ $Q(s) = 0$ ” is not really justified. Why the central controller should be the best controller? Also is not clear how variations of $Q(s)$ are linked to variation of properties of the closed loop system.

3.2.1.4 Comments on the DGKF solution

Developing the state equations for the ‘central’ suboptimal controller:

$$\begin{bmatrix} \hat{x}' \\ u \end{bmatrix} = \begin{bmatrix} A_\infty & -Z_\infty L_\infty \\ \hline F_\infty & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ y \end{bmatrix}$$

We have:

$$\begin{aligned}
 \hat{x}' &= A_\infty \hat{x} - Z_\infty L_\infty y \\
 \hat{x}' &= (A + \gamma^{-2} B_1 B_1^T X_\infty + B_2 F_\infty + Z_\infty L_\infty C_2) \hat{x} - Z_\infty L_\infty y \\
 \hat{x}' &= A \hat{x} + B_1 \hat{w}_{worst} + B_2 u + Z_\infty L_\infty (C_2 \hat{x} - y) \\
 u &= F_\infty \hat{x}
 \end{aligned}$$

Where has been defined:

$$\hat{w}_{worst} = \gamma^{-2} B_1^T X_\infty \hat{x}$$

W_{worst} can be interpreted as the worst case estimated disturbance.

The term $Z_\infty L_\infty (C_2 \hat{x} - y) = Z_\infty L_\infty (\hat{y} - y)$ is the optimal estimation of the error.

So the next estimated state of the controller is a contribution of:

- The current estimated state
- The current signal control u
- The worst disturbance
- The estimated error

By analysis of the diagram of the controller we find that is an “observer – state feedback” controller (Figure 3-3). I.e. the controller is made of two separate parts: an estimation of the state and a state feedback. This is very similar to the ‘Luenberger controller’ described in paragraph 2.1.5.

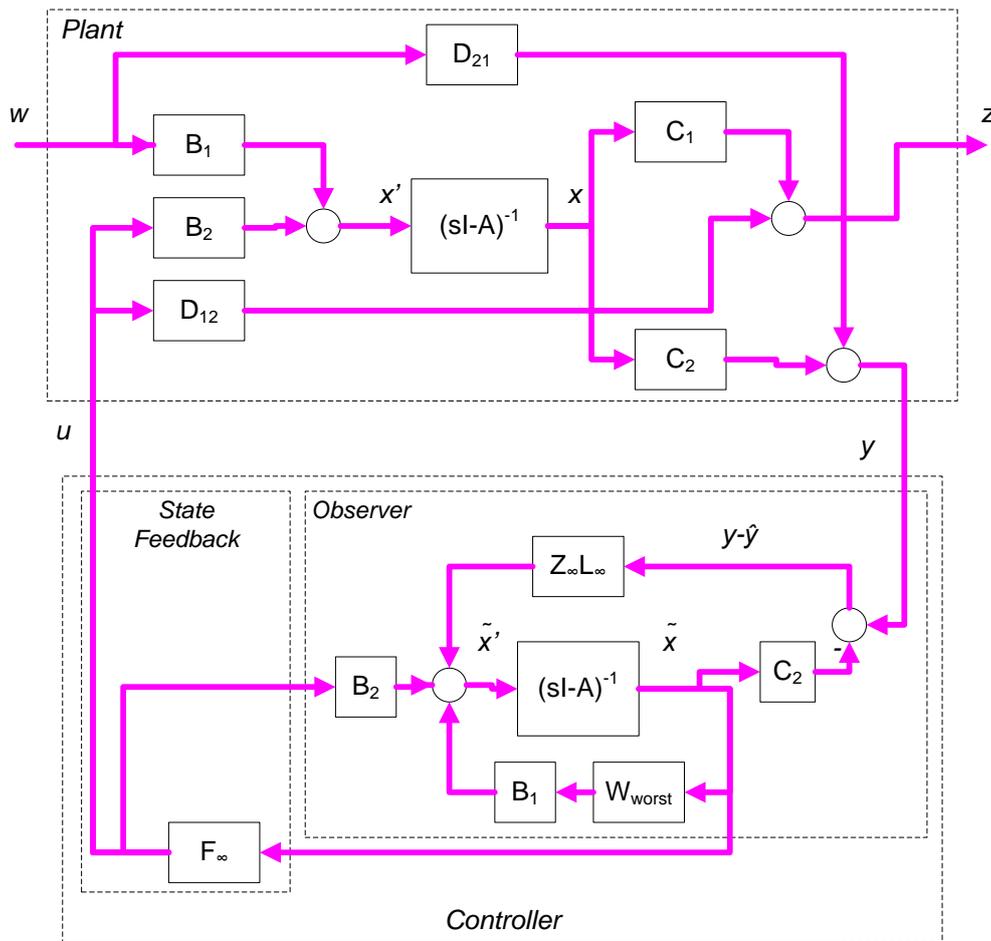


Figure 3-3: Separation structure of H-infinity controllers

The demonstration of the DGKF solution was presented in the famous “DGKF” article by Doyle, Glover, P. Khargonekar, and B. Francis, see (Doyle, 1989). A full development of the solution is included in (Zhou, 1995).

The solution is indeed very technical and does not provide tutorial value: on this thesis we prefer to develop the LMI solution to the H-infinity problem.

Just a sketch of the solution is commented. The previous structure of the controller (separation state feedback – observer) inspires the following strategy for the demonstration: first several special problems are introduced, the filtering problem (observer part) and the full information problem (state feedback part). The general solution is made by resolving each problem separately and combining them.

3.2.1 The LMI solution

The LMI solution was introduced by Gahinet and Apkarian (Gahinet, 1994).

Being stated the same H-Infinity problem that previously (paragraph 3.1), the following assumptions are done:

(C1) (A, B_2) is stabilizable, and (C_2, A) is detectable.

(This condition means that the plant does not contains uncontrollable of unobservable modes, i.e. that we can stabilize the system with state feedback and that an observer exists).

(C2) $(D_{22} = 0)$.

Note that assumptions C3 and C4 of the DGKF solution are not required here.

The problem is stated as:

To find a controller that guarantees that the H-infinity norm of the TF from w to z is minor that a given number and stabilizes internally the system.

$$\|T_{wz}\|_{\infty} < \gamma, \quad \gamma > 0$$

The “Bounded Real Lemma” has been introduced previously. It relates the problem of bounding the H-infinity norm of a system with the existence of a solution for a LMI.

$$\|T_{wz}\|_{\infty} < \gamma, \quad \gamma > 0, \text{ iff}$$
$$F(X) = \begin{bmatrix} A^T X + XA & XB & C^T \\ B^T X & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0, \quad X > 0$$

But on this formula $[A, B, C, D]$ is the state space representation of the closed loop system. The bounded real lemma defines a necessary condition that if satisfied guarantees the existence of a controller that fulfill the specifications. However the bounded real lemma does not defines a synthesis formula for the controller. It is needed to express the previous LMI in function of the open loop plant and the controller.

Before demonstrating the general case (that is quite complicated) the specific case of static state feedback control is demonstrated. It is quite simple and provides an introduction to the general case.

3.2.1.1 Specific case: static state feedback

A simple static state feedback controller is imposed:

- $y = x$ ($C_2 = I, D_{21}=0, D_{22}=0$)
- $u = K y$, where K is a constant feedback ($D_K=K_C, A_K, B_K, C_K=0$)

With these, the general formula for the state space representation of the closed loop:

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{cc|c} A + B_2 D_K C_2 & B_2 C_K & B_1 + B_2 D_K D_{21} \\ B_K C_2 & A_K & B_K D_{21} \\ \hline C_1 + D_{12} D_K C_2 & D_{12} C_K & D_{11} + D_{12} D_K D_{21} \end{array} \right] \quad (\text{Eq 3-6})$$

Is greatly simplified to:

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{c|c} A + B_2 K_C & B_1 \\ \hline C_1 + D_{12} K_C & D_{11} \end{array} \right] \quad (\text{Eq 3-7})$$

Applying the general LMI formula (Eq 3-5) to the closed loop state space expression (Eq 3-7) we have:

$$F(X) = \left[\begin{array}{ccc|c} (A + B_2 K_C)^T X + X(A + B_2 K_C) & X B_1 & (C_1 + D_{12} K_C)^T & \\ B_1^T X & -\mathcal{I} & D_{11}^T & \\ \hline C_1 + D_{12} K_C & D_{11} & -\mathcal{I} & \end{array} \right] \prec 0$$

That is no longer an LMI because there are non-linear terms with product of the incognita K_C and X . (Note also that exists terms only on K_C and only on X on so a simple change of variable $Y = K_C X$ will not work).

The strategy of the solution is to put back the previous in form of a LMI. In order to “re-linearize” two steps are applied:

Step 1: apply a congruence transformation T with:

$$T = \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \prec 0, \text{ being } Y = X^{-1}$$

The congruence transformation (refer to appendix 8.1.7) is a simply change of coordinates that does not change the “definiteness” of $F(X)$

$$G(Y) = T^T F(X) T \prec 0$$

$$G(Y) = \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}^T \begin{bmatrix} (A+B_2K_C)^T X + X(A+B_2K_C) & XB_1 & (C_1+D_{12}K_C)^T \\ B_1^T X & -\gamma I & D_{11}^T \\ C_1+D_{12}K_C & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \prec 0$$

Giving:

$$G(Y) = \begin{bmatrix} Y(A+B_2K_C)^T XY + YX(A+B_2K_C)Y & YXB_1 & Y(C_1+D_{12}K_C)^T \\ B_1^T XY & -\gamma I & D_{11}^T \\ (C_1+D_{12}K_C)Y & D_{11} & -\gamma I \end{bmatrix} \prec 0$$

$$G(Y) = \begin{bmatrix} Y(A+B_2K_C)^T + (A+B_2K_C)Y & B_1 & Y(C_1+D_{12}K_C)^T \\ B_1^T & -\gamma I & D_{11}^T \\ (C_1+D_{12}K_C)Y & D_{11} & -\gamma I \end{bmatrix} \prec 0$$

It shall be noted that $G(Y)$ has now terms on the product $(Y K_C)$, but has no terms only on Y or only on K_C .

Step 2: apply a change of variable $W=K_C Y$

$$G(Y, W) = \begin{bmatrix} (YA+B_2W)^T + AY + B_2W & B_1 & (C_1Y + D_{12}W)^T \\ B_1^T & -\gamma I & D_{11}^T \\ C_1Y + D_{12}W & D_{11} & -\gamma I \end{bmatrix} \prec 0, \text{ } Y \succ 0, \text{ } W \succ 0$$

That is a LMI in variables Y, W . Solving numerically the LMI for Y and W , we obtain the desired controller as:

$$K_C = WY^{-1}$$

3.2.1.2 General Case

The general case follows the same steps that the previous case but is much more complicated mathematically.

- Find the expression of the LMI in terms of the open loop plant and controller: in fact this generates two pseudo-LMIs
- Re-linearize the LMI (apply a congruent transformation plus a change of variable)
- Reconstruct the controller

3.2.1.2.1 Finding the LMI in terms of open loop plant and controller

Assuming state space representations for the open loop plant (the plant augmented with appropriate weights) and the controller:

$$P = \begin{bmatrix} A & B \\ C & D \end{bmatrix}; \quad K = \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}$$

The closed loop transfer functions can be expressed as:

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{cc|c} A + B_2 D_K C_2 & B_2 C_K & B_1 + B_2 D_K D_{21} \\ B_K C_2 & A_K & B_K D_{21} \\ \hline C_1 + D_{12} D_K C_2 & D_{12} C_K & D_{11} + D_{12} D_K D_{21} \end{array} \right]$$

The following notation is defined:

$$\begin{aligned} \bar{A} &= \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}, & \bar{B}_1 &= \begin{bmatrix} B_1 \\ 0 \end{bmatrix}, & \bar{B}_2 &= \begin{bmatrix} 0 & B_2 \\ I & 0 \end{bmatrix} \\ \bar{C}_1 &= \begin{bmatrix} C_1 & 0 \end{bmatrix}, & \bar{C}_2 &= \begin{bmatrix} 0 & I \\ C_2 & 0 \end{bmatrix} \\ \bar{D}_{11} &= D_{11}, & \bar{D}_{21} &= \begin{bmatrix} 0 \\ D_{21} \end{bmatrix}, & \bar{D}_{12} &= \begin{bmatrix} 0 & D_{12} \end{bmatrix} \end{aligned}$$

Note: It shall be remarked that the new matrices $\bar{A}, \bar{B}_1, \dots$ depend only on the open loop plant and not in the controller.

With previous notation the closed loop transfer function can be expressed as:

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{cc|c} \bar{A} + \bar{B}_2 K \bar{C}_2 & \bar{B}_1 + \bar{B}_2 K \bar{D}_{21} \\ \hline \bar{C}_1 + \bar{D}_{12} K \bar{C}_2 & \bar{D}_{11} + \bar{D}_{12} K \bar{D}_{21} \end{array} \right] \quad (\text{Eq 3-8})$$

The expressions for (A_L, B_L, C_L, D_L) shall be substituted by the expressions using the open loop plant and controller on the LMI general formula (Eq 3-5).

When this is done, the result is no longer a LMI, because we have multiplications of two incognita X and K , for example for the term (1, 1) of the LMI we have:

$$A^T X + XA = A_{CL}^T X + XA_{CL} = (\bar{A} + \bar{B}_2 K \bar{C}_2)^T X + X(\bar{A} + \bar{B}_2 K \bar{C}_2)$$

A change of variables is defined as follows:

$$P_X = \begin{bmatrix} \bar{B}_2^T X & 0 & \bar{D}_{12}^T \\ \bar{C}_2 & \bar{D}_{21} & 0 \end{bmatrix}$$

Note: in notation input/output,

$$P_X = \begin{bmatrix} \bar{B}_u^T X & 0 & \bar{D}_{zu} \\ \bar{C}_y & \bar{D}_{yw} & 0 \end{bmatrix}$$

P_X is the direct influence of the control signal in the state and in the weighted signals. Q is the contribution to the output of the state and the exogenous signals

Then writing the LMI in function of $\bar{A}, \bar{B}_1 \dots$ variables

$$H(X) = \begin{bmatrix} \bar{A}^T X + X\bar{A} & X\bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X & -\mathcal{A} & \bar{D}_{11}^T \\ \bar{C}_1 & \bar{D}_{11} & -\mathcal{A} \end{bmatrix}$$

The LMI $F(X)$ can be written as a more complex LMI:

$$F(X) = H(X) + Q^T K^T P_X + P_X^T K Q < 0 \quad (\text{Eq 3-9})$$

The demonstration is just long, tedious calculus. Refer to appendix 8.2.6 for a demonstration.

3.2.1.2.2 Finding the LMI for synthesis of the controller

The previous equation (Eq 3-9) contains 2 unknown variables: P_X (that depends of X) and K that is the state space expression of the controller.

$$F(X) = H(X) + Q^T K^T P_X + P_X^T K Q < 0$$

But by the ‘‘Elimination Lemma’’ (refer to appendix 8.2.7 for a discussion and demonstration) the previous equation has a solution on K iff:

$$W_{P_X}^T H(X) W_{P_X} < 0$$

$$W_Q^T H(X) W_Q < 0$$

Where:

- W_{P_x} is the orthogonal operator of a basis of the kernel of P_x .
- W_Q is the orthogonal operator of a basis of the kernel of P_x .

The elimination lemma removes the (still unknown) controller data (K) from the previous LMI, at cost of introducing two separate LMIs. The first LMI only takes into account the inputs that contribute to the state and weighted signals. The second LMI only takes into account the states and external signals that influences the output.

Note:

$$\begin{aligned} \ker P_x &= \ker \left(\begin{bmatrix} \bar{B}_2^T X & 0 & \bar{D}_{12} \end{bmatrix} \right) \\ \ker Q &= \ker \left(\begin{bmatrix} \bar{C}_2 & \bar{D}_{21} & 0 \end{bmatrix} \right) \end{aligned} \quad (\text{in input-output}) \quad \begin{aligned} &\begin{bmatrix} \bar{B}_u^T X & 0 & \bar{D}_{zu} \end{bmatrix} v = 0 \\ &\begin{bmatrix} \bar{C}_y & \bar{D}_{yw} & 0 \end{bmatrix} v = 0 \end{aligned}$$

Let's be u a vector of the kernel of P_x . This means that the vector u contribution to the state and to the weighted signal is null, because

$$\begin{aligned} x' &= Ax + B_1 w + B_2 u \\ z &= C_1 x + D_{11} w + D_{12} u \end{aligned} \quad \text{and} \quad B_2 u \equiv 0, \quad D_{12} u \equiv 0$$

This means that W_{P_x} is made of all vectors u (all inputs) that instead contributes to the state and weighted signal.

Similarly, let's be w a vector of the kernel of Q . This means that the vector w (*states, external signals*) contribution to the output is null, because

$$z = C_2 x + D_{21} w + D_{22} u \quad \text{and} \quad C_2 y \equiv 0, \quad D_{21} y \equiv 0$$

This means that W_Q is made of all vectors y (states and external signals) that instead contributes to the output.

But in previous W_P and $H(X)$ depend on X that means that the expression is not a LMI. (Note that W_P is the orthogonal of the kernel of P_x that depends on X).

$$P_x = \begin{bmatrix} \bar{B}_2^T X & 0 & \bar{D}_{12} \end{bmatrix}$$

Note also that there is no problem with the second inequality: $W_Q H(X)$ is linear in X because W_Q does not depend on X .

To resolve this non linearity on the first inequality we apply a congruence transformation (exactly the same that in the simplified static feedback problem):

$$T = \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \prec 0, \text{ being } Y = X^{-1},$$

$$T(Y) = T^T H(X) T \prec 0$$

Multiplying,

$$T(Y) = \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}^T \begin{bmatrix} \bar{A}^T X + X \bar{A} & X \bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X & -\gamma I & \bar{D}_{11}^T \\ \bar{C}_1 & \bar{D}_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

$$T(Y) = \begin{bmatrix} Y & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}^T \begin{bmatrix} \bar{A}^T X Y + X \bar{A} Y & X \bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X Y & -\gamma I & \bar{D}_{11}^T \\ \bar{C}_1 Y & \bar{D}_{11} & -\gamma I \end{bmatrix}$$

$$T(Y) = \begin{bmatrix} \bar{A}^T Y + Y \bar{A} & \bar{B}_1 & Y^T \bar{C}_1^T \\ \bar{B}_1^T & -\gamma I & \bar{D}_{11}^T \\ \bar{C}_1 Y & \bar{D}_{11} & -\gamma I \end{bmatrix} \quad (\text{Eq 3-10})$$

With this transformation the problem is stated as:

The equation (Eq 3-9):

$$F(X) = H(X) + Q^T K^T P_x + P_x^T K Q \prec 0$$

Has a solution K iff:

$$W_p^T T(Y) W_p \prec 0$$

$$W_Q^T H(X) W_Q \prec 0$$

That expands to:

$$W_p^T T(Y) W_p = W_p^T \begin{bmatrix} \bar{A}^T Y + Y \bar{A} & \bar{B}_1 & Y^T \bar{C}_1^T \\ \bar{B}_1^T & -\gamma I & \bar{D}_{11}^T \\ \bar{C}_1 Y & \bar{D}_{11} & -\gamma I \end{bmatrix} W_p \prec 0 \quad (\text{Eq 3-11})$$

$$W_Q^T H(X) W_Q = W_Q^T \begin{bmatrix} \bar{A}^T X + X \bar{A} & X \bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X & -\gamma I & \bar{D}_{11}^T \\ \bar{C}_1 & \bar{D}_{11} & -\gamma I \end{bmatrix} W_Q \prec 0 \quad (\text{Eq 3-12})$$

X is of dimension $(n_p + n_k)$. Let's partition X and Y accordingly as follows:

$$X = \begin{bmatrix} X_P & X_2 \\ X_2^T & I \end{bmatrix}, \quad Y = X^{-1} = \begin{bmatrix} Y_P & Y_2 \\ Y_2^T & I \end{bmatrix}$$

In turn, it can be demonstrated (refer to paragraph 8.2.8) that the previous two LMI conditions (Eq 3-11) and (Eq 3-12) are equivalent to the following 3 LMI conditions (independent of controller variables and expressed only in function of the open loop plant):

$$\begin{bmatrix} N_X & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T X_P + X_P A & X_P B_1 & C_1^T \\ B_1^T X_P & -\mathcal{A} & D_{11}^T \\ C_1 & D_{11} & -\mathcal{A} \end{bmatrix} \begin{bmatrix} N_X & 0 \\ 0 & I \end{bmatrix} \prec 0, \quad (\text{Eq 3-13})$$

$$\begin{bmatrix} N_Y & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T Y_P + Y_P A & Y_P C_1^T & B_1 \\ C_1 Y_P & -\mathcal{A} & D_{11} \\ B_1^T & D_{11}^T & -\mathcal{A} \end{bmatrix} \begin{bmatrix} N_Y & 0 \\ 0 & I \end{bmatrix} \prec 0, \quad (\text{Eq 3-14})$$

$$\begin{bmatrix} X_P & 0 \\ 0 & Y_P \end{bmatrix} \geq 0 \quad (\text{Eq 3-15})$$

Where:

$$\begin{aligned} \text{Im } N_X &= \ker \begin{bmatrix} C_2 & D_{21} \end{bmatrix} \\ \text{Im } N_Y &= \ker \begin{bmatrix} B_2^T & D_{12}^T \end{bmatrix} \end{aligned}$$

Summary: a controller can be synthesized for the closed loop system achieving

$$\|T_{wz}\|_{\infty} < \gamma \text{ iff the previous 3 LMIs have solution.}$$

3.2.1.2.3 Controller reconstruction

The previous paragraphs demonstrate that a controller exists (existence problem). The synthesis procedure for the controller is:

From the partition for X and Y construct:

$$X = \begin{bmatrix} X_P & X_2 \\ X_2^T & I \end{bmatrix}, \quad X - Y^{-1} = X_2 X_2^T$$

Then construct:

$$P_X = \begin{bmatrix} \bar{B}_2^T X & 0 & \bar{D}_{12}^T \\ \bar{C}_2 & \bar{D}_{21} & 0 \end{bmatrix}$$

Then solve the LMI in K (with X, P_X, Q known):

$$F(X) = H(X) + Q^T K^T P_X + P_X^T K Q < 0$$

K is the controller.

3.2.2 Properties of the H-Infinity controllers

The following properties are applicable to the H-Infinity controllers whatever the solution method used:

Property 1: All pass property

The optimum cost function T_{zw} is 'all-pass', i.e., it is equal to 1 for all the frequencies.

On one side this is negative because means that the optimal controller will not roll-off at high frequencies (meaning that the optimal controller is not proper and is not realizable).

In practice is positive because we are not interested in the optimal controller but in sub-optimal controllers. For a sub-optimal controller this all-pass property means that the desired shape for the closed loop transfer function can be reached (in a limited range of frequencies) if the appropriate weights are selected.

For example, if we take S_o :

$$\|W_s S_o\|_\infty \approx 1 \Rightarrow S_o \approx \frac{1}{W_s}$$

I.e. S_o is similar to the inverse of the desired weight (that have been chosen according to the design objectives).

Property 2: Dimension of the controller

The H-Infinity controller has a number of states equal to the number of states of the augmented plant (i.e. sum of the states of the open loop plant plus states of the weights).

In general the complexity (number of states) of the H-Infinity trends to be greater that the classical methods. Note however that the very recently "H-Infinity Structured" approach allows to synthetize controllers of desired order and structure.

Property 3: Possible pole zero cancellation

As many other methods of modern control the H-Infinity method trends to provide controllers that invert the plant being controlled.

Depending on the particular control problem and in particular when the original open loop plant is ill-conditioned this can be a serious problem. Pole zero cancellation cannot be perfect and for such ill-conditioned plants the closed loop performances can be very bad.

This thesis will introduce methods for management of these cases.

3.3 History of H-infinity

Some notes about the historic evolution of H-infinity theory are provided in this paragraph.

The development of H-infinity theory was started by (Zames, 1981) who studied the minimization of the sensitivity function of a SISO LTI feedback system.

$$\|S\|_{\infty} \equiv \sup_{w \in R} |S(iw)|$$

The problem was posed in the frequency domain. Solutions available at that time involved analytic functions (Nevanlinna-Pick interpolation) or operator-theoretic methods. These methods were quite complicated and provided limited insight on the structure of the solutions. Progress in these first years was slow.

A more practicable solution was proposed by Doyle and Glover (Doyle, 1984). It is based on the Youla parameterization approach (see appendix 8.2.2 for a discussion) that defines the following expression for all stabilizing controllers of a given plant.

$$K(s) = (1 - Q(s)P(s))^{-1} Q(s)$$

From previous we can get the expression for Q(s) (omitting the “s” by simplicity):

$$(1 - QP) K = Q;$$

$$K - QPK = Q;$$

$$K = Q + QPK;$$

$$K = Q (1 + PK)$$

$$Q(s) = K (1 + PK)^{-1} = K S_0$$

And we can get the expression of the usual closed loop transfer functions in function of the parameter Q:

$$T = PK (1 + KP)^{-1} = PQ$$

$$S = 1 - T = 1 - PQ$$

The objective of obtaining nominal performance is

$$\|W_p(s)S_o(s)\|_\infty < 1$$

And expressing S in function of Q:

$$\|W_p(s)(1 - P(s)Q(s))\|_\infty < 1$$

$$\|W_p(s) - W_p(s)P(s)Q(s)\|_\infty < 1$$

Now renaming $T_1 = W_p$, $T_2 = -W_pP$, $T_3 = 1$, we have

$$\|T_1(s) + T_2Q(s)T_3\|_\infty < 1$$

That is known as the “matching model” problem (Figure 3-4):

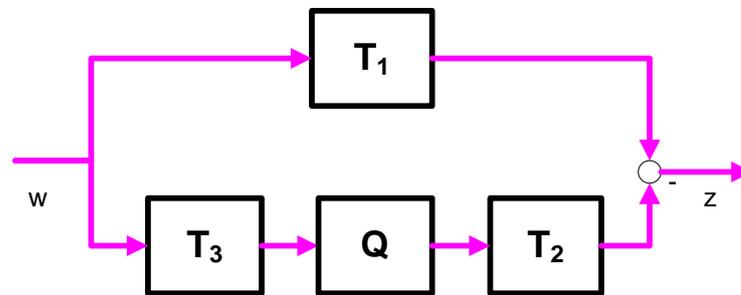


Figure 3-4: The model matching problem

In turn, the ‘matching model problem’ can be demonstrated to be equivalent to the ‘Hankel approximation problem’ for which a solution was developed by (Glover, 1984). The book (Maciejowski, 1989) contains a fully detailed demonstration of the 1984 solution.

The 1984 solution had several disadvantages: the controllers generated were of high order and the computation of the controller was very demanding.

A more satisfactory state space solution was found in 1998 due to the work of Doyle, Glover, Khargonekar and Francis, (Doyle, 1989) (popularly “known as the DGKF paper”). The degree of the synthesized controller was equal to the dimension of the augmented plant. This made the controller implementation feasible and starts the practical application of H-infinity.

A different approach is the solution of the problems by means of LMIs. The LMI solution was proposed in the article of Gahinet and Apkarian (Gahinet, 1994). A later paper with the generalization to multi-objective synthesis was proposed in the article of (Scherer, 1997).

The main advantages of the LMI solution over the DGKF solutions are:

- Less number of assumptions (even if there are workarounds for some of them in DGKF)

- Demonstrations are simpler and provides more educational value
- Multi-objective synthesis can be integrated normally in the framework

3.4 H-infinity Mixed Sensitivity

In the previous chapter the H-infinity problem has been introduced as finding a controller that minimizes the H-infinity norm of the closed loop transfer functions.

There are multiple alternatives for choosing the transfer functions to be weighted and there are multiple alternatives for choosing the weights.

The Mixed Sensitivity variant of the H-Infinity method is the variant most frequently used for this choice. The origin of its name is due to the fact that a “mix” of transfer functions are weighted. Usually they are S_o and T_o but other variants are possible.

For example, we may want to minimize S_o and T_o , trying to achieve a good reference tracking, robustness to output disturbances and robustness against multiplicative uncertainty at plant output by posing the following problem.

$$\|T_{zw}\|_{\infty} = \left\| \begin{array}{c} W_1 S_o \\ W_3 T_o \end{array} \right\|_{\infty} < \gamma$$

Or alternatively we can give more importance to the control signal, trying to guarantee that it meet some actuator constrains.

$$\|T_{zw}\|_{\infty} = \left\| \begin{array}{c} W_1 S_o \\ W_2 K S_o \end{array} \right\|_{\infty} < \gamma$$

We can also use 3 transfer functions:

$$\|T_{zw}\|_{\infty} = \left\| \begin{array}{c} W_1 S_o \\ W_2 K S_o \\ W_3 T_o \end{array} \right\|_{\infty} < \gamma$$

All the previous examples are particular cases of the H-Infinity standard problem. The stack of TF is simply the TF of the exogenous signals (w) to the regulated signals (z) of the augmented plant.

It shall be commented that an error is present when the H-Infinity norm of the previous problems is calculated. This is because the expression:

$$\|T_{zw}\|_{\infty} = \left\| \begin{array}{c} W_1 S_o \\ W_3 T_o \end{array} \right\|_{\infty} < 1$$

Is not exactly the same that compute the norm separately: $\|W_1 S_o\|_{\infty} < 1$ and $\|W_3 T_o\|_{\infty} < 1$

Usually we take as H_∞ norm of a vector the square root of the sum of the square of the components (i.e. the usual Euclidian norm).

$$\|T_{zw}\|_\infty = \sqrt{(\|W_1 S_o\|_\infty)^2 + (\|W_3 T_o\|_\infty)^2}$$

But if for example:

$$\|W_1 S_o\|_\infty = 1 - \varepsilon \quad \text{and} \quad \|W_3 T_o\|_\infty = 1 - \varepsilon,$$

Separately both components have an H_∞ norm minor than 1 and however the computed Euclidian norm is greater than 1 $\approx (\sqrt{2}(1 - \varepsilon))$.

This means that **the method H-infinity Mixed Sensitivity is conservative** because discards as invalid ($H_\infty > 1$) solutions that are valid. The error factor is minor than $\sqrt{2} \approx 3db$ that is low. So usually this error can be ignored and the method provides valid results.

The previous H-Infinity Mixed Sensitivity problems are studied in detail in the next paragraph. It will be explained how to build the augmented plant with the weights and that the transfer function of the closed loop is in fact a LFT of the plant and the controller.

3.4.1 H-infinity Mixed Sensitivity S-KS: reference tracking

The objective is to follow a given reference signal. The TFs to shape are S_o and KS_o .

With a proper shape of S_o we will achieve a good tracking of the reference and robustness against disturbances at the output of the plant. With a proper shape of KS_o we will limit the control effort at high frequencies (Figure 3-5).

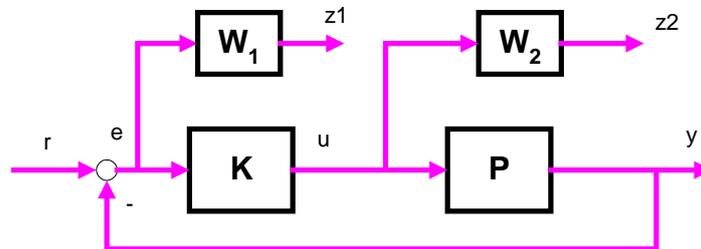


Figure 3-5: H-inf Mixed Sensitivity for reference tracking

We want to minimize T_{zw}

$$\|T_{zw}\|_{\infty} = \left\| \begin{bmatrix} W_1 S_o \\ W_2 K S_o \end{bmatrix} \right\|_{\infty} < \gamma$$

It is demonstrated now that the S-KS problem is just a particular case of the general formulation. Let's reformat the previous system of (Figure 3-5) as shown in (Figure 3-6):

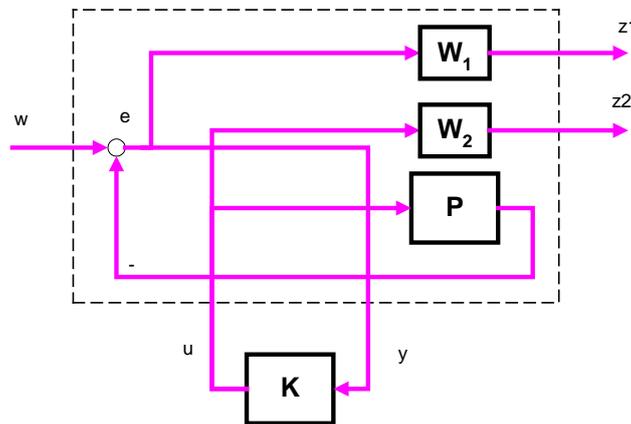


Figure 3-6: H-inf Mixed Sensitivity for reference tracking as LFT

Note that no connection has been changed: only the layout of the diagram has been changed. We have defined the external inputs to the plant (references and disturbances) as 'w' and the signals to weight as 'z'. The input to the controller is 'y' and the output of the controller is named 'u'. The dashed line defines the augmented plant.

By analysis of the previous plant we obtain the following relations:

$$\begin{aligned} z_1 &= W_1 w - W_1 P u \\ z_2 &= W_2 u \\ v &= w - P u \end{aligned}$$

That can be written in matrix form:

$$\begin{bmatrix} z_1 \\ z_2 \\ v \end{bmatrix} = \begin{bmatrix} W_1 & -W_1 P \\ 0 & W_2 \\ 1 & -P \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$$

Now we can use the general LFT formula for finding the TF from the external signals (w) to the regulated signals (z):

$$F_l(M, \Delta_l) \equiv M_{11} + M_{12} \Delta_l (1 - M_{22} \Delta_l)^{-1} M_{21}$$

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1 \\ 0 \end{bmatrix} + \begin{bmatrix} -W_1PK \\ W_2K \end{bmatrix} \begin{bmatrix} 1 - (-P)K \\ 1 \end{bmatrix}^{-1} w$$

Developing the terms, we have

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1 - W_1PK(1 + PK)^{-1} \\ W_2K(1 + PK)^{-1} \end{bmatrix} w$$

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1PK(1 + PK)^{-1} \\ W_2K(1 + PK)^{-1} \end{bmatrix} w = \begin{bmatrix} W_1S_o \\ W_2KS_o \end{bmatrix} w$$

That is the formulation of the S/KS problem.

So it has been demonstrated that the **Mixed Sensitivity S/KS problem is just the minimization of the norm of the TF (external signals to the regulated signals) of the augmented plant.**

3.4.2 H-infinity Mixed Sensitivity S-T

The objective is to follow a given reference signal. However in this variant we want to have more control on the shape of T , probably for modeling the behavior at high frequencies. The TFs to shape are S_o and T_o (Figure 3-7).

With a proper shape of S_o a good tracking of the reference and robustness against disturbances at the output of the plant can be achieved. With a proper shape of T_o robustness to modeling errors at high frequencies can be achieved.

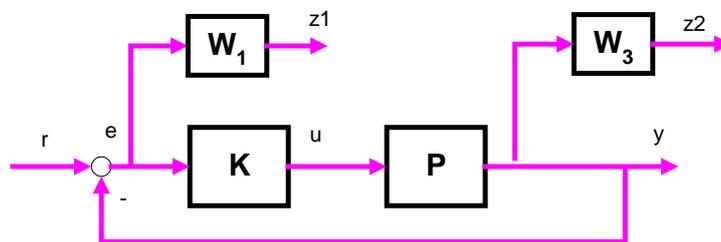


Figure 3-7: H-inf Mixed Sensitivity S-T

We want to minimize T_{zw}

$$\|T_{zw}\|_{\infty} = \left\| \begin{bmatrix} W_1 S_o \\ W_3 T_o \end{bmatrix} \right\|_{\infty} < \gamma$$

It is demonstrated now that the S-T problem is just another particular case of the general formulation. Let's reorder the previous system of (Figure 3-7) as a LFT (Figure 3-8):

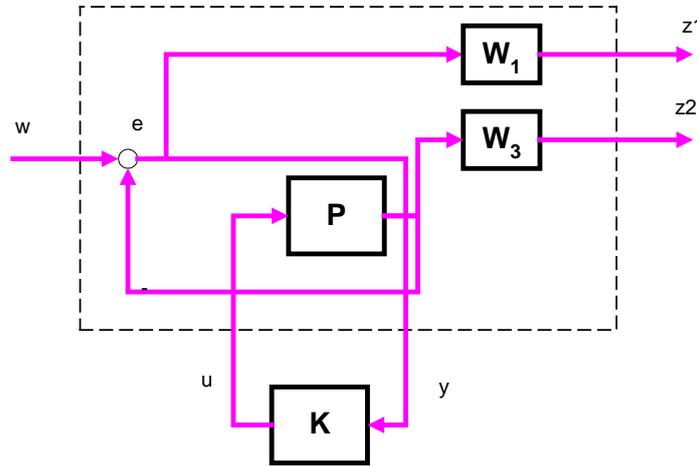


Figure 3-8: H-inf Mixed Sensitivity S-T a LFT

By analysis of the previous plant we obtain the following relations:

$$\begin{aligned} z_1 &= W_1 v = W_1 w - W_1 P u \\ z_3 &= W_3 P u \\ v &= w - P u \end{aligned}$$

That can be written in matrix form:

$$\begin{bmatrix} z_1 \\ z_3 \\ v \end{bmatrix} = \begin{bmatrix} W_1 & -W_1 P \\ 0 & W_3 P \\ 1 & -P \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$$

Now we can use again the general LFT formula for finding the TF from the external signals (w) to the regulated signals (z):

$$F_l(M, \Delta_l) \equiv M_{11} + M_{12} \Delta_l (1 - M_{22} \Delta_l)^{-1} M_{21}$$

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1 \\ 0 \end{bmatrix} + \begin{bmatrix} -W_1 P K \\ W_3 P K \end{bmatrix} \left[1 - (-P) K \right]^{-1} 1$$

Developing the terms, we have

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1 - W_1 PK(1 + PK)^{-1} \\ W_3 PK(1 + PK)^{-1} \end{bmatrix} \omega$$
$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1 PK(1 + PK)^{-1} \\ W_3 PK(1 + PK)^{-1} \end{bmatrix} \omega = \begin{bmatrix} W_1 S_o \\ W_3 T_o \end{bmatrix} \omega$$

So it has been demonstrated that the **Mixed Sensitivity S/T problem is just the minimization of the norm of the TF (external signals to the regulated signals) of the augmented plant.**

3.4.1 Problems of H-infinity Mixed Sensitivity

The H-infinity Mixed Sensitivity weights closed loops functions as S , T , KS_o . The open loop plant is not “seen” by the method and is not weighted.

This can be an “a priori” problem as said by (Maciejowski, 1989): we pretend to design a controller without taking into account the characteristics of the open loop plant; we only specify its closed loop behavior. This can work or not, depending on the plant. In fact it will not work for ill conditioned plants.

This topic is one of the objects of this thesis and is developed extensively in chapter 4.

3.5 Current state of the art for H-infinity

After the availability of the DKGF solution in 1988 H-Infinity became a practical method for controller synthesis. The arrival of the LMI solution in 1994 multiplied the choices and the connection to other fields.

The decade 2000-2010 has assisted to the application of the H-infinity problem (previously limited to the LTI systems) to a wider variety of problems: nonlinear problems, discrete time problems, Linear Parameter Varying (LPV) problems, fixed-order controller problems, etc.

This has been made possible mainly by the LMI solutions where multi-objective goals, and in particular Linear Parameter Varying (LPV) problems can be introduced in a more easy way. LPV is still an important research topic that is introduced later.

A new application with growing interest is the H-Infinity filtering. The classical Kalman filtering provides an estimated plant with the smallest standard deviation of the error

between the real plant and the estimation. However the classical Kalman filtering needs some conditions to be hold: the average value of the noise shall be zero and the standard deviation of the noise shall be known. The H-infinity filtering instead minimizes the worst case error (not the quadratic error) and can be applied even when the characteristics of the noise are not known.

Filtering is used for example in satellites for fault detection of the thrusters. The failure of the thrusters can be detected from Ground by telemetry analysis, but the satellite is not always on visibility from the Ground Stations. An autonomous on board Fault Diagnosis capability is desirable. On the H-infinity framework, the problem is reduced to find an observer that minimizes the H-infinity norm of the difference between a combination of selected inputs (control signals and command references) and outputs (measures).

The more important breakthrough in the H-Infinity world in the last decade is the H-Infinity Structured. Its interest deserves a dedicate paragraph. A brief introduction to LPV control (which solution strongly connected with the LMI theory) is also provided.

3.5.1 Structured H-Infinity

3.5.1.1 Practical problems with “classical” H-Infinity

H-Infinity synthesis (either the DGKF synthesis or the LMI synthesis) has some limitations that have slowed its adoption in the industry, in particular the Aerospace industry. There are 2 main causes that might explain this fact.

A first cause was that H-infinity controllers have more states than classical controllers. A controller of high order needs more computational power. This is usually not a concern in industrial plants but it was a concern in the aerospace world.

Note:

The 1750 processor commonly used in aerospace during the eighties and nineties is a 16-bit computer with a 20MHz clock speed and limited floating point support. Some examples of usage are the F-16 and F18 fighters and ESA missions as Rosetta and Envisat). It is to be remarked that the 1750 processor will be used in satellite communications up to 2020 (the space communications world is very conservative a major driver is cost-saving!). The reason of such low computational power is that the processors are required to be “radiation-hardened”. This implies different technologies that commercial chips: the technologies used in the space chips cannot work at high clock speeds.

This first cause has become a minor concern with the availability of more powerful processors since year 2000 (ERC32 and LEON family by ESA for aerospace that include a full mathematical processor of 64-bits). Be aware that at 2014 new ESA missions include computers with LEON2/LEON3 a clock running from 66 to 80 MHz!).

A second problem however has been persistent and in fact is still present today. The H-infinity method generates controllers without defined structure. The controller generated has n states being n the sum of the number of states of the open loop plant and the number of states of the weights. The controller has no defined structure (the controller is a “black-box” for the designer). This lack of structure made more difficult to understand and tune the controller.

This is different from traditional techniques (example tuned PIDs) where the controller is defined by a number of PID blocks plus filters interconnected in a well-known setup. The existence of a predefined structure of the classical methods allows more insight for the control engineer: the meaning of the proportional, integral and derivative terms, pre-filters, feed-forward terms, roll-off terms, etc. are well known and make possible manual tuning.

The problem lack of structure of the H-Infinity controller has been resolved by the recent work of Apkarian and Nole (Apkarian, 2006) . The new method is called “H-infinity Structured” and allows choosing a given structure for the generating controller. This method is studied in the next paragraph.

3.5.1.2 Fundamentals of H-Infinity Structured

Given the general control problem (Figure 3-9):

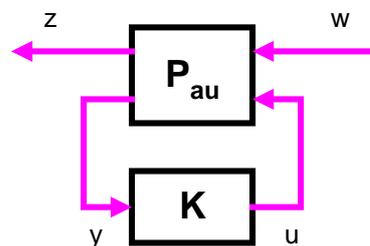


Figure 3-9: Robust control problem

With a space state representation:

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}$$

The H-Infinity Structured problem of control is expressed as:

To find a controller that stabilizes internally the system and guarantees that the H-infinity norm of the TF from w to z is minor than a given number and in addition satisfies that the controller belongs to a predefined set of controllers

$$\|T_{wz}(K)\|_{\infty} < \gamma \text{ subject to } K \in \mathbf{K}$$

where \mathbf{K} is the family of controllers satisfying a given structural constraint

It is useful to reminder the following identities:

$$\|T_{wz}(K, \omega)\|_{\infty} = \max_{\omega \in [0, \infty]} \bar{\sigma}(T_{wz}(K, \omega)) = \max_{\omega \in [0, \infty]} \lambda_{\max}(T_{wz}^H(K, \omega)T_{wz}(K, \omega))^{1/2}$$

Formalization of the concept of structured controller

Assuming that the controller K is structured with a vector θ of parameters (where it is assumed that the dependency of K in θ is smooth):

$$K(\theta) = \left[\begin{array}{c|c} A_K(\theta) & B_K(\theta) \\ \hline C_K(\theta) & D_K(\theta) \end{array} \right]$$

The closed loop function will be:

$$T_{wz}(P, K(\theta)) = \left[\begin{array}{c|c} A(K(\theta)) & B(K(\theta)) \\ \hline C(K(\theta)) & D(K(\theta)) \end{array} \right]$$

With TF:

$$T_{wz}(P, K(\theta)) = C(K(\theta))(sI - A(K(\theta)))^{-1}B(K(\theta)) + D(K(\theta))$$

The objective becomes to minimize the closed loop transfer function

$$f(\theta) := \|T_{wz}\|_{\infty} = \max_{\omega \in R} \bar{\sigma}(C(K(\theta))(sI - A(K(\theta)))^{-1}B(K(\theta)) + D(K(\theta)))$$

With the additional condition of that the closed system loop is stable

$$D_s = \{\theta \in R^n : A(K(\theta)) \text{ is stable}\}$$

Note that a nested minimization problem has been defined: the closed loop function depends on the controller K and in turn the controller K depends on the parameter θ .

The following composite function has been defined (mathematically it is a 'functional'), that assign to each set of parameters θ a value of the H-infinity norm:

$$f : \theta \rightarrow \left(\|\cdot\|_{\infty} \circ T_{zw} \circ K(\theta) \right)$$

The previous minimization problem is non-smooth: the vector θ of parameters is the cause of the problem becoming non-smooth.

The minimization algorithms commonly used are based on the use of the derivative of the function to minimize a cost: starting from a given point, calculate the steepest descent direction (with the derivative) and repeat until a minimum is found. But the composite function is non-derivable because it is non-smooth.

We have to introduce the concept of sub-differential that replaces the derivative when calculating the steepest descent direction in the minimization problem.

Sub-differentials

The concept of sub-differential is a generalization of the concept of derivation for functions that are not smooth. For a smooth function we have a unique value for the derivative in a given dimension. For a non-smooth function we have a set of values for the sub-derivative in each dimension (Figure 3-10).

Intuitively, the sub-derivative at a point x_0 is the set of all the lines that are under the function: i.e. computing the value in a point x close to x_0 as a first order expansion gives a value always minor than the real value of the function in x .

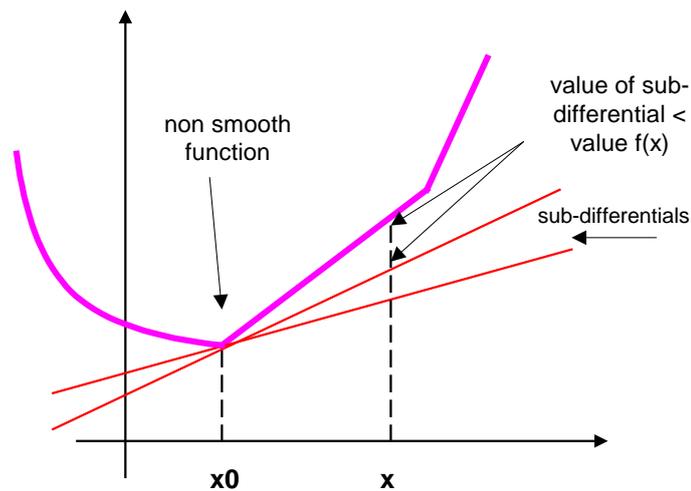


Figure 3-10: Sub-differential

Formally the sub-derivative is expressed as:

$$f(x) - f(x_0) \geq v \bullet (x - x_0)$$

Where the 'dot' denotes the scalar product. Remainder:

$$x \bullet y \equiv \langle x, y \rangle \equiv Tr(x^H y)$$

Sub-differential of the H-infinity norm

Let be a transfer function $G(s)$. We know that:

$$\|G(s)\|_\infty = \bar{\sigma}(G(i\omega))$$

Let be a singular decomposition of $G(s)$

$$G = U\Sigma V^H = U \begin{bmatrix} \bar{\sigma}_{\max} & & \\ & \dots & \\ & & \dots \end{bmatrix} V^H$$

Taken only the first column of U and V and the biggest singular value we have:

$$G(s) = u \bar{\sigma}(G(s)) v$$

$$G(s) = u \|G(s)\|_{\infty} v$$

$$u = \|G(s)\|_{\infty}^{-1} G(s) v$$

By analogy with a Lyapunov quadratic function $V(x) = x^T P x$, let's introduce the linear functional Φ defined as:

$$\Phi(H) = \text{Re}(u^H H(i\omega) v)$$

And substituting in previous:

$$\Phi(H) = \text{Re}(u^H H(i\omega) v) = \text{Re}\left(\left(\|G(s)\|_{\infty}^{-1} G(s) v\right)^H H(i\omega) v\right) \tag{Eq 3-16}$$

$$\Phi(H) = \|G(s)\|_{\infty}^{-1} \text{Re}(v^H G(s)^H H(i\omega) v)$$

That is continuous in the space of the H_{∞} stable transfer functions and is a subgradient of $\|\cdot\|_{\infty}$ in $G(s)$. (This is because the expression for Φ is divided by the H-infinity norm of $G(s)$, so in principle any first order expansion of $G(s)$ centered on this point will be minor that the value of $G(s)$ on the vicinity.

Derivative of closed loop transfer function with regard the controller K

Given a plant P and a controller K , the closed loop transfer function is given by the usual LFT formula

$$T_{zw}(K) = P_{11} + P_{12} K (I - P_{22} K)^{-1} P_{21}$$

The derivative of T at K is given by

$$T'_{zw}(K) \delta K = P_{12} (I - P_{22} K)^{-1} \delta K (I - K P_{22})^{-1} P_{21} \tag{Eq 3-17}$$

Note:

We have to find the derivative of the term $K(I - P_{22} K)^{-1}$. The derivative of the scalar function is

$$\frac{\delta}{\delta x} \left(\frac{x}{(1 - px)} \right) = \frac{1}{(1 - px)^2}$$

So by analogy the derivative is

$$(I - P_{22}K)^{-2} \delta K = (I - P_{22}K)^{-1} \delta K (I - KP_{22})^{-1}$$

Clarke sub differential of the composite function

The composite function is defined as

$$f : \theta \rightarrow \left(\|\cdot\|_{\infty} \circ T_{zw} \circ K(\theta) \right)$$

I.e. for each set of parameters θ of the controller, calculates the closed loop function and then calculates the H-infinity norm.

Then f is Clarke-sub differentiable and allows applying the usual chain rule for the derivative:

$$\partial_c f(\theta) = \partial(\|T_{zw}\|_{\infty}) \frac{\delta T_{zw}}{\delta K} \frac{\delta K}{\delta \theta}$$

But by equation (Eq 3-17) we have the expression for $\frac{\delta T_{zw}}{\delta K}$ and by equation (Eq 3-16) we have the expression Φ for a sub-differential of the H-infinity norm $\partial_c(\|T_{zw}\|_{\infty})$

$$\partial_c f(\theta) = \Phi(T_{zw}) \frac{\delta T_{zw}}{\delta K} \frac{\delta K}{\delta \theta}$$

So we arrive to an expression for the Clarke sub-differential for $f : \theta \rightarrow \left(\|\cdot\|_{\infty} \circ T_{zw} \circ K(\theta) \right)$:

$$\partial_c f(\theta) = \Phi_Y = \|T_{zw}\|_{\infty}^{-1} \text{Re} \left((I - KP_{22})^{-1} P_{21} T_{zw}^H Q Y Q^H P_{12} (I - P_{22}K)^{-1} \right) \quad (\text{Eq 3-18})$$

The differential of K respect θ is particular of each problem and depends on the structure of the controller.

Optimization algorithm

The problem stated is a minimization problem: find the controller that minimizes the H-infinity norm of the composite function $f : \theta \rightarrow \left(\|\cdot\|_{\infty} \circ T_{zw} \circ K(\theta) \right)$. Instead of a normal derivative the Clarke sub-differential has to be used as follows:

1. Chose a closed-loop stabilizing controller K
2. If Clarke sub-differential is 0 (local minimum, then STOP)
3. Compute step descent direction as

$$H = -\frac{\Phi_Y}{\text{Tr}(\Phi_Y^H \Phi_Y)} \quad \text{with } \Phi_Y \text{ of (Eq 3-18)}$$

4. Do a line search along this step descent, using $x_{new} = x + H\delta$

5. Go to 2

In practice the steepest descent algorithm have some numerical problems. More advanced methods are used. Refer to (Apkarian, 2006) for a full description of such minimization methods.

As usually with the new control methods the mathematics of the H-Infinity Structured technique is very complex. However commercial packages are already available. The H-Infinity Structured Matlab toolbox is used later on this thesis.

3.5.2 Linear Parameter Variant (LPV) systems

The classical approach to deal with non-linear plants is the gain schedule approach: some operation points are chosen, the nonlinear plant is linearized at these points and a LTI (classical, H-Infinity or whatever) is designed for each point.

Then the controllers are scheduled based on some parameter (time, Mach, etc.). The problem with this manual approach is that the control robustness guaranteed for the controllers at the “operation points” is no longer guaranteed for the “scheduled controllers”. Also effects as wind-up arose in the regions when the scheduling changes the controller.

The LPV theory formalizes the concept of “manual gain scheduling” guaranteeing robustness not only for the closed loop plant at the “operation points” but for the full set of plants at all operative conditions.

In a Linear Parameter Variant system the state space equations of the plant and the controller are not constant and depend (smoothly) on a parameter ρ that changes along time, i.e. $\rho = \rho(t)$. Without loss of generality (by introducing an appropriate scaling) we can impose that the parameter values be on $(-1 < \rho < 1)$.

The parameter ρ is not known in advanced but it is assumed that can be measured. It is also assumed that the matrices (A,B,C,D) are affine (depends linearly) in ρ .

The problem is formulated as: given a plant that depends smoothly on the parameter ρ :

$$\begin{aligned}x' &= A(\rho(t))x(t) + B(\rho(t))w(t) \\z &= C(\rho(t))x(t) + D(\rho(t))w(t)\end{aligned}$$

Find a controller $K = K(\rho)$ that made stable the plant and guarantees that $\|T_{wz}\|_{\infty} < \gamma$ for all values of ρ .

For each value of ρ we have a “traditional” LMI problem which solution is known. But this would mean to solve an infinite number of LMIs. What is done is to choose a basis of

the parameter space of ρ . Any value of ρ can be expressed as a lineal combination of vector of the basis.

$$\rho = \sum_{i=1}^n \alpha_i \rho_i$$

And then solve the following system of LMIs:

$$F(X) = \begin{bmatrix} A^T(\rho_i)X + XA(\rho_i) & XB(\rho_i) & C^T(\rho_i) \\ B^T(\rho_i)X & -\mathcal{M} & D^T(\rho_i) \\ C(\rho_i) & D(\rho_i) & -\mathcal{M} \end{bmatrix} \prec 0, \quad i = 1..n$$

The advantage of the LPV method is that guarantees stability and performances for any trajectory of the parameter inside the allowable region.

The disadvantage is that the method is conservative, i.e. the optimization can discard controllers that would work.

There is a lot of research on LPV systems. A very recent book with examples of LPV applications is (Mohammadpour, 2012).

4 The problems of the H-Infinity method when applied to ill-conditioned plants

This chapter describes the problems that the most commonly used H-Infinity method (the H-Infinity Mixed Sensitivity) has with ill-conditioned plants in presence of disturbances and how to solve them.

First we will introduce a famous control problem, the control of a distillation plant in presence of disturbances in the actuators. This problem was introduced by Skogestad, Morari and Doyle. A simple model of a column distillation is used to demonstrate that ill-conditioned plants are very sensitive to model uncertainty in the actuators. The problem was formalized as a benchmark problem (the Control of a Distillation Column ‘CDC’ benchmark problem) in (Limebeer, 1991).

After the introduction of the CDC benchmark it is checked that in fact H-infinity Mixed Sensitivity does not work with this problem. Then alternative optimizations to the H-Infinity method to overcome the problems of the Mixed Sensitivity approach are described. The results of this chapter have been accepted for publication, see the reference (Sanchez, 2015).

Finally the H-Infinity controllers are confronted with reference μ controllers using the CDC benchmark.

4.1 The CDC distillation process benchmark

The plant model and the design specifications for the CDC benchmark problem (Limebeer, 1991) are presented hereafter.

4.1.1 Plant Model

The process to be controlled is a distillation column (see Figure 4-1). The inputs are the reflux (dL) and boilup (dV) flows. The outputs are the composition of the distillate (dy_D) and residual (dx_B). The column is feed with a mixture (F). The control objective is that the distillate be as pure as possible and enough quantity of distillate is obtained. By actuating the boiler the mixture evaporates and distillate exit by the upper part of the column. The reflux valve allows controlling the flow of distillate that is re-injected in the column for increasing the purity of the distillate. The outputs are the product compositions at each flow. The resulting simplified model is:

$$\begin{bmatrix} dy_D \\ dx_B \end{bmatrix} = G_{LV} \begin{bmatrix} dL \\ dV \end{bmatrix}$$

$$\begin{bmatrix} dy_D \\ dx_B \end{bmatrix} = \frac{1}{75s + 1} \begin{bmatrix} 0.878 & -0.864 \\ 1.082 & -1.096 \end{bmatrix} \begin{bmatrix} k_1 e^{-\tau_1 s} & 0 \\ 0 & k_1 e^{-\tau_2 s} \end{bmatrix} \begin{bmatrix} dL \\ dV \end{bmatrix}$$

Where k_1 and k_2 are the actuator gains ($0.8 \leq k_i \leq 1.2$), i.e. a 20% un uncertainty in each input channel and τ is the delay, ($0 \leq \tau_i \leq 1 \text{ min}$). The channel 1 will be the one actuated by the reflux (dL) and the channel 2 the one actuated by the boil-up (dV). Note that the previous model is expressed in time units of minutes. It is important to remark that the uncertainty in the actuators is independent for each channel. However the delay has been considered in the literature the same for both channels (i.e. $\tau_1 = \tau_2$).

This means the gain the plant is strongly dependent on the directions of the inputs. For example, if the sign of the inputs is opposed (the reflux decreases and the boil-up increases), a fast and big response is obtained (high quantity of distillate but of low purity) but if the inputs have the same sign the response is very small (high purity distillate is obtained but in low quantity)). This made the plant easy to control for some input combinations and very difficult to control for others. Indeed as Skogestad remarks it is difficult to guarantee that the control signal are at the same time big but its difference is small given the high level of uncertainty present. This ill conditioning as can be deduced from the singular values diagram (Figure 4-2).

We assume in this and next chapters the following notation for naming a particular plant of the uncertainty set of plants:

$g(idK_1)(idK_2)(idDelay)$ where:

	i = 1	i = 2	i = 3
idK _i	0.8	1	1.2
idDelay	0 min	0.5 min	1 min

For example, g_{221} is the nominal plant with no delay, i.e. with: ($k_1 = 1.0, k_2 = 1.0, delay = 0 \text{ min}$), and g_{133} is the plant with: ($k_1 = 0.8, k_2 = 1.2, delay = 1 \text{ min}$)

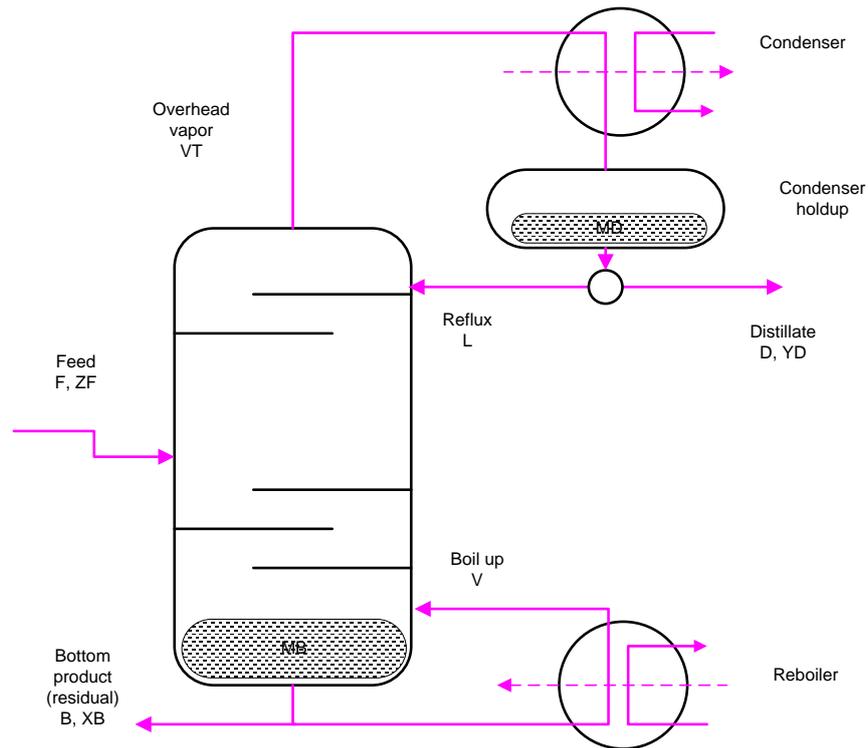


Figure 4-1: Distillation Plant

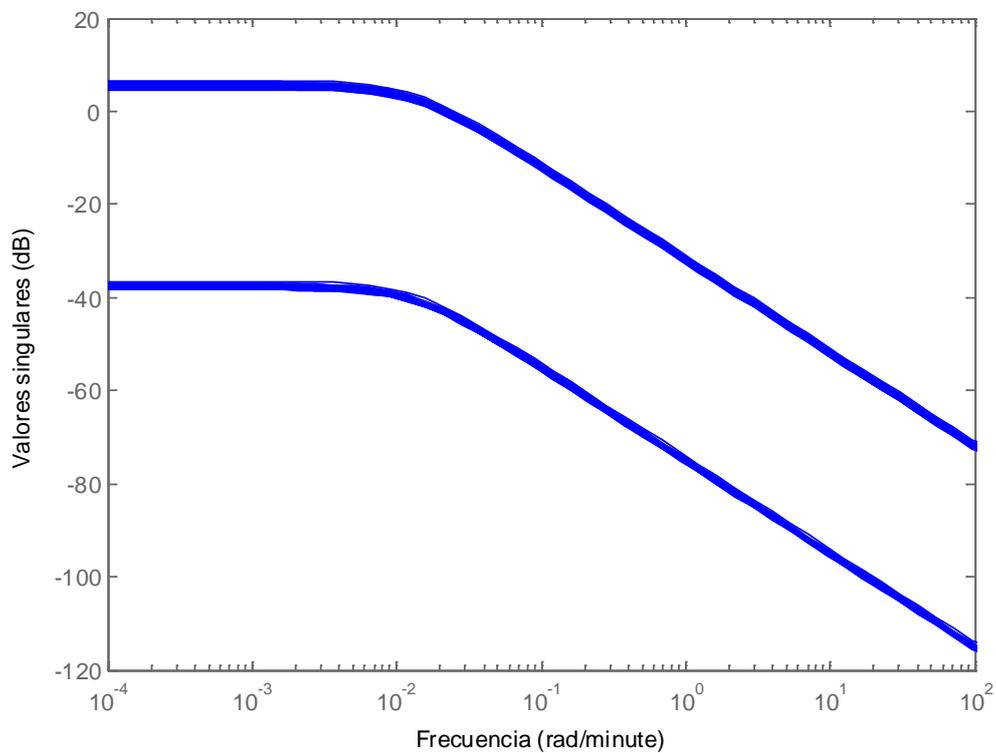


Figure 4-2: Singular values of the Open Loop Plant

The delay is usually modeled by a Padé approximation. In our case we use a second order model ($n = 2$). For a delay (Θ) of 60 seconds, this introduces the following transfer

function in the system:

$$e^\theta \approx \frac{\left(1 - \frac{\theta}{2n}s\right)^n}{\left(1 + \frac{\theta}{2n}s\right)^n}$$

With $(\theta = 0$ and $n = 2)$ becomes

$$delay = \frac{s^2 - 0.1s + 0.003333}{s^2 + 0.1s + 0.003333}$$

That has a right half zero in $z_{delay} = 0.005 \pm 0.033i$. Remembering the fundamental limitations presented in paragraph 2.1.6.3, it means that the sensitivity function S at that z_{delay} cannot be minor than 1 at this frequency ($1/60 = 0.0167$ rad/s ≈ 1 rad/min).

This means that at frequencies about 1 rad/min the control will have no effect in the system. This imposes a limitation on the design of weights as will be shown after. It shall be noted that as the transfer function has a roll-off the effect of the zero will be manifested at lower frequencies (e.g. a decade before the zero for first order systems).

Remark about the plant: Some reference μ controllers that used later modify the CDC plant by normalizing it introducing a factor of 100 in the static gain, see (Skogestad, 1996).

$$\begin{bmatrix} dy_D \\ dx_B \end{bmatrix} = \frac{1}{75s+1} \begin{bmatrix} 87.8 & -8.64 \\ 108.2 & -109.6 \end{bmatrix} \begin{bmatrix} k_1 e^{-\tau_1 s} & 0 \\ 0 & k_1 e^{-\tau_2 s} \end{bmatrix} \begin{bmatrix} dL \\ dV \end{bmatrix}$$

We have preferred in this work to work with the original CDC plant and not with the scaled version. Scaling (if any) will be introduced as part of the controller in this work.

4.1.2 Control Specifications of the CDC problem

The following specifications are defined in (Limebeer, 1991):

S1) Closed-loop stability

S2) Outputs performances

For a unit step demand in channel 1, the plant outputs (tracking) and (interaction) should satisfy:

- $y_1(t)$ should be < 1.1 for all t , and it should go from 0 to > 0.9 in no more than 30 minutes
- $y_2(t)$ should not exceed 0.5
- $1.01 \geq y_1(\infty) \geq 0.99$
- $0.01 \geq y_2(\infty) \geq -0.01$

Corresponding performances apply for a unit step demand on channel 2.

For a step $u = [0.4 \ 0.6]'$, the plant outputs (tracking) and (interaction) should satisfy:

- $y_1(t)$ should be < 0.5 for all t , and it should go from 0 to > 0.35 in no more than 30 minutes
- $y_2(t)$ should be < 0.7 for all t , and it should go from 0 to > 0.55 in no more than 30 minutes
- $0.41 \geq y_1(\infty) \geq 0.39$
- $0.61 \geq y_2(\infty) \geq -0.59$

In order to limit the control effort (avoiding controllers with unrealistic gains and bandwidths) the specifications (S3) and (S4) are defined:

S3) Control effort: $KS_o < 50 \text{ dB}$

S4) $KS_o < 1$ for $\omega > 150 \text{ rad/min}$

The defined input scenarios are:

Scenario 1: increment the percentage of desired product on the distillate without change the percentage on the residual

$$R_1' = [Y_D \ X_B] = [1 \ 0];$$

Scenario 2: complement of scenario 1

$$R_2' = [Y_D \ X_B] = [0 \ 1];$$

Scenario 3: simultaneous increase on both references

$$R_3' = [Y_D \ X_B] = [0.4 \ 0.6];$$

4.2 Problems of H-Infinity Sensitivity with ill conditioned plants and disturbances

4.2.1 The cause: the controller trends to invert the plant

It is known that the H-Infinity Mixed Sensitivity approach may produce bad results for ill conditioned plants. The reason is that the H-Infinity algorithm produces a controller that cancels the poles of the plant with its transmission zeros. This pole – zero cancellation is not desirable if the plant to control is subjected to uncertainty. In effect, the designed controller will perform very well for the nominal plant but may perform badly if perturbations are present. Of course, in real control problems uncertainties will be always present.

The reference (Christen, 1997) offers both, a formal demonstration and an intuitive demonstration of the fact that the H-Infinity Mixed Sensitivity controllers will trend to invert the plant. The intuitive demonstration is as follows:

The H-Infinity algorithm generates controllers that are all-pass i.e. that have more or less the same magnitude of the major singular value over all the range of frequencies. This means $T_{zw} \approx 1$ if the controller is optimal or nearly optimal.

Suppose we are weighting S_o with a weight W_P . Suppose we have found a good controller so that:

$$W_P S_o \approx U \approx 1$$

This means that S_o is approximately the inverse of W_P .

As the definition of S_o is:

$$S_o = (1 + GK)^{-1}$$

That at low frequencies can be approximated to $(GK)^{-1}$ having:

$$\begin{aligned} W_P S_o &\approx 1 && \rightarrow \\ W_P (K^{-1} G^{-1}) &= 1 && \rightarrow \\ K &= G^{-1} W_P \end{aligned}$$

This means that K will contain the inverse of the plant. This is the pole zero cancellation effect referred previously.

4.2.2 Demonstration of the problem with the CDC benchmark

The problem of the H-infinity Mixed Sensitivity approach with the CDC benchmark is demonstrated here. Let's generate the controller taking the following weights W_e for S_o and W_y for T_o . (The guidelines for weight selection will be explained later).

As weight W_e for S_o we follow the guidelines:

- Choose a 2nd order system that meets the specifications (chose ω_n and delta)
- $T_{o_id} = \omega_n^2 / (s^2 + 2 \text{ delta } \omega_n s + \omega_n^2)$
- Calculate $S_{o_id} = 1 - T_{o_id}$
- Define the weight as the inverse of S_{o_id}

Specifically, we choose

- $\omega_n = 10/75$; delta = 1.5; (i.e. an ideal system 10 times faster than the open loop)
- ($\tau = 75$ minutes)

As weight W_y for T_o we take an ideal first order system that meet the specifications.

- Choose a first order system that meets the specifications. Introduce a zero to make it proper.
- $T_{o_id} = (0.1s + 60) / (75/20 s + 60)$
- Define the weight as the inverse of T_{o_id}

The expression of the weights (in rad/min) is:

$$W_s = I_2 \left(\frac{s^2 + 0.4s + 1.778 * 1e^{-6}}{s^2 + 0.4s + 0.01778} \right)$$

$$W_y = I_2 \left(\frac{225s + 60}{0.1s + 60} \right)$$

The results of the benchmark for the criteria S2 are shown in the following tables.

plant ch	set-point tracking			interaction	
plant	t=30	max	t=100	max	t=100
g223_u1	0.958	1.000	1.000	0.000	-0.000
g113_u1	0.914	1.000	1.000	0.000	-0.000
g133_u1	-1.403	5.191	0.993	7.287	-0.009
g313_u1	3.302	6.469	1.008	7.261	0.010
g333_u1	0.982	1.000	1.000	0.000	-0.000
plant	t=30	max	t=100	max	t=100
g223_u2	0.958	1.000	1.000	0.000	0.000
g113_u2	0.914	1.000	1.000	0.000	0.000
g133_u2	3.300	6.490	1.007	4.660	0.006
g313_u2	-1.408	5.171	0.992	4.643	-0.006
g333_u2	0.982	1.000	1.000	0.000	0.000
plant	t=30	max	t=100	max	t=100
g223_u3	0.383	0.400	0.400	0.600	0.600
g113_u3	0.365	0.400	0.400	0.600	0.600
g133_u3	0.567	0.744	0.401	1.005	0.601
g313_u3	0.191	0.399	0.399	0.599	0.599
g333_u3	0.393	0.400	0.400	0.600	0.600

Table 4-1: S2 criteria for Mixed Sensitivity controller

The temporal response of S_o and T_o for a set of 25 Monte Carlo plants for the scenario 1 are shown in (Figure 4-3), (Figure 4-4) and (Figure 4-5).

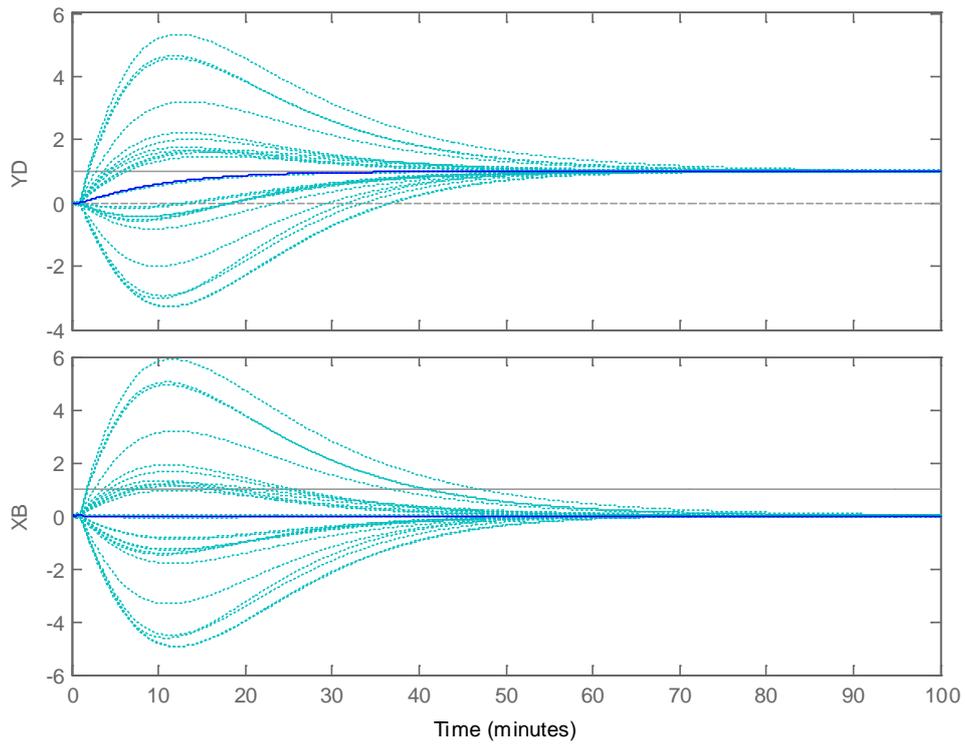


Figure 4-3: H-inf Mixed Sensitivity controller response to sce1

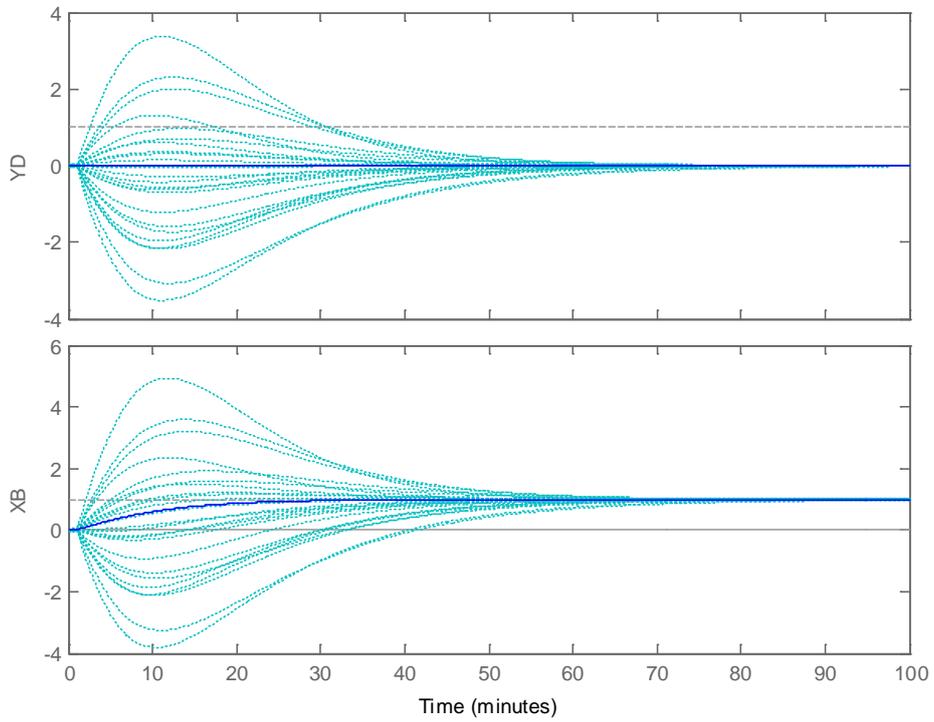


Figure 4-4: H-inf Mixed Sensitivity controller response to sce2

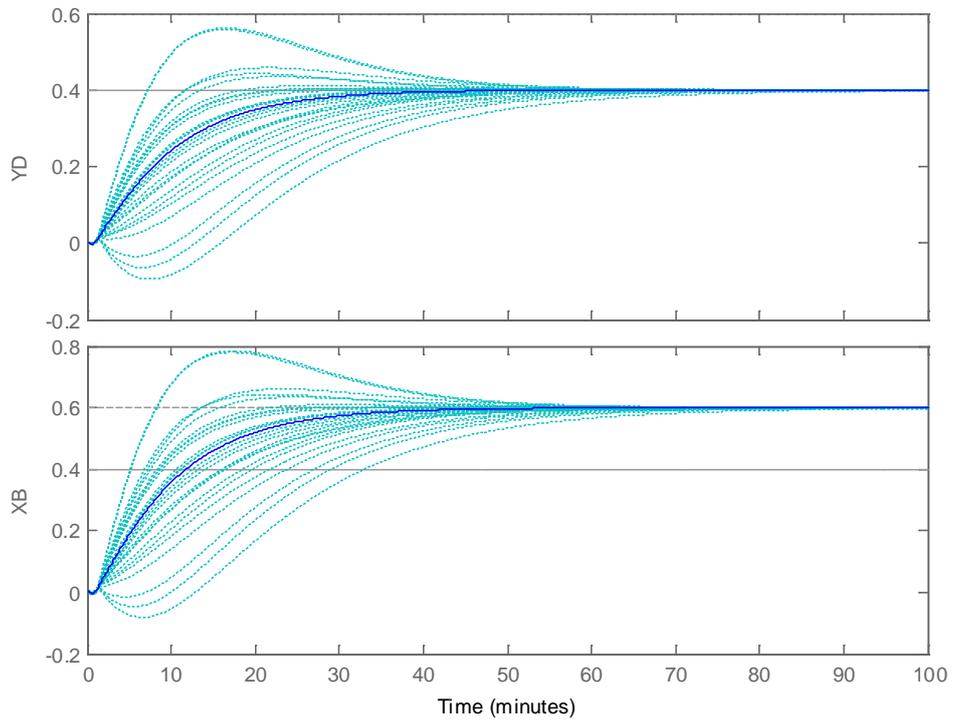


Figure 4-5: H-inf Mixed Sensitivity controller response to sce3

The singular value diagrams of S_o and T_o for a set of 25 Monte Carlo plants is shown in (Figure 4-6). It can be clearly appreciated the big variation of T_o with this controller with disturbances in the plant. It can be appreciated that for some plant the magnitude of T_o is the same order that the magnitude of S_oP that means that the influence of the input disturbances in the output (S_oP) is of the same order that the output itself (T_o).

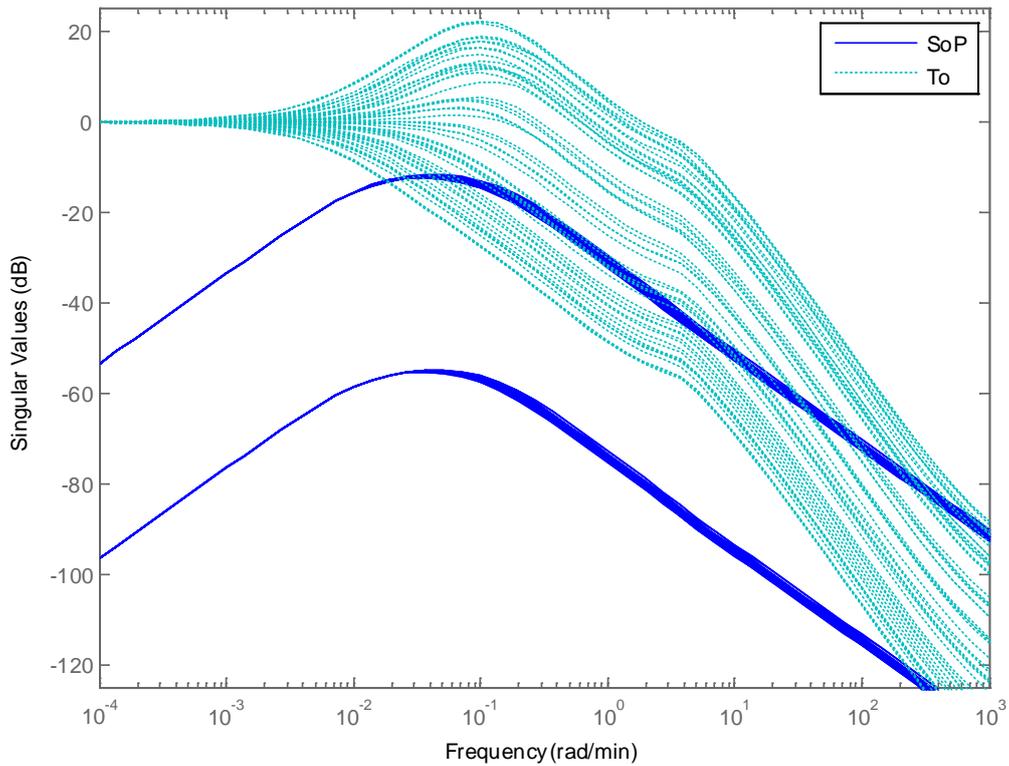


Figure 4-6: H-inf Mixed Sensitivity SoP and To

These results are in line with the predictions of (Christen, 1997). The controller has cancelled the poles of the plants with zeros (Figure 4-7). In presence of disturbances, the cancellation is not perfect and the control is not acceptable, provoking big cross-coupling and overshoots.

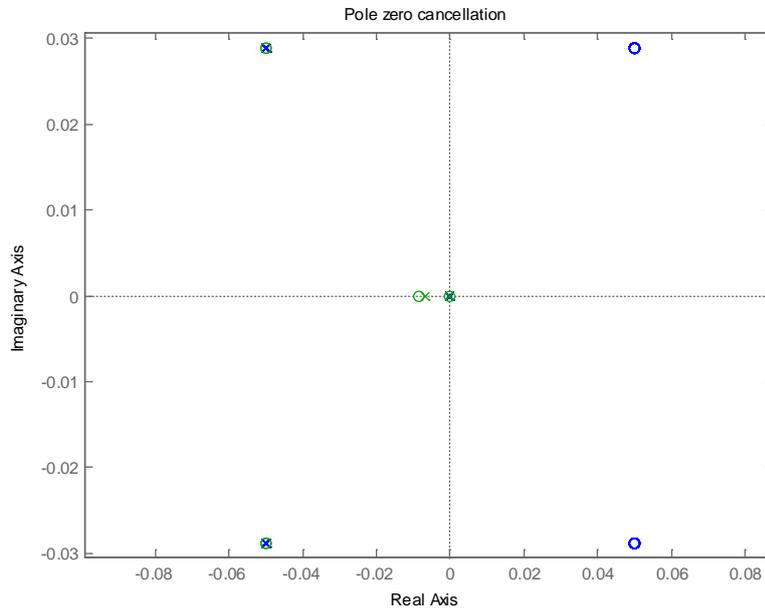


Figure 4-7: Pole zero cancellation in H-infinity Mixed Sensitivity controller

We shall remark that this cancellation effect happens also when plant delay is zero. The results of the benchmark with time delay = 0 seconds are shown in the following table. The specifications are not acceptable for g_{131} and g_{311} .

H-Infinity Mixed sensitivity (no delay)					
plant ch	set-point tracking			interaction	
plant	t=30	max	t=100	max	t=100
g223_u1	0.959	1.000	1.000	0.001	-0.000
g113_u1	0.914	1.000	1.000	0.001	-0.000
g133_u1	-1.233	5.815	1.001	8.161	0.001
g313_u1	3.302	5.901	1.028	6.627	0.036
g333_u1	0.982	1.000	1.000	0.001	-0.000

Table 4-2: S2 criteria for Mixed Sensitivity controller with no delay

4.3 Alternative H-Infinity methods for control of ill conditioned plants

We propose several optimizations to the H-Infinity theory for the management of ill conditioned plants. The first one is the Kwakernaak approach. The second one is the Christen-Geering approach. The third one is the H-Infinity structured approach.

We describe all of them in the following paragraphs.

4.3.1 Scaling of the plant

As said previously, scaling is fundamental in the H-Infinity theory.

For the CDC problem:

- Scaling of the references: it is not needed because the references are already in the range $[0..1]$ ($r_{\max} = 1$)
- Scaling of the measures: it is not needed because the outputs are already in the range $[0..1]$ ($y_{\max} = 1$)
- Scaling between channels: not needed because both channels are in the same units
- Scaling of the weighted signals: it is needed. See hereafter

We need to scale the transfer function KS_o . Our rationale for the choosing of the scaling is as follows:

- The design specification (S3) requires 'Control effort: $KS_o < 50 \text{ dB}$ '
- As KS_o is equal to $K/(1+PK)$, it reach its maximum value when P is small
- The plant with less gain is g_{111} and its minimum singular value is 0.0111
- $KS_o < 50 \text{ dB} = 316$;
- $K/(1+0.0111K) < 316$
- => maximum gain allowed is for K is about 126

With this, we will have a KS_o scaled output of 1 for the maximum allowed gain for the "input to controller" transfer function.

For taking into account the fact that the open loop response to the second channel is about a 25% greater than for the first channel we take the following scaling.

$$G_s = \begin{bmatrix} 100 & 0 \\ 0 & 126 \end{bmatrix} G_{nom}$$

The factor of 100 used by Skogestad is similar to our scaling and is justified by previous explanation. Note however that differently to Skogestad that uses directly the new "normalized" plant; we prefer to keep the plant proposed by the CDC benchmark "unchanged" and integrating the scaling in the synthesized controller, as explained in (Figure 4-8) and (Figure 4-9):

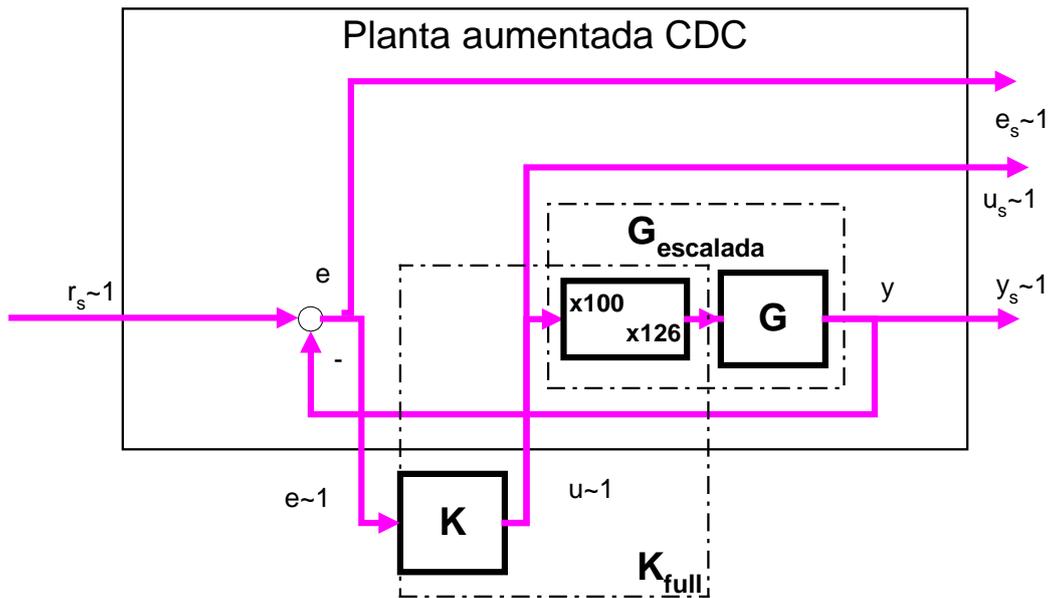


Figure 4-8: Scaling for the plant for the design

And after the controller is synthesized we integrate the scaling in the controller:

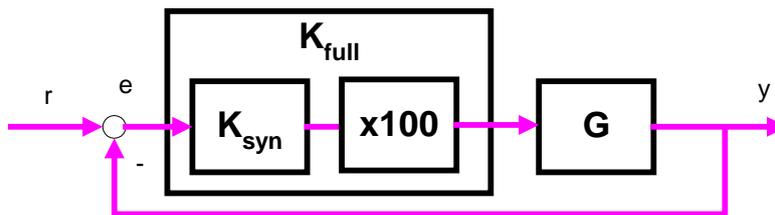


Figure 4-9: Integrating the scaling on the controller

4.3.2 Kwakernaak optimization for H-Infinity

This proposal was first described by Kwakernaak in (Kwakernaak, 1993) and further developed by Cao and Iori in (Cao, 1997).

The idea is to add a pre-filter V to the references that includes the dynamic of the plant i.e. has the same poles (Figure 4-10). The idea is that a perturbation affecting the plant will affect in the same way to the pre-filter and both perturbations 'will cancel each other'.

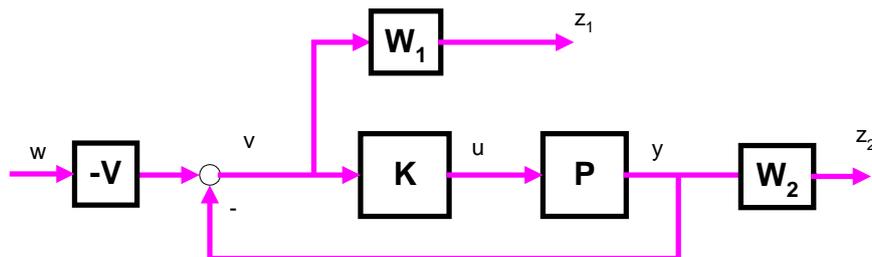


Figure 4-10: H-infinity controller with pre-filter

It shall be noted however that an unjustified 'license' is taken by the KWA approach: the pre-filter V is designed to have the same denominator as the plant, however, the inputs to the plant are the actuators (dL , dV) and the inputs to the pre-filter are instead the references on dX_B , dY_D .

We can write the TF from the inputs to the outputs of the augmented plant as:

$$\begin{aligned} z_1 &= W_1 v = W_1 (-Vw - Pu) = -W_1 Vw - W_1 Pu \\ z_2 &= W_2 u \\ v &= -Vw - Pu \end{aligned}$$

In matrix form:

$$\begin{bmatrix} z_1 \\ z_2 \\ v \end{bmatrix} = \begin{bmatrix} -W_1 V & -W_1 P \\ 0 & W_2 \\ -V & -P \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$$

That has a space state representation (the demonstration is provided in appendix 8.3.1.2):

$$P_{AU} = \left[\begin{array}{cc|cc|cc} A_V & 0 & 0 & 0 & B_V & 0 \\ 0 & A_P & 0 & 0 & 0 & B_P \\ \hline -B_1 C_V & -B_1 C_P & A_1 & 0 & -B_1 D_V & -B_1 D_P \\ 0 & 0 & 0 & A_2 & 0 & B_2 \\ \hline -D_1 C_V & -D_1 C_P & C_1 & 0 & -D_1 D_V & -D_1 D_P \\ 0 & 0 & 0 & C_2 & 0 & D_2 \\ -C_V & C_P & 0 & 0 & -D_V & -D_P \end{array} \right]$$

We can choose typical weights for S and T (described later). For the pre-filter V , the article (Cao, 1997) proposes to take $V = N / D$ where D is the denominator of the plant. For the numerator it is proposed to choose a simple weight $W_v = J * (s + \omega_B)^n$, where ω_B is the bandwidth frequency and n is appropriate for making the inverse of the transfer function proper.

However, the example proposed in (Cao, 1997) is a SISO system. In a MIMO case as the distillation column we should take a pre-filter as:

$$V = \frac{1}{75s+1} \begin{bmatrix} (s+0.005) & 0 \\ 0 & (s+0.005) \end{bmatrix}$$

When we attempt such design the results of the control are not acceptable because the directionality of the original plant has not been taken into account (i.e. the directionality

of V does not match the directionality of P and so does not match its behavior).

We can introduce directionality of the plant by including G_o (the static gain of the plant) in the pre-filter V

$$V = \frac{1}{75s+1} G_o \begin{bmatrix} (s+0.005) & 0 \\ 0 & (s+0.005) \end{bmatrix}$$

This improves the response, but the obtained control is not robust to simulations with time delay.

For taking into account the time delay, we may take the denominator of the nominal transfer function with delay and a third order zero in order to make the prefilter V proper.

$$V = \frac{1}{75s^3 + 8.5s^2 + 0.35s + 0.003333} G_o \begin{bmatrix} (s+0.005)^3 & 0 \\ 0 & (s+0.005)^3 \end{bmatrix}$$

The results improve (time delay simulations are now stable) but are not good for the extreme perturbations on the gains (cases $[k_1, k_2] = [0.8 \ 1.2], [1.2 \ 0.8]$).

A more straightforward alternative for choosing the pre-filter V is proposed on this work:

Let's take a singular value decomposition (SVD) of G_o :

$$[V, S, U] = SVD(G_o)$$

And define the pre-filter as an additive uncertainty Δ over P , but with the same directionality that the plant P , i.e.

$$V = P + V * \Delta * U \quad \text{where } \Delta \text{ is } [0.2 \ 0; 0 \ 0.2]$$

An example following this strategy will be presented later.

4.3.3 Christen – Geering approach

Christen and Geering proposed in (Christen, 1997) a variation over the standard problem. (We will name it “CHGE” approach hereafter). The fundamental point is that the transfer function S_oP (the transfer function from d_u to z_p , i.e. from disturbances at plant input to the plant outputs) is weighted. A pre-filter scalar weight W_d is included in the references (Figure 4-11).

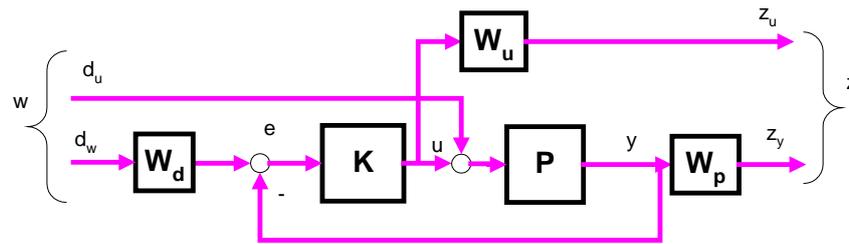


Figure 4-11: CHGE problem

Note that this approach manages the “a-priori” critic of Maciejowski to the H-Infinity method: that it does not take into account the open loop plant. In fact the CHGE approach is weighting S_oP : the plant P is taken into account explicitly.

The transfer matrix form for the augmented plant is:

$$\begin{bmatrix} z_u \\ z_y \end{bmatrix} = \begin{bmatrix} W_u T_i & W_u K S_o W_d \\ W_p S_o P & W_p T_o W_d \end{bmatrix} \begin{bmatrix} d_u \\ d_w \end{bmatrix}$$

The following state space representation is proposed by (Christen, 1997). (The demonstration is provided in appendix 8.3.1.3).

$$\left[\begin{array}{ccc|ccc} A_u & 0 & 0 & 0 & 0 & B_u \\ 0 & A_y & B_y C_P & 0 & 0 & 0 \\ 0 & 0 & A_P & B_P & 0 & B_P \\ \hline C_u & 0 & 0 & 0 & 0 & D_u \\ 0 & C_y & D_y C_P & 0 & 0 & 0 \\ 0 & 0 & -C_P & 0 & w_d & 0 \end{array} \right]$$

Choice of Weight W_d

The reference (Christen, 1997) proposes to take a weight $W_d \ll 1$. This made that the second column of transfer matrices (KS_o and S_o) do not contribute too much to the norm, so practically we are weighting only T_i and S_oP . This recommendation will be followed on our work.

Choice of Weight W_p for S_oP

The more important election is the weight for S_oP . The reference (Christen, 1997) proposes to take a weight made by two parts: one part for S_o and the other part for P .

For the S_o part of W_p , the common proposal in the literature is to take a low pass filter:

$$W_{S_o} = \frac{\frac{s}{s + \omega_B} + \omega_B}{s + \omega_B A}$$

We propose a more direct approach using an 'ideal 2nd order system'

- Choose a 2nd order system that meets the specifications (chose ω_n and delta)
 - o The TF of the system is: $T_{o_id} = \omega_n^2 / (s^2 + 2 \text{ delta } \omega_n s + \omega_n^2)$
- Calculate $S_{o_id} = 1 - T_{o_id}$
- Define the weight as the inverse of S_{o_id}

Note: A practical point is that usually the algorithm H-infinity implemented in Matlab works better is the weight has the same number of poles and zeros (is strictly proper). For this we can add some zeros far from the bandwidth. We can do this simply by computing $S_{o_id} = 1.00001 - T_{o_id}$

For the P part of the W_p , the reference (Christen, 1997) warns about taking the inverse of P itself because in such case, the controller will include the inverse of P reproducing the pole-zero cancellation problem. Instead, the article proposes to take the steady gain of P : P_0 ($\neq G_0$).

There are two possible strategies about the choice of P_0 that are explored now:

- Which P should be chosen for taking its steady gain P_0 ? (e.g., should be taken P_0 of g_{22x} , or of g_{13x} , or of g_{33x} ...?).
- Or instead, should we take P_0 from g_{22x} (the nominal plant) and add to this nominal static plant a model of the uncertainty?

Variant 1 for the choice of P_0 : which static plant to choose from the uncertainty set?

As said, it could be P_0 for g_{22x} or P_0 for g_{13x} , etc. (Note that the delay does not play any role on the steady gain). A possible approach is to take the 'averaged P_0 plant'. For example we could do:

Step 1: Obtain a singular value decomposition for the set of plants: $[U, S, V] = \text{svd}(P_0 g_{xx})$

- S are the singular values of P_0 (in a 2 x 2 system, the maximum and minimum gains for any input),
- V is made of 2 column vectors that define the inputs for which the maximum and minimum gains are achieved
- U is made of 2 column vectors that define the values of the outputs for the inputs V

(Note that S and V depend on each plant of the uncertainty set, but U is the same for all the plants).

Step 2: Then, take the averaged values of V , S and U (by columns)

The exercise with the distillation plant and the G_0 for the set of plants g_{22x} , g_{11x} , g_{13x} , g_{31x} , g_{33x} obtains the following results:

$$U_{AVG} = \begin{bmatrix} -0.6246 & -0.7809 \\ -0.7809 & 0.6246 \end{bmatrix}$$

$$S_{AVG} = \begin{bmatrix} 1.9877 & 0 \\ 0 & 0.0136 \end{bmatrix}$$

$$V_{AVG} = \begin{bmatrix} -0.7011 & -0.7021 \\ 0.7021 & -0.7011 \end{bmatrix}$$

That we can compare with the U, S, V decomposition of the nominal plant g_{22x} :

$$U_{g_{22x}} = \begin{bmatrix} -0.6246 & -0.7809 \\ -0.7809 & 0.6246 \end{bmatrix}$$

$$S_{g_{22x}} = \begin{bmatrix} 1.9721 & 0 \\ 0 & 0.0139 \end{bmatrix}$$

$$V_{g_{22x}} = \begin{bmatrix} -0.7066 & -0.7077 \\ 0.7077 & -0.7066 \end{bmatrix}$$

The previous analysis demonstrates that the U, S, V matrices of the 'averaged P_0 ' are close to the values of P_0 for g_{22x} (the nominal plant). This implies that (at least on the case of the distillation plant), we can simply take as static gain P_0 the one of g_{22x} .

Variant 2 for the choice of P_0 : take P_0 from g_{22x} and add to this nominal static plant a uncertainty model

What is proposed is to take the part of the weight for P as the inverse of a multiplicative perturbation (Δ) over P_0 :

$$W_{p(part)} = inv(P_0 * (1 + \Delta))$$

A point still to be decided is what perturbation Δ shall be taken: for example, we can take $\Delta = [0.2 \ 0; \ 0 \ 0.2]$ or $\Delta = [-0.2 \ 0; \ 0 \ -0.2]$ or $\Delta = [-0.2 \ 0; \ 0 \ 0.2]$...

Preliminary simulations demonstrate that an obvious election as $\Delta = [0.2 \ 0; \ 0 \ 0.2]$ may be not the optimum value.

A more systematic way is proposed:

- Build a set of uncertainties $P_0 * (1 + \Delta)$, where $\Delta = [\Delta_1 \ 0; \ 0 \ \Delta_2]$ and $\Delta_i \in [-0.2, 0.2]$
- Plot the sigma diagrams of this set
- Find the case with maximum singular values and gets its Δ_1, Δ_2

Summarizing this paragraph, we propose to choose a weight W_p equal to:

$$W_p = \underbrace{\text{inv}(P_o(1 + \Delta))}_{\text{part } (P)} \underbrace{\text{inv} \begin{pmatrix} 1 - \frac{w_n^2}{s^2 + 2\delta w_n s + w_n^2} & 0 \\ 0 & 1 - \frac{w_n^2}{s^2 + 2\delta w_n s + w_n^2} \end{pmatrix}}_{\text{part } (S_o)}$$

Where:

- The left term $\text{inv}(P_o * (1 + \Delta))$ is the part or the weight for P and the right term is the part of the weight for S_o
- For the variant 1 (let's name it 'CHGE pure'), $\Delta = [\text{zero}]$ and P_o is the steady gain of the nominal plant
- For the variant 2, $\Delta = [\Delta_1 \ 0; \ 0 \ \Delta_2]$ with Δ_1, Δ_2 the values of uncertainty that maximize the singular values of $P_o * (1 + \Delta)$

Choice of Weight W_u for T_i

The guidelines for the weight for T_o proposed in the literature are to select the weight as a high gain filter with bandwidth a bit greater than the desired bandwidth of the closed loop system. As T_o trends to match the inverse of W_u , T_o will present a slope after the bandwidth so achieving a good resistance to unstructured uncertainties and noise at the plant output.

Note that on MIMO, T_i is in general not equal to T_o , but in principle they are similar. We can follow the same strategy for the S_o part of W_p : choose a weight T_i as the inverse of an ideal system T_{o_id} . The weight needs to be a proper transfer function so we have to add zeros.

Another point shall be taken into account in the CHGE approach, related to the scaling of the transfer functions. The H-Infinity optimization algorithm assumes that the terms being weighted have a similar magnitude (if not, only the term with the greater magnitude contributes significantly to the overall weighted value). Previously we have introduced a factor similar to $1/P_o$ in the weight W_p . This means that the TF for W_p is $1/P_o$ times smaller than the term for W_u . We should introduce a similar factor in the weight for W_u , restoring the situation where both weighted TF contribute equivalently to the norm.

In the particular problem of the distillation column, however, we cannot introduce as factor P_o itself because as P_o is ill conditioned we *would pass* this ill conditioning to T_i . Instead, we use a constant diagonal scalar factor 'A' in W_u equal to the maximum singular value of P_o .

Summarizing, W_u is chosen as:

- Take a 2nd order ideal system that meets the specifications (chose ω_n and delta)
- Multiply by a factor ' $A = \max(\text{svd}(P_o))$ '
- Choose the weight as $\text{inv}(A * T_{o_id})$
- Include zeros to made the weight proper

4.3.4 Apkarian structured approach

The theoretical fundamentals of the method have been sketched previously in chapter 3.5.1.2. Now let's explain the practical approach:

The concept of "structure of the controller" supposes that the controller has a state space representation as:

$$\begin{bmatrix} \dot{x}_K \\ u \end{bmatrix} = \begin{bmatrix} A_K(\theta) & B_K(\theta) \\ C_K(\theta) & D_K(\theta) \end{bmatrix} \begin{bmatrix} x_K \\ Y \end{bmatrix}$$

And the real matrices A_K, B_K, C_K, D_K depends smoothly on a design parameter $\theta \in \mathbb{R}^n$. The vector θ is referred as the vector of tunable parameters.

Example 1: for a (realizable) PID the tunable parameters are (T, K_P, K_D, K_I) being the transfer function and space state:

$$C(s) = K_p + \frac{K_i}{s} + \frac{K_d s}{Ts + 1} \quad C(s) = \left[\begin{array}{cc|c} 0 & 0 & K_i \\ 0 & -\frac{1}{T} & -K_d T^2 \\ \hline 1 & 1 & K_p + \frac{K_d}{T} \end{array} \right]$$

Example 2: for a decentralized controller, the transfer function and space state is:

$$C(s) = \begin{bmatrix} C_1 & \dots & 0 \\ \dots & C_i & \dots \\ 0 & \dots & C_n \end{bmatrix} \quad C(s) = \left[\begin{array}{c|c} A_{K1} & B_{K1} \\ \hline & B_{Kn} \\ A_{Kn} & \\ \hline C_{K1} & D_{K1} \\ & \\ & C_{Kn} & D_{Kn} \end{array} \right]$$

Note how the H-Infinity structured works with a typical example where we want to weight S_o, KS_o and S_oP :

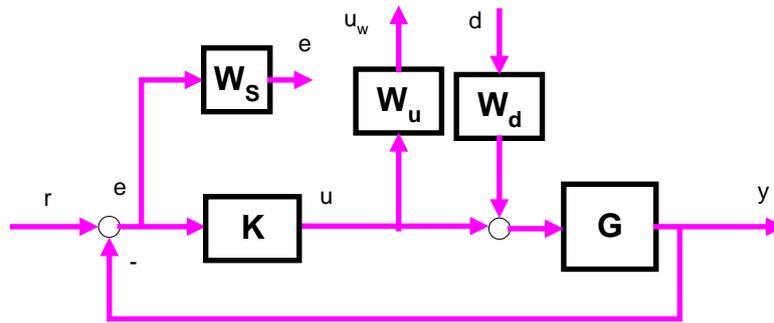


Figure 4-12: A typical example for the H-infinity Structured

First, the augmented plant (Figure 4-12) is defined as:

- Having as inputs the exogenous signals (i.e. the reference and the disturbances)
- Having as outputs the weighted signals (i.e. the error, the measured outputs and the control signals)

Then the diagonal augmented plan with weights is formed as:

$$\text{OutputWeights} * P * \text{InputsWeights}$$

- Inputs weights are a diagonal transfer matrix composed of an identity matrix for the references (that are not weighted) and W_d .
- Output weights are a diagonal transfer matrix, composed of W_p , an identity matrix for the measured signals (that are not weighted) and W_u .

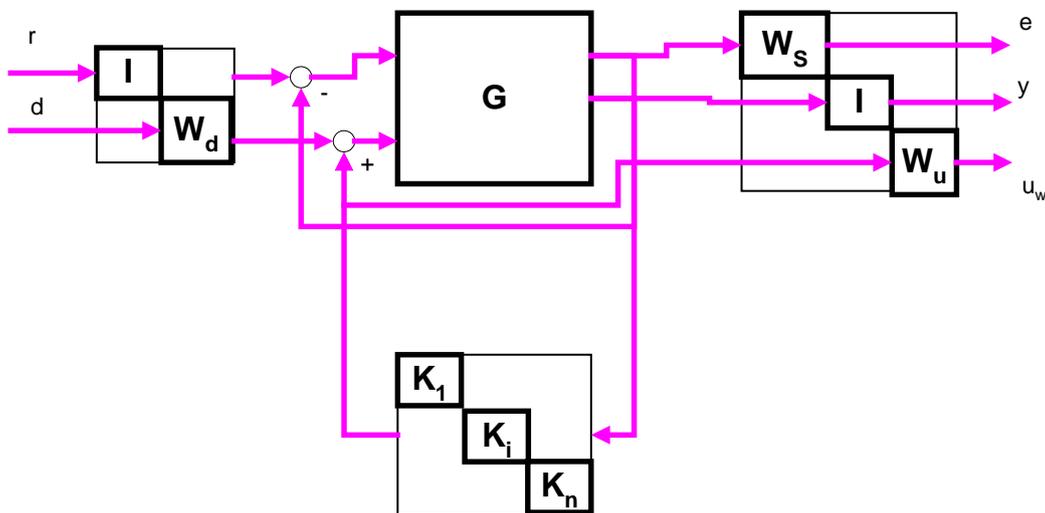


Figure 4-13: The Structured controller in diagonal view

4.4 CDC Benchmarks results

We will confront here the following controllers for the CDC benchmark:

- Reference μ controllers:

- The μ controller in (Skogestad, 1996)
- The optimized μ controller (Lundstrom, 1999)
- A μ controller elaborated from (Balas, 2012)
- The new H-infinity controllers:
 - A H-Infinity Kwakeernaak controller
 - A H-infinity Christeen Geering controller
 - A H-infinity Structured (Apkarian) controller

4.4.1 Reference μ controller (SKO96)

We use as a first reference the μ controller given in (Skogestad, 1996). We have synthesized the controller using the Matlab script 'sec8_124.m' provided in the reference (Skogestad, 1996). It shall be noted that the Matlab script uses a filter $T=1/(5s+1)$ on the reference signal when performing the simulations. This assumption is not part of the original CDC benchmark problem so we provide here the results without filtering the step reference.

The number for the criteria S2 is given in the following tables. The rise time is never met and the cross-coupling is always greater than allowed, in particular for plants g_{313} , g_{133} . The steady error limits are also violated but the deviations are in general minor. In general the control presents to big overshoots.

plant ch	set-point tracking			interaction		
Scenario 1						
plant	t=30	max	t=100	max	t=100	
g223_u1	0.660	0.930	0.930	0.770	0.057	
g113_u1	0.599	0.888	0.888	0.735	0.091	
g133_u1	0.613	0.924	0.924	0.551	0.059	
g313_u1	0.688	0.923	0.923	1.051	0.067	
g333_u1	0.699	0.957	0.957	0.826	0.035	
Scenario 2						
plant	t=30	max	t=100	max	t=100	
g223_u2	0.785	0.987	0.954	0.770	0.056	
g113_u2	0.731	0.939	0.927	0.735	0.089	
g133_u2	0.818	1.212	0.953	0.948	0.061	
g313_u2	0.740	0.946	0.946	0.631	0.062	
g333_u2	0.805	1.052	0.972	0.826	0.034	
Scenario 3						

plant	t=30	max	t=100	max	t=100
g223_u3	0.434	0.718	0.406	0.900	0.595
g113_u3	0.420	0.684	0.409	0.857	0.593
g133_u3	0.437	0.734	0.406	0.923	0.595
g313_u3	0.432	0.702	0.406	0.877	0.595
g333_u3	0.426	0.768	0.403	0.962	0.597

Table 4-3: S2 criteria for SKO96 controller

The responses to a Monte Carlo simulation to the scenario 1, 2 and 3 for a set of 25 plants with maximum delay are shown in (Figure 4-14), (Figure 4-15) and (Figure 4-16).

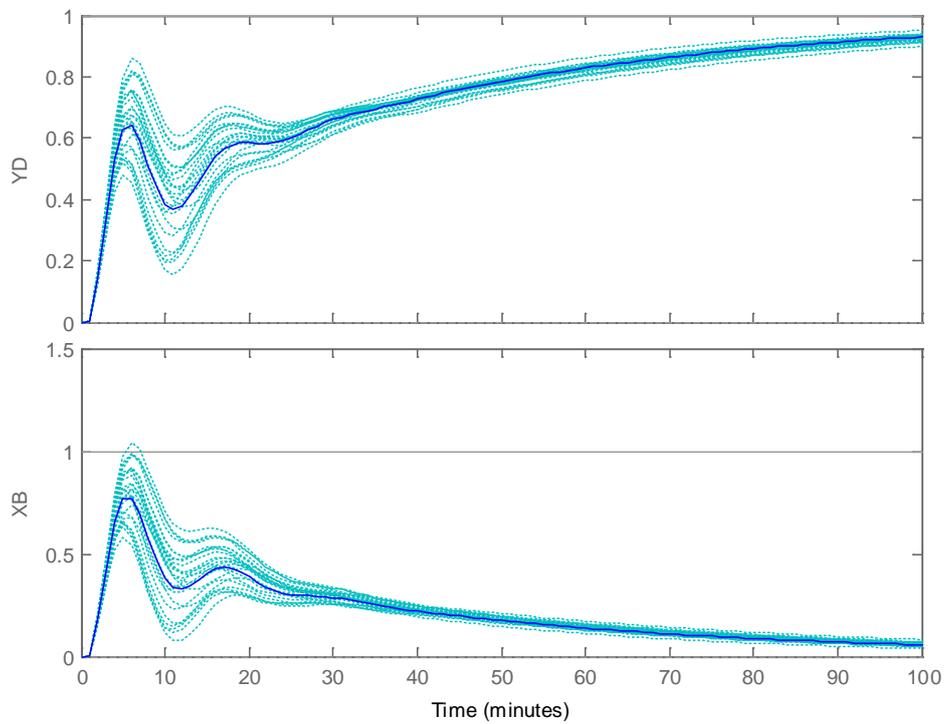


Figure 4-14: SKO96 mu controller response to sce1

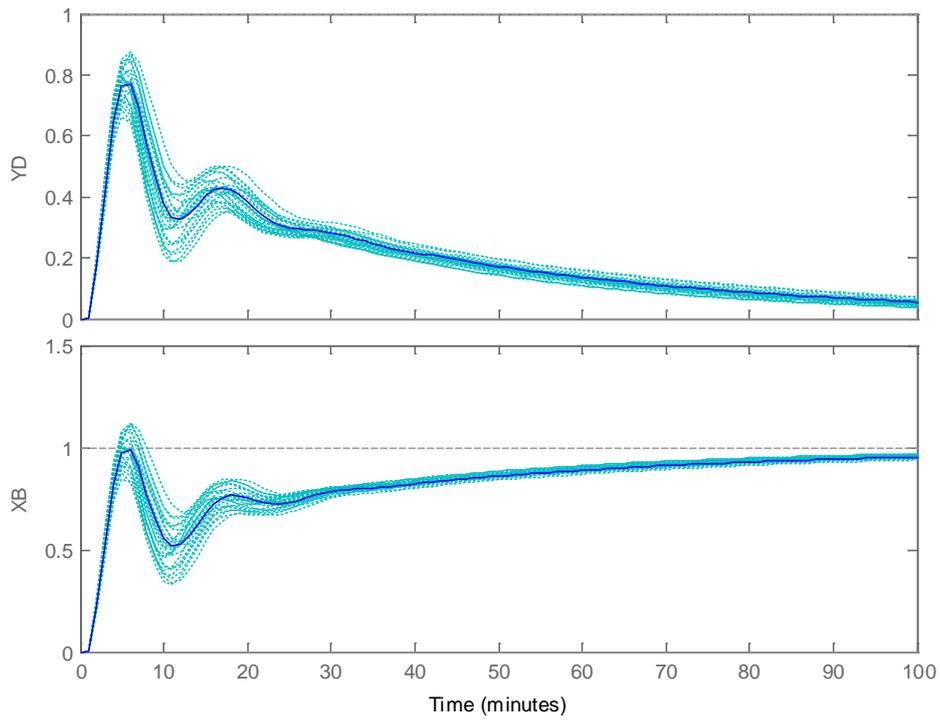


Figure 4-15: SKO96 mu controller response to sce2

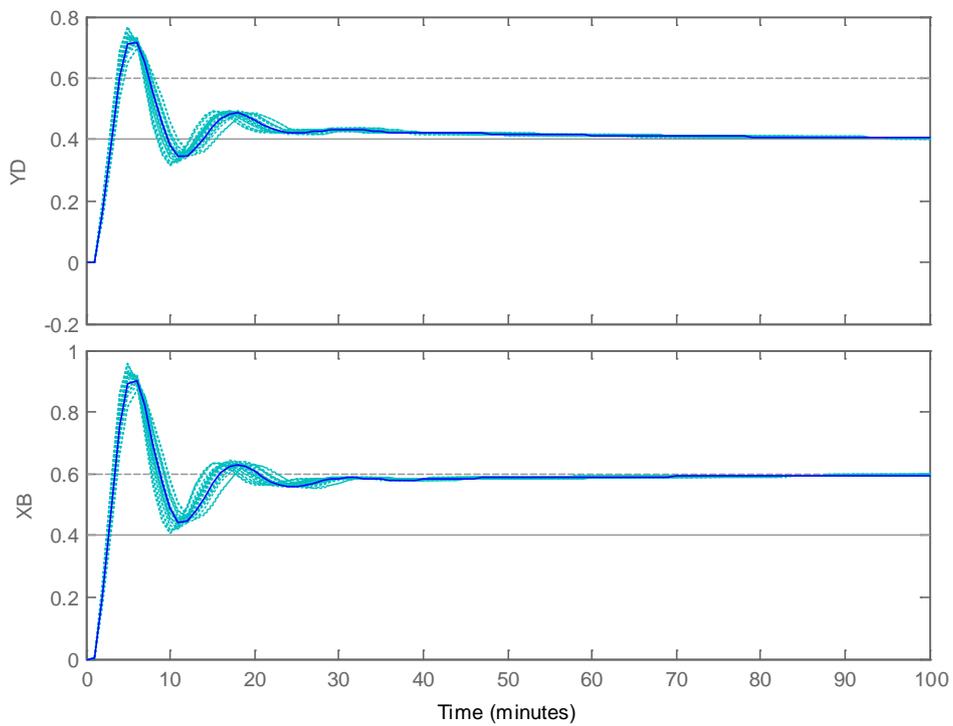


Figure 4-16: SKO96 mu controller response to sce3

The lack of robustness of the design is shown in (Figure 4-17) with big peaks in S_oP , T_o .

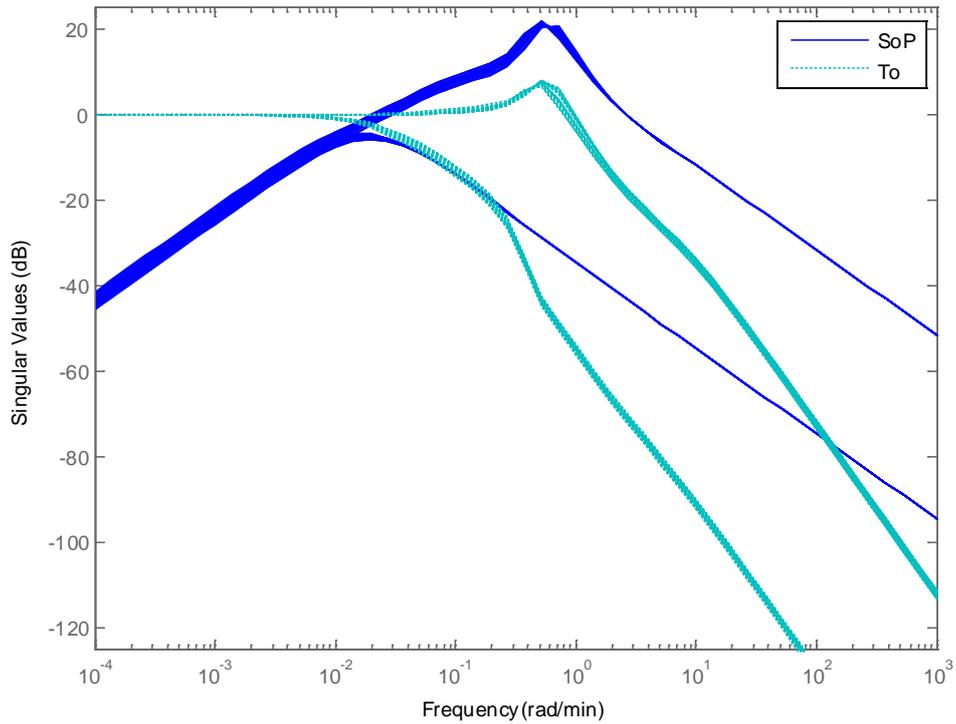


Figure 4-17: SKO96 mu controller S_oP and T_o

The specification S3 ($KS_o < 50 \text{ dB}$) is fulfilled. The specification S4 ($KS_o = 0.6 < 1$ for $\omega > 105 \text{ rad/min}$) is fulfilled (Figure 4-18).

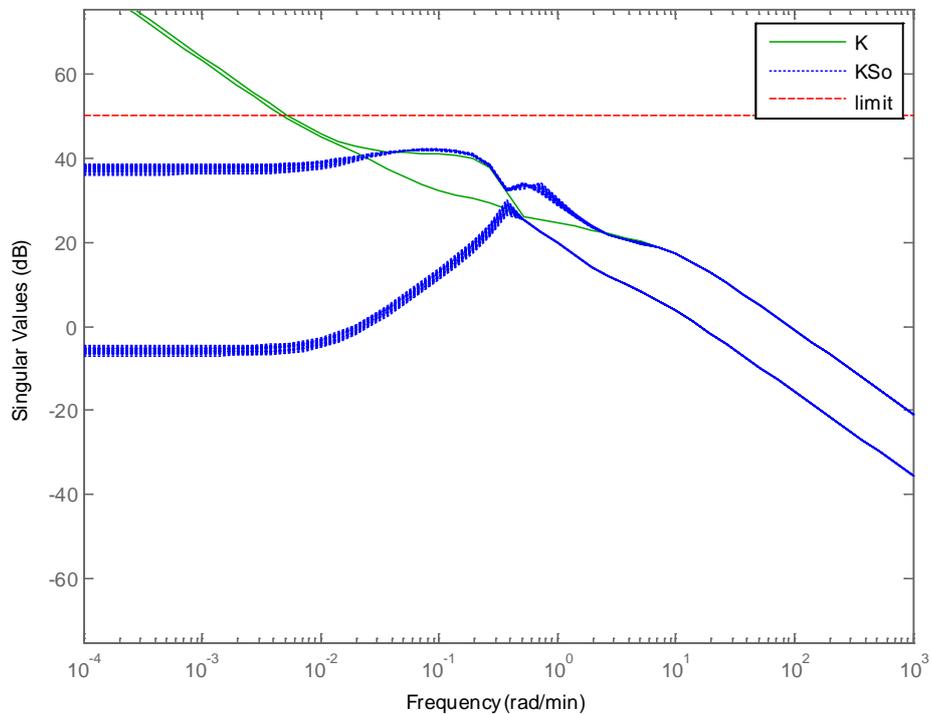


Figure 4-18: SKO96 mu controller K and KS_o

4.4.2 Second reference μ controller (LUND99)

We use as a second reference the first μ controller given in (Lundstrom, 1999). The results presented in the original article satisfy the criteria of the CDC benchmark. We have synthesized the controller using the Matlab script 'sec8_124.m' provided in the reference (Skogestad, 1996) with weights of Lundstrom because we are not able to reproduce exactly the results of the article. (We guess that the article uses also a filter on the reference. As the CDC benchmark does not allow a filter on the reference, we have not included it).

The results for the criteria S2 are shown hereafter.

plant ch	set-point tracking			interaction		
Scenario 1						
plant	t=30	max	t=100	max	t=100	
g223_u1	0.850	1.004	1.001	0.595	-0.001	
g113_u1	0.773	0.998	0.998	0.583	0.001	
g133_u1	0.691	1.017	1.001	0.482	-0.000	
g313_u1	0.978	1.030	1.000	0.959	-0.001	
g333_u1	0.920	1.025	1.000	0.609	0.000	
Scenario 2						
plant	t=30	max	t=100	max	t=100	
g223_u2	0.905	1.002	1.000	0.595	-0.001	
g113_u2	0.856	0.999	0.999	0.583	0.001	
g133_u2	1.038	1.170	1.000	0.761	-0.001	
g313_u2	0.753	1.019	1.001	0.523	-0.000	
g333_u2	0.949	1.016	1.000	0.610	0.000	
Scenario 3						
plant	t=30	max	t=100	max	t=100	
g223_u3	0.413	0.551	0.400	0.689	0.600	
g113_u3	0.419	0.542	0.400	0.679	0.600	
g133_u3	0.425	0.558	0.400	0.701	0.600	
g313_u3	0.403	0.544	0.400	0.679	0.600	
g333_u3	0.407	0.565	0.400	0.708	0.600	

Table 4-4: S2 criteria for the LUND99 controller

This controller presents much better results than previous. The rise time is violated in

some case but the violation is small except for plant g_{133} and g_{313} . The maximum is only violated for g_{133} in scenario 2. The stationary error is always fulfilled. The worst behavior is on the cross-coupling than exceeds the allowed values. Violation is only important for g_{313} in scenario 1.

The responses to a Monte Carlo simulation to the scenario 1, 2 and 3 for a set of 25 plants with maximum delay are shown in (Figure 4-19), (Figure 4-20) and (Figure 4-21):

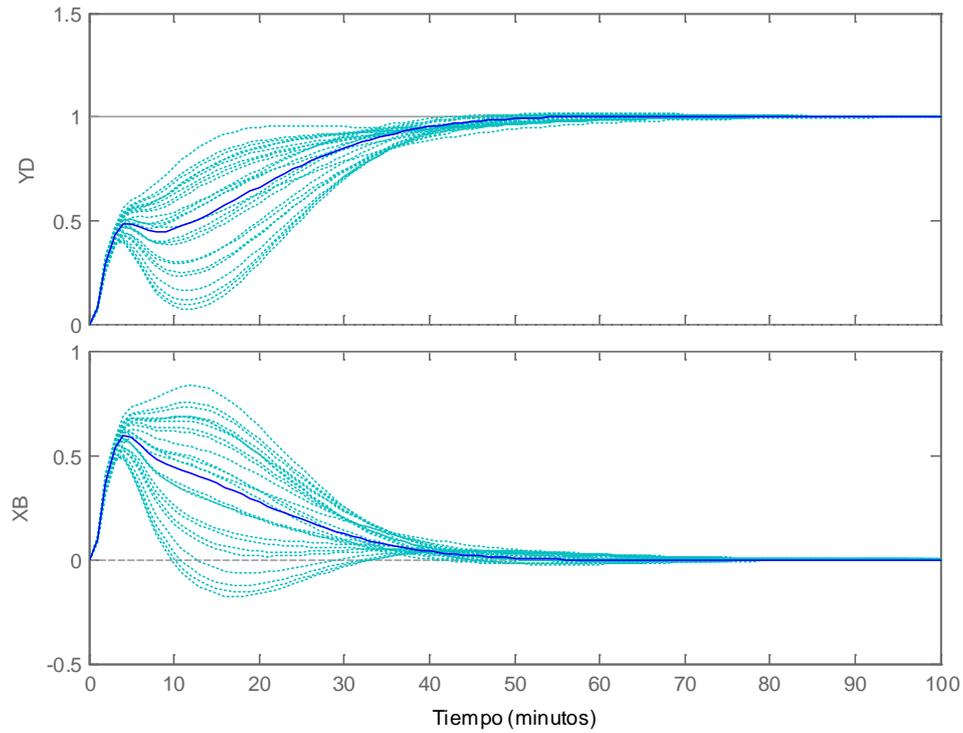


Figure 4-19 LUND99 mu controller response to sce1

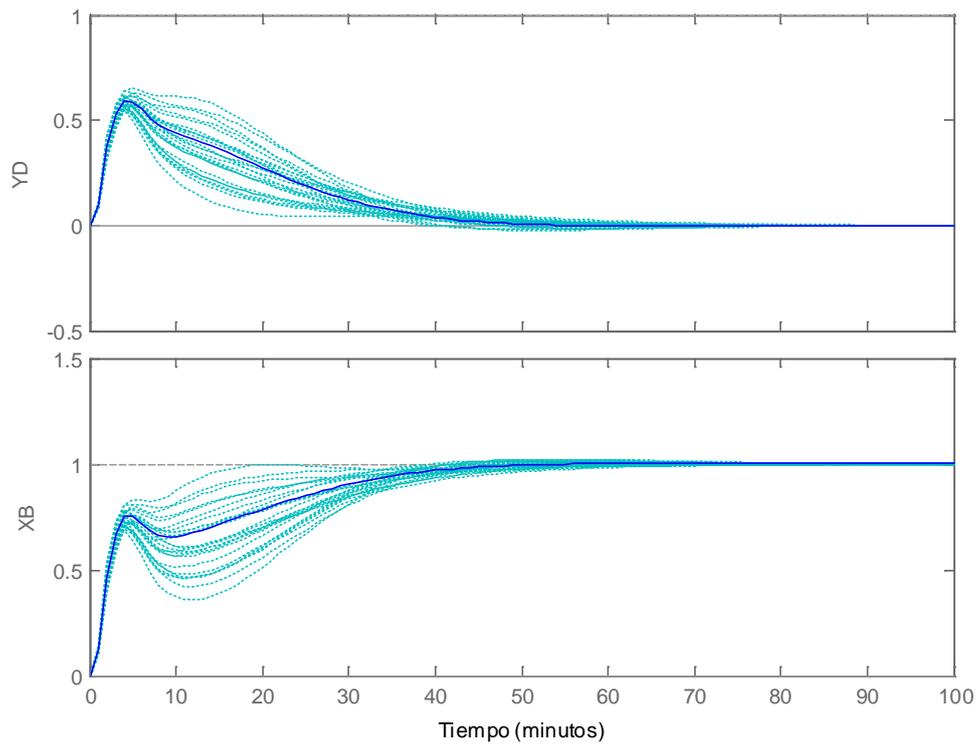


Figure 4-20 LUND99 mu controller response to sce2

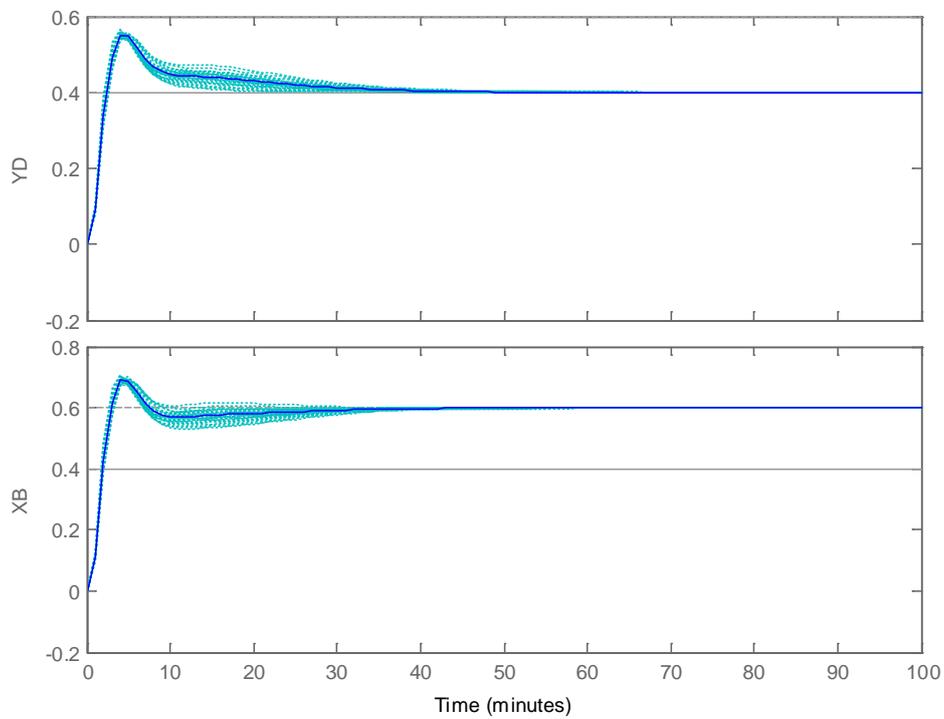


Figure 4-21: LUND99 mu controller response to sce3

Note however that the controller does not fulfill the criteria S3 and S4. At high frequencies the maximum gain allowed is surpassed (Figure 4-22).

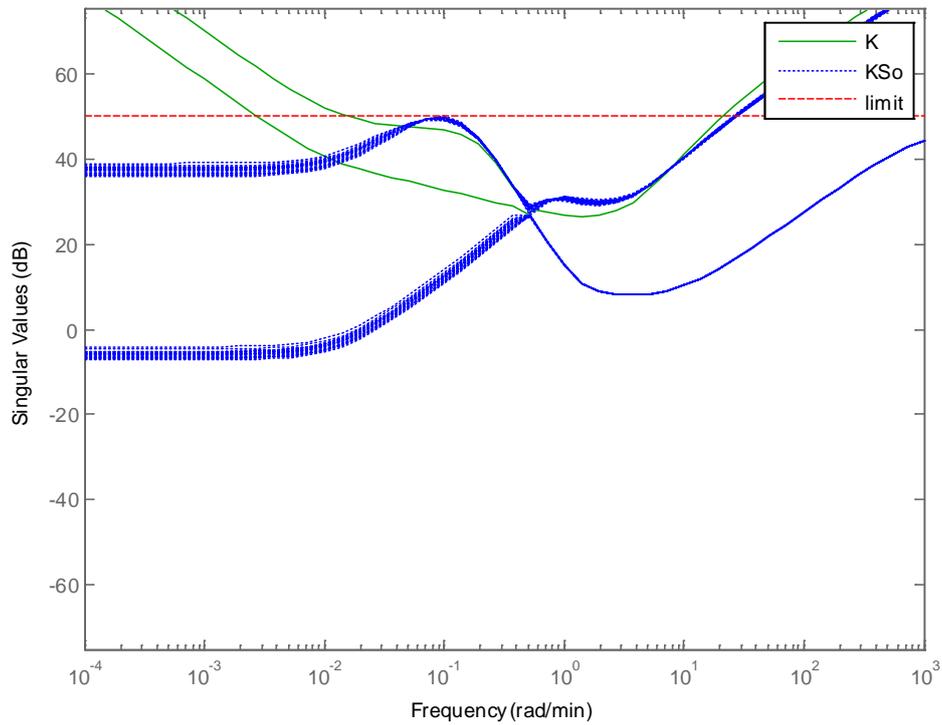


Figure 4-22: LUND99 mu controller K and K_{S_o}

The singular values diagram of S_oP and T_o demonstrate the robustness of the plant (Figure 4-23). The maximum singular value of S_oP is always well below the maximum singular value of T_o .

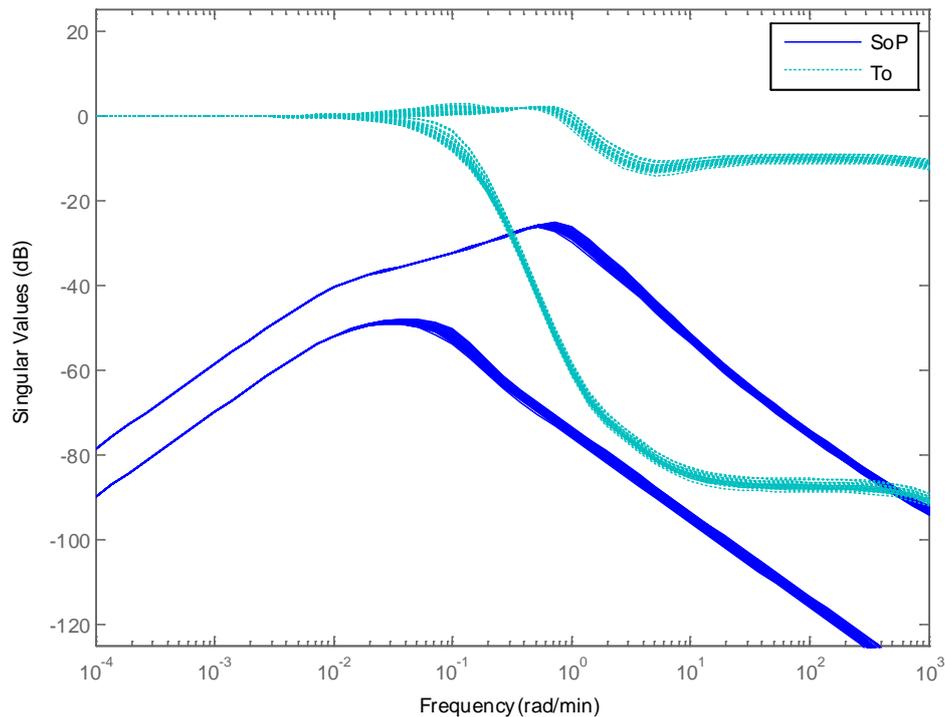


Figure 4-23: LUND99 mu controller SoP and To

4.4.3 Third reference μ controller (Balas 2012)

This controller is a development by the author elaborated from several ideas and examples proposed in a course by (Balas, 2012) at the European Space Agency (ESTEC). The interest of this controller is basically to check the improvement of the Matlab Robust Control μ toolbox routines along last 10 years.

The weighting strategy is unchanged with previous μ design. A weight W_d models the actuators uncertainty with an error of 20% at low frequency and 200% at high frequency. The weight is taken from (Lundstrom, 1999)

$$W_d = \begin{pmatrix} 0.2 \frac{5s+1}{2.5s+1} & 0 \\ 0 & 0.2 \frac{5s+1}{2.5s+1} \end{pmatrix}$$

A second weight W_p is derived from an ideal second order model that fulfills the specifications:

- $\omega_n = 10/75$ (rad/min) ; $\delta = 1.0$; ('ten times faster than the open loop')

A 'LTI uncertain object' is created and added to the perturbed plant with:

```
Delta = ultidyn('Delta',[2 2]);
GPert = G * (ss(eye(2))+ Delta *WD);
```

The system is formed as shown in (Figure 4-24).

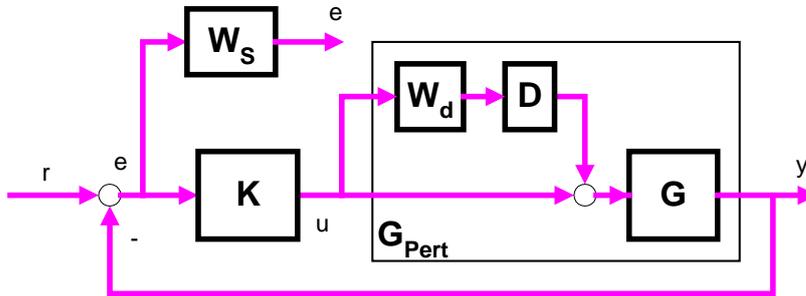


Figure 4-24: block diagram with controller Balas 2012

The μ toolbox function `dksyn()` is invoked. The generated controller has 32 states.

Note: The new `dksyn()` function is able to find the controller in seconds (as opposite with previous version of μ toolbox that can take hours).

The results for the criteria S2 are shown in the following tables

plant ch	set-point tracking			interaction	
Scenario 1					
plant	t=30	max	t=100	max	t=100
g223_u1	0.852	1.002	1.002	0.573	-0.002
g113_u1	0.779	0.997	0.997	0.575	0.001
g133_u1	0.835	1.025	0.997	0.318	-0.007
g313_u1	0.840	1.130	1.006	1.216	0.003
g333_u1	0.916	1.009	1.002	0.571	-0.003
Scenario 2					
plant	t=30	max	t=100	max	t=100
g223_u2	0.906	1.001	1.001	0.573	-0.002
g113_u2	0.859	0.998	0.998	0.575	0.001
g133_u2	0.900	1.337	1.005	0.964	0.001
g313_u2	0.895	1.036	0.997	0.334	-0.005
g333_u2	0.947	1.006	1.001	0.571	-0.003
Scenario 2					
plant	t=30	max	t=100	max	t=100
g333_u2	0.947	1.006	1.001	0.571	-0.003
g223_u3	0.413	0.552	0.399	0.697	0.600
g113_u3	0.419	0.557	0.400	0.704	0.599
g133_u3	0.414	0.587	0.400	0.743	0.600
g313_u3	0.414	0.518	0.399	0.652	0.599
g333_u3	0.407	0.547	0.399	0.690	0.600

Table 4-5: S2 criteria for the Balas 2012 controller

This controller is quite comparable to the LUND99 controller. The rise time is violated in some cases but the violation is small except for plant g_{133} and g_{313} . The maximum is only violated for g_{313} in scenario 1 and g_{133} in scenario 2. The stationary error is always fulfilled. The worst behavior is on the cross-coupling than exceeds the allowed values. Violation is only important for g_{313} in scenario 1.

The responses to a Monte Carlo simulation for the scenario 1 and 2 for a set of 25 plants with maximum delay are shown in (Figure 4-25), (Figure 4-26) and (Figure 4-27).

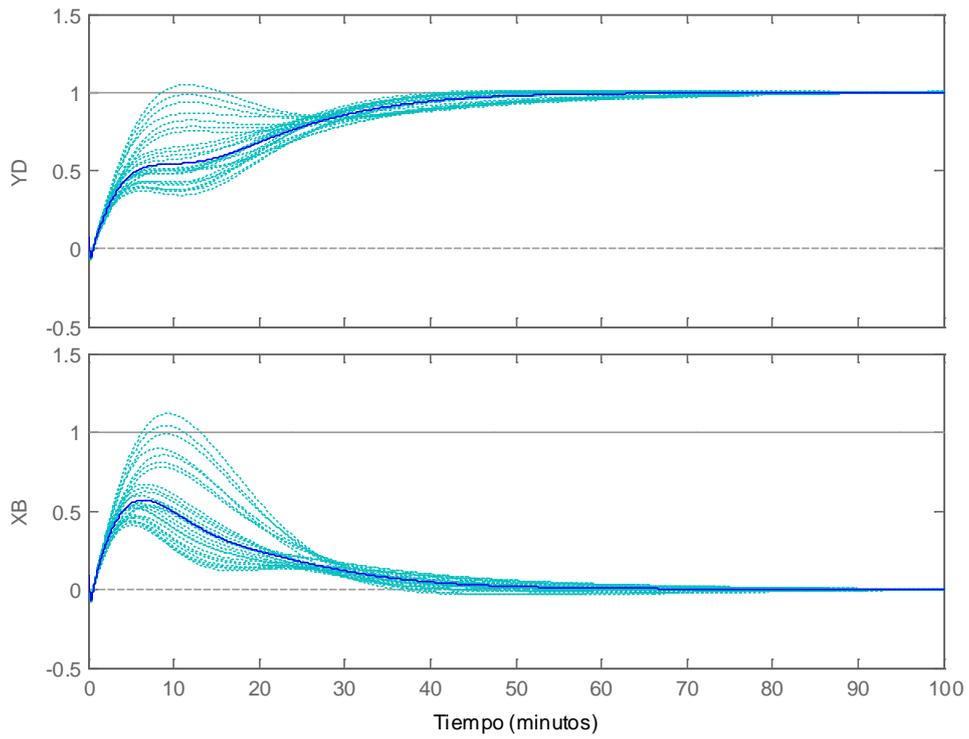


Figure 4-25 Balas 2012 mu controller response to sce1

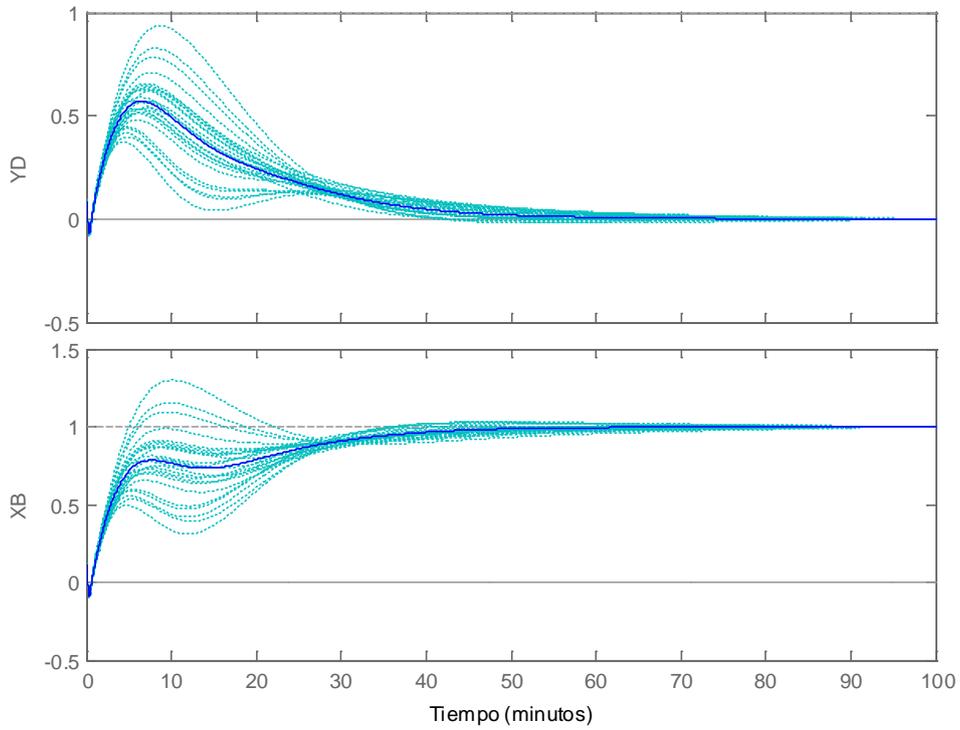


Figure 4-26 Balas 2012 mu controller response to sce2

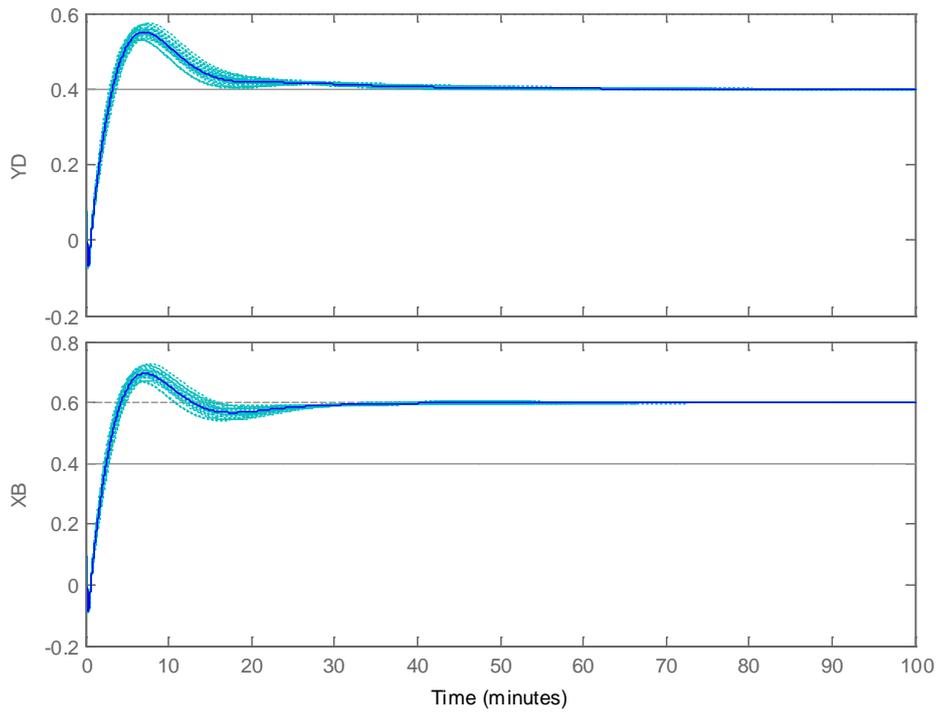


Figure 4-27: Balas 2012 mu controller response to sce3

The specification S3 ($KS_o < 50 \text{ dB}$) is fulfilled. The specification S4 ($KS_o = < 1 \text{ for } \omega > 105 \text{ rad/min}$) is not fulfilled ($KS_o = 100$) (Figure 4-28).

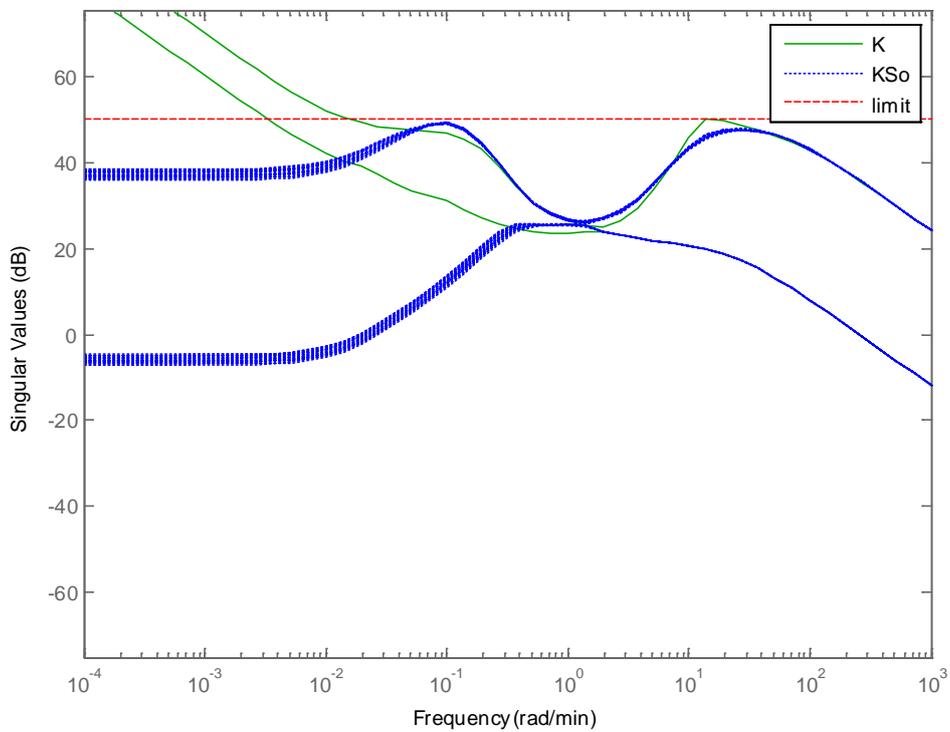


Figure 4-28: Balas 2012 mu controller KS_o

The singular values diagram of S_oP and T_o demonstrate the robustness of the plant (Figure 4-29). The maximum singular value of S_oP is always well below the maximum singular value of T_o .

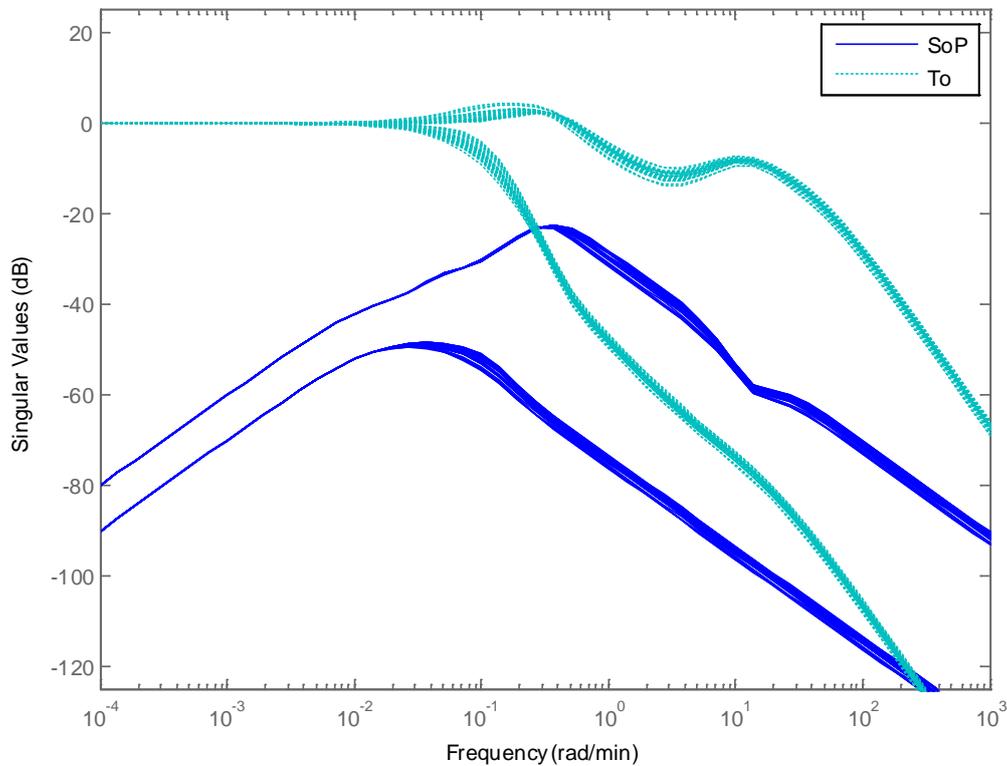


Figure 4-29: Balas 2012 μ controller S_oP and T_o

As summary, the controller LUND99 and the controller Balas2012 controller perform similarly. The controller LUND99 has 33 states and the controller Balas2012 has 32 states so are almost identical in term of complexity.

A big difference however is the degree of tuning needed. The authors of LUND99 recognize that the controller tuning required a lot of effort (a lot of hours for tuning). The Balas controller designed in this chapter has good results without requiring extensive tuning (less than 2-3 hours).

The previous results testimonies the big improvement on the μ toolbox in the last years.

4.4.4 H-Infinity Kwakeernaak (KWA) controller

We follow the guidelines proposed before for the selection of weight for S_o . We select as ideal T_o system a second order system with:

- $\omega_n = 0.0075$; delta = 1.5; (channel 1)
- $\omega_n = 0.010$; delta = 1.5; (channel 2)
- (i.e. the channel 2 is slightly faster)

$$W_p = \text{inv} \begin{pmatrix} \frac{s^2 + 0.0225s}{s^2 + 0.0225s + 5.625e - 005} & 0 \\ 0 & \frac{s^2 + 0.003s}{s^2 + 0.003s + 0.0001} \end{pmatrix}$$

The literature suggests a high pass filter for the KS_o weight (so limiting the control effort at high frequencies). We select:

$$W_u = \begin{pmatrix} \frac{5s+1}{0.1s+1} & 0 \\ 0 & \frac{5s+1}{0.1s+1} \end{pmatrix}$$

For the pre-filter V :

- Take a SVD of G_0 : $[V,S,U] = \text{svd}(G_0)$
- Define the filter as an additive uncertainty Δ over P , but with the same directionality that the plant P , i.e. $V = P + V * \Delta * U$, where Δ is $[0.2 \ 0; 0 \ 0.2]$

We select as pre-filter V :

$$V = P + \begin{pmatrix} -0.6246 & -0.7809 \\ -0.7809 & 0.6246 \end{pmatrix} \begin{pmatrix} 0.2 & 0 \\ 0 & 0.2 \end{pmatrix} \begin{pmatrix} -0.7066 & -0.7077 \\ 0.7077 & -0.7066 \end{pmatrix}$$

Note that P in the previous expression is not P_0 (static plant) but the plant itself (g_{223}). This is required on the Kwakernaak formulation because the representation on space state proposed assumes that the denominators of V and P are identical.

The results for the criteria S2 are shown in the following tables. In general the control is quite acceptable. The rise time is generally satisfied. The maximum is generally not satisfied but the exceedings are small. The cross coupling is generally fulfilled in the scenario 1 but not in the scenario 2. The control is worst for the plants with $k_1 <> k_2$ (g_{133} and g_{313}) as usual.

plant ch	set-point tracking			interaction	
Scenario 1					
plant	t=30	max	t=100	max	t=100
g223_u1	1.023	1.165	1.012	0.415	-0.003
g113_u1	0.922	1.159	1.045	0.361	-0.013
g133_u1	1.126	1.268	1.009	0.411	-0.016
g313_u1	0.876	1.095	1.025	0.901	0.008
g333_u1	1.098	1.174	1.001	0.496	-0.000
Scenario 2					
plant	t=30	max	t=100	max	t=100

g223_u2	1.014	1.040	1.002	0.731	-0.010
g113_u2	0.990	1.038	1.010	0.704	-0.036
g133_u2	0.883	1.412	1.013	1.039	-0.007
g313_u2	1.142	1.172	0.994	0.638	-0.020
g333_u2	1.031	1.085	1.000	0.861	-0.001
Scenario 3					
plant	t=30	max	t=100	max	t=100
g223_u3	0.399	0.575	0.399	0.720	0.600
g113_u3	0.406	0.549	0.396	0.691	0.601
g133_u3	0.390	0.587	0.399	0.737	0.601
g313_u3	0.410	0.563	0.398	0.703	0.599
g333_u3	0.393	0.678	0.400	0.849	0.600

Table 4-6: S2 criteria for the H-inf KWA controller

The temporal response for a set of 25 Monte Carlo plants for the scenario 1, 2 and 3 is shown in (Figure 4-30), (Figure 4-31) and (Figure 4-32).

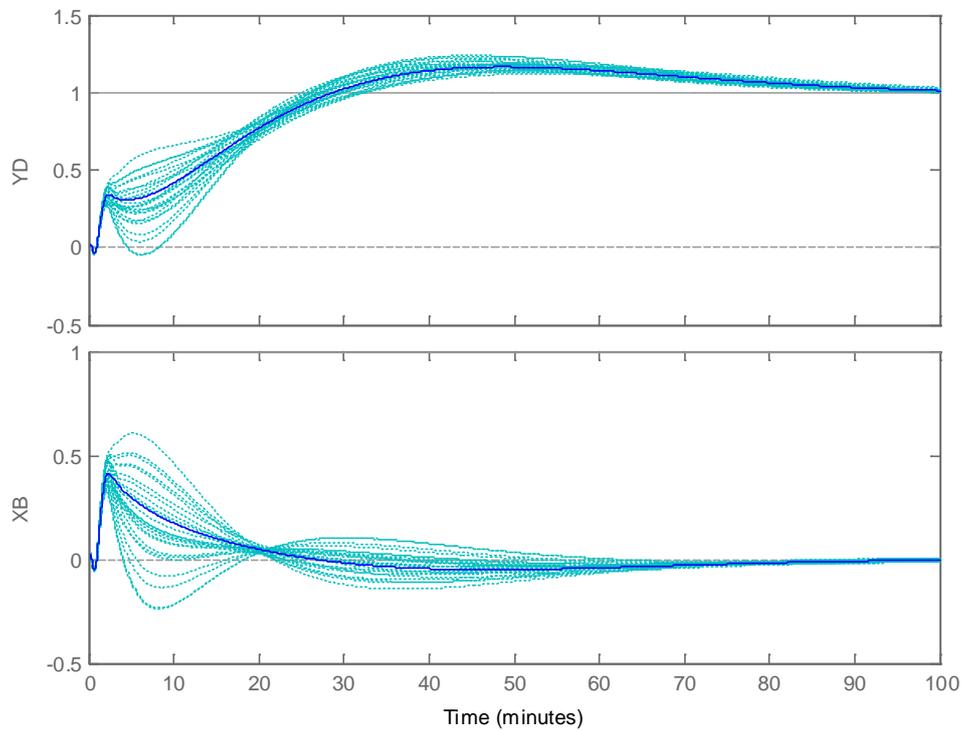


Figure 4-30: KWA Controller response to sce1

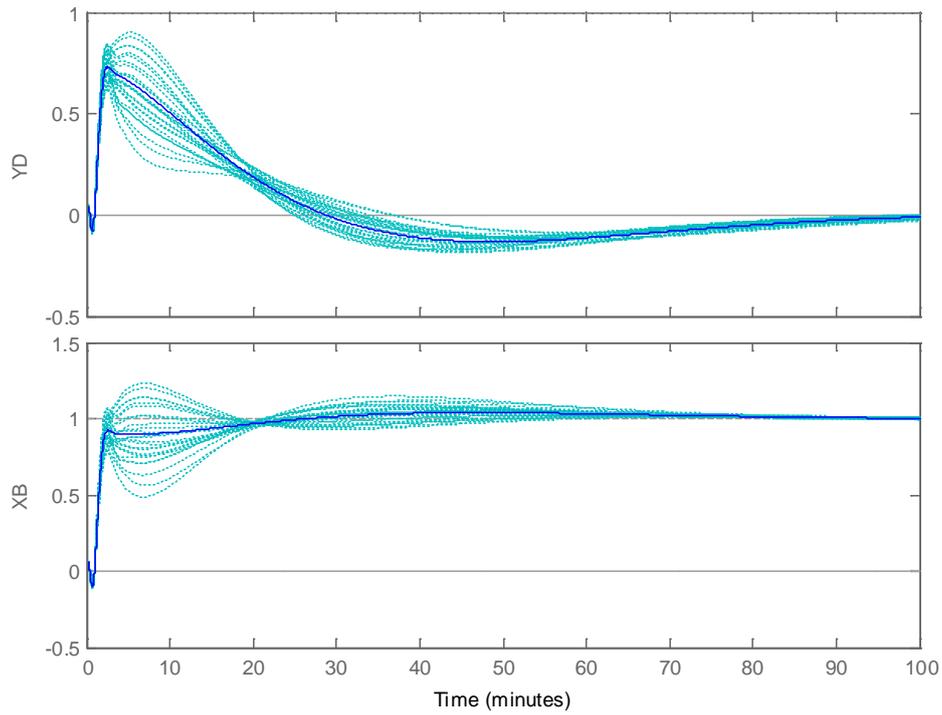


Figure 4-31: KWA Controller response to sce2

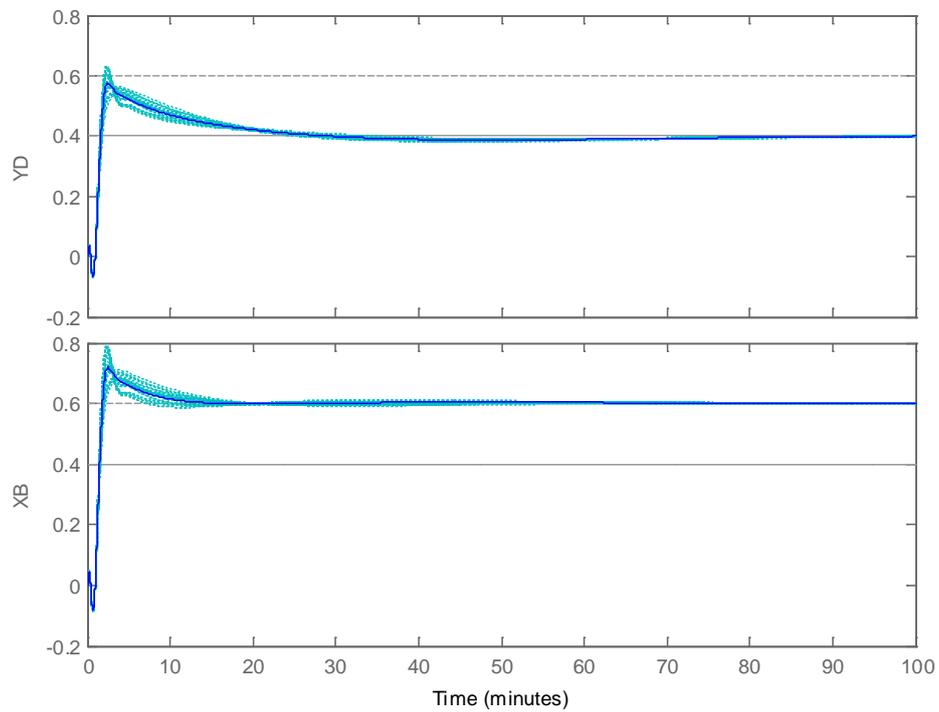


Figure 4-32: KWA Controller response to sce3

The (Figure 4-33) shows however that the criterion S3 is very slightly surpassed at 10 *rad/min* frequencies. The specification S4 ($KS_o < 1$ for $\omega > 105$ *rad/min*) is fulfilled ($KS_o = 0.37$).

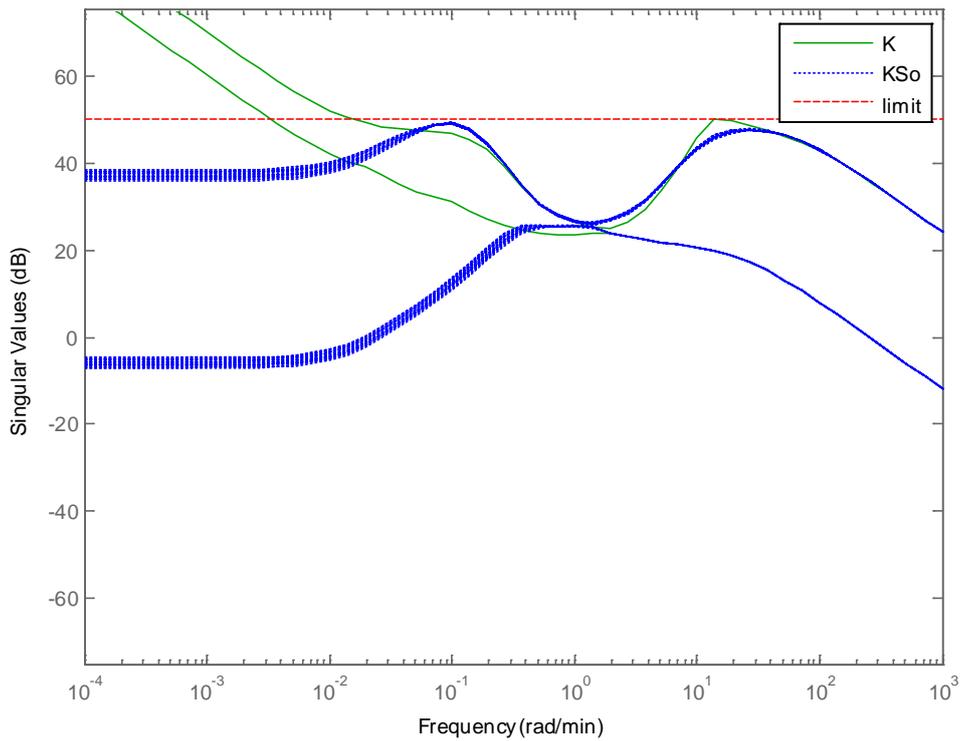


Figure 4-33 KWA controller K_{So}

The (Figure 4-34) shows the robustness on the control on the low variability of S_oP and T_o under disturbances on the actuator's gain.

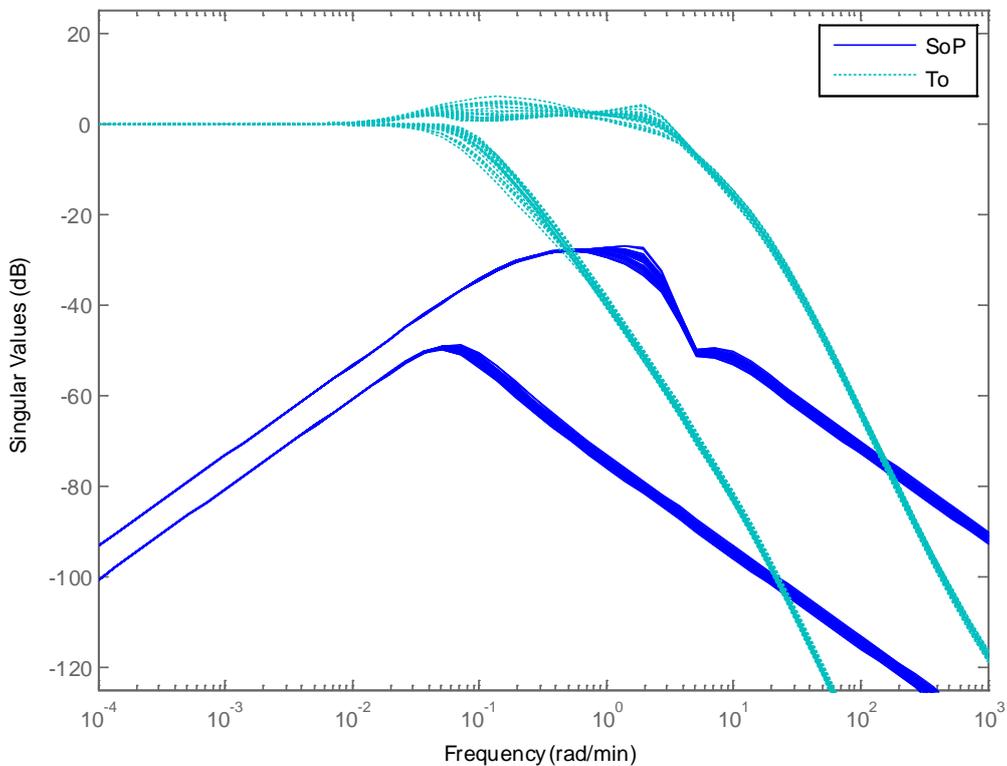


Figure 4-34: KWA S_oP and T_o Perturbed

The KWA controller has 12 states.

As summary, the optimization KWA improves a lot the results of the H-infinity Mixed Sensitivity controller.

4.4.5 H-Infinity Christen-Geering (CHGE) controller

We follow the guidelines proposed before for the selection of weight for S_oP .

For the S_o part we select ideal second order systems with:

- $\omega_n = 10/75$; delta = 0.9; (channel 1)
- $\omega_n = 5/75$; delta = 0.9; (channel 2)
- (i.e. we select ideal systems 10 and 5 times faster than the open loop)

For the P_o part of the weight, we attempt the two variants proposed previously:

Variant 1: proposed by CHGE, take P_o as the steady gain of scaled P nominal (g_{22x}) i.e.

$$P_o = \begin{bmatrix} 100 & 0 \\ 0 & 126 \end{bmatrix} \begin{bmatrix} 0.8780 & -0.8640 \\ 1.0820 & -1.0960 \end{bmatrix}$$

Variant 2: proposed by us, take the weight for P as the inverse of a multiplicative perturbation (Δ) over P_o , for example:

$$W_{P_part} = inv(P_o * (1 + \Delta))$$

And the following method for deciding the Δ_i values:

- Build a set of uncertainties $P_o * (1 + \Delta)$, where $\Delta = [\Delta_1 \ 0; 0 \ \Delta_2]$ and $\Delta_i \in [-0.2, 0.2]$
- Plot the $svd()$ of these uncertainties
- Find the one with maximum singular values and gets its Δ_1, Δ_2

The selected values are $\Delta = [0.1666 \ 0; 0 \ 0.1374]$. Note that these values differs from the ‘a priori’ supposition of $\Delta = [0.2 \ 0; 0 \ 0.2]$ as optimal values.

For the weight W_u for T_i : We take the same ideal second order systems that for S_o :

- $\omega_n = 15/75$; delta = 0.9; (channel 1)
- $\omega_n = 10/75$; delta = 0.9; (channel 2)
- a second order zero ($0.0001 \ s + 1$) has been added to the previous ideal systems to make the weight proper

As previously said, in the standard H-infinity Mixed Sensitivity problem S_oW and T_oW are of the same magnitude. In the CHGE variant, S_oP and T_i are of different magnitude in a factor equal to the ‘static gain of P ’. We have to introduce this factor in the weight for T_i in order to achieve good results of the minimization algorithm.

A gain equal to the maximum singular value of the nominal plant g_{223} has been introduced in the W_u weight.

Summarizing, we take the following weights (expressed in rad/min).

For the variant 1, W_p :

$$W_p = \text{inv}(P_o) \text{inv} \left(\begin{array}{cc} 1.00001 - \frac{0.1333^2}{s^2 + 0.24s + 0.1333^2} & 0 \\ 0 & 1.00001 - \frac{0.0667^2}{s^2 + 0.12s + 0.0667^2} \end{array} \right)$$

For the variant 2, W_p :

$$W_p = \text{inv} \left(P_o \left(1 + \begin{bmatrix} 0.1666 & 0 \\ 0 & 0.1374 \end{bmatrix} \right) \right) \text{inv} \left(\begin{array}{cc} 1.00001 - \frac{0.1333^2}{s^2 + 0.24s + 0.1333^2} & 0 \\ 0 & 1.00001 - \frac{0.0667^2}{s^2 + 0.12s + 0.0667^2} \end{array} \right)$$

For both variants, W_u :

$$W_u = \text{inv} \left(299.84 \begin{array}{cc} \frac{0.2^2 (1e^{-8}s + 0.002s + 1)}{s^2 + 0.36s + 0.02^2} & 0 \\ 0 & \frac{0.1333^2 (1e^{-8}s + 0.002s + 1)}{s^2 + 0.24s + 0.1333^2} \end{array} \right)$$

The results for the criteria S2 are shown in the following tables. The results of both variants are shown. In general the behaviors of both variant are very similar, being variant 2 a bit better (less number of violations). The worst case is g_{313} with a cross-coupling of 1.317 and 1.289, much greater of the maximum allowed 0.5.

The stationary error of the tracking at $t = 100$ min is also out of spec in all cases, but the error is not too big (around 0.05).

Variant 1 (CHGE pure): inv(P0)

plant ch	set-point tracking		interaction		
Scenario 1					
plant	t=30	max	t=100	max	t=100
g223_u1	0.910	1.040	1.037	0.738	-0.100
g113_u1	0.879	1.033	1.033	0.752	-0.071
g133_u1	0.532	1.233	1.116	0.634	-0.011
g313_u1	1.176	1.269	0.973	1.317	-0.145

Variant 2: inv(P0 (1 + Δ))

plant ch	set-point tracking		interaction		
Scenario 1					
plant	t=30	max	t=100	max	t=100
g223_u1	0.893	1.035	1.035	0.743	-0.087
g113_u1	0.865	1.026	1.026	0.758	-0.045
g133_u1	0.521	1.195	1.119	0.636	-0.005
g313_u1	1.155	1.233	0.973	1.289	-0.113

g333_u1	0.942	1.049	1.031	0.856	-0.099
Scenario 2					
plant	t=30	max	t=100	max	t=100
g223_u2	0.640	1.081	1.081	0.406	-0.030
g113_u2	0.585	1.057	1.057	0.364	-0.027
g133_u2	1.075	1.163	1.009	0.729	-0.094
g313_u2	0.296	1.117	1.117	0.433	0.022
g333_u2	0.695	1.088	1.080	0.480	-0.025
Scenario 3					
plant	t=30	max	t=100	max	t=100
g223_u3	0.420	0.478	0.397	0.608	0.608
g113_u3	0.428	0.462	0.397	0.606	0.606
g133_u3	0.456	0.498	0.390	0.641	0.601
g313_u3	0.395	0.522	0.402	0.647	0.612
g333_u3	0.415	0.564	0.397	0.700	0.608

g333_u1	0.923	1.042	1.033	0.856	-0.097
Scenario 2					
plant	t=30	max	t=100	max	t=100
g223_u2	0.615	1.070	1.070	0.406	-0.028
g113_u2	0.564	1.036	1.036	0.364	-0.021
g133_u2	1.039	1.112	1.003	0.703	-0.096
g313_u2	0.280	1.092	1.092	0.432	0.022
g333_u2	0.665	1.080	1.078	0.479	-0.027
Scenario 3					
plant	t=30	max	t=100	max	t=100
g223_u3	0.422	0.479	0.397	0.607	0.607
g113_u3	0.429	0.463	0.398	0.604	0.604
g133_u3	0.457	0.496	0.390	0.636	0.600
g313_u3	0.397	0.522	0.402	0.647	0.610
g333_u3	0.417	0.564	0.397	0.699	0.608

Table 4-7: S2 criteria for CHGE controllers

The responses to a Montecarlo simulation (25 plants) with the CHGE variant 2 (variant 1 is very similar) are shown in (Figure 4-35), (Figure 4-36) and (Figure 4-37). It can be appreciated a fulfillment of raise time for most of the cases, but also a cross-coupling greater than 0.5 in a lot of cases.

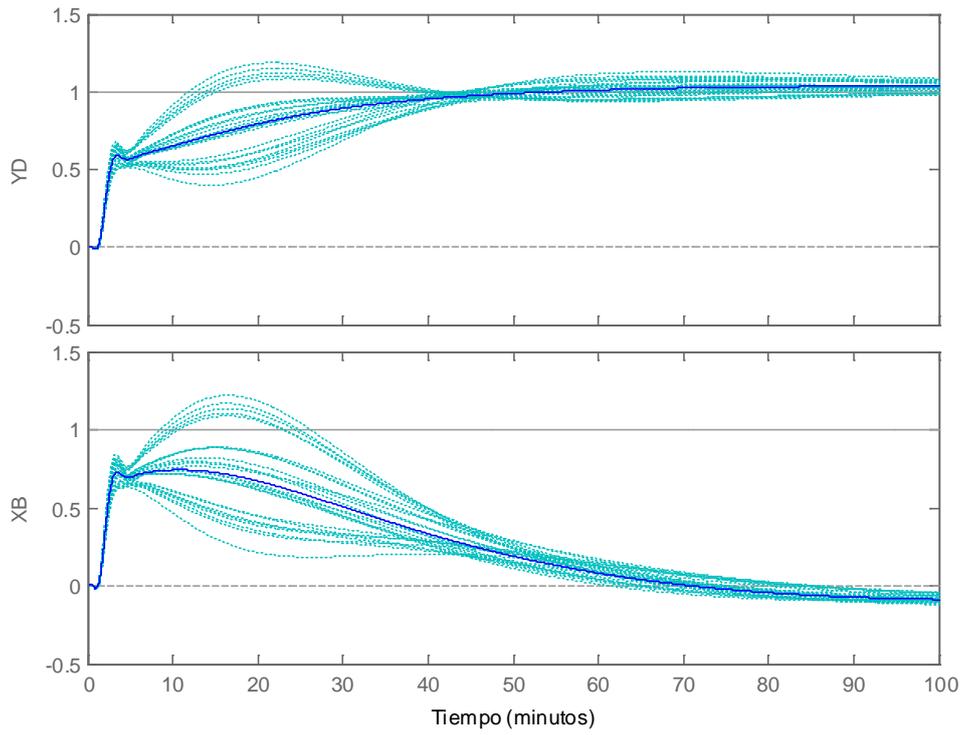


Figure 4-35: CHGE controller response to sce1

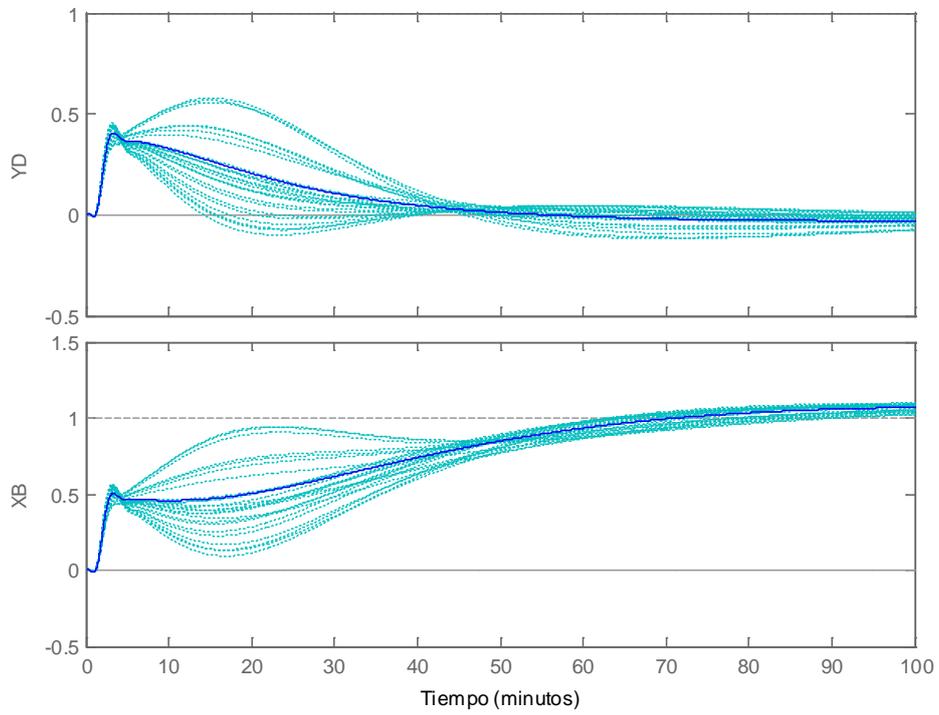


Figure 4-36: CHGE controller response to sce2

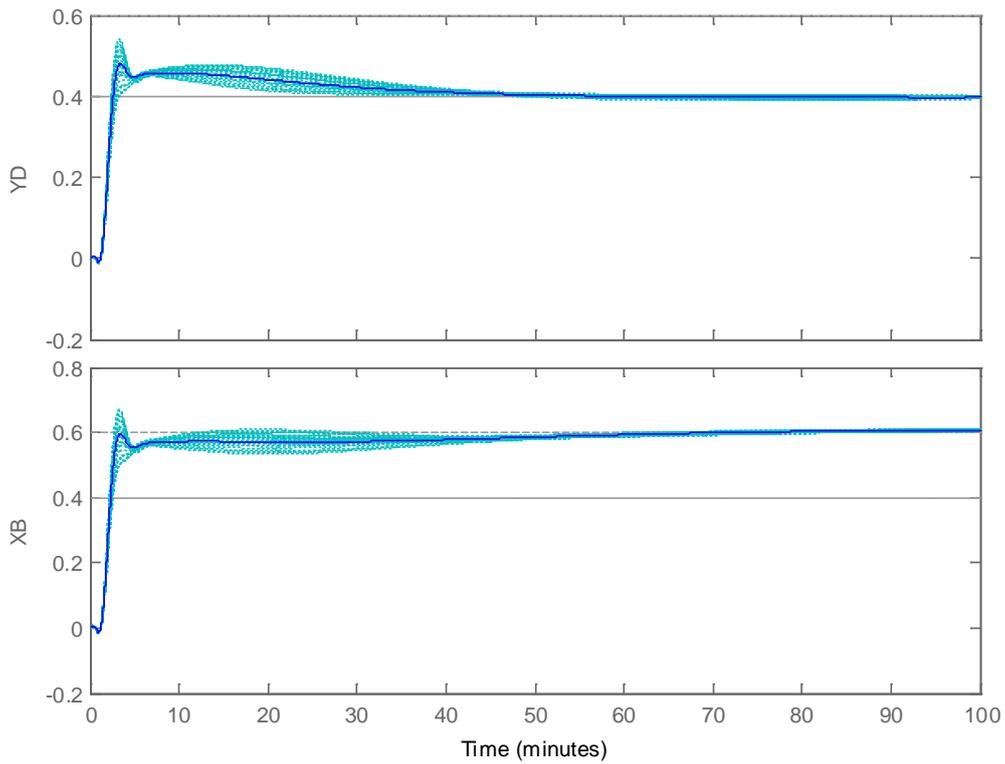


Figure 4-37: CHGE controller response to sce3

The specification S3 ($KS_o < 50 \text{ dB}$) is fulfilled. The specification S4 ($KS_o < 1$ for $\omega > 105 \text{ rad/min}$) is also fulfilled ($KS_o = 0.016$) (Figure 4-38).

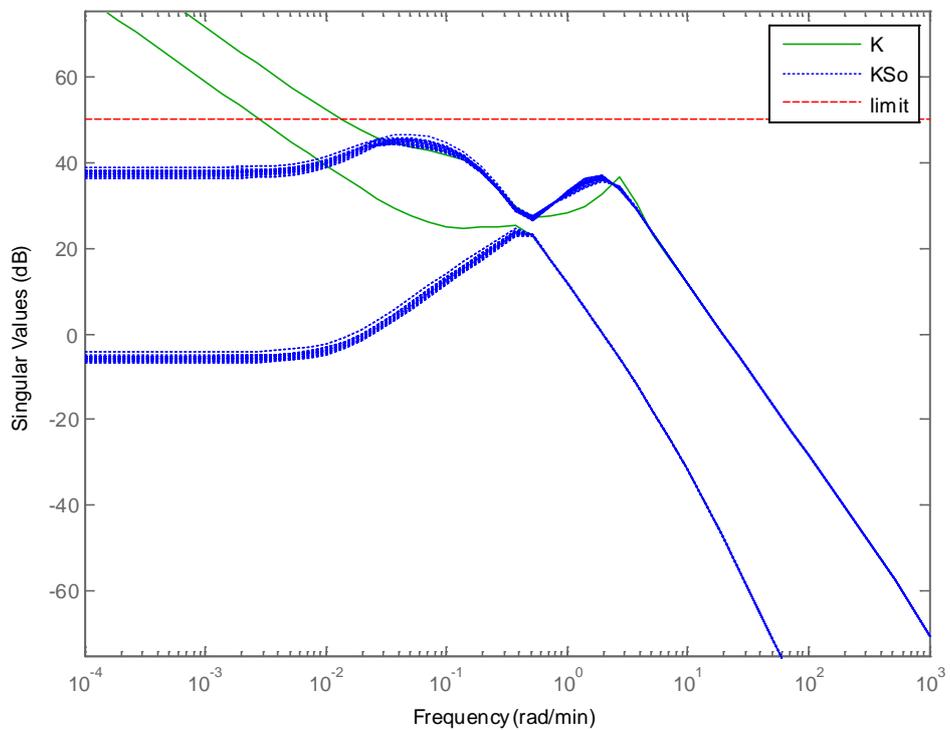


Figure 4-38: CHGE controller KS_o

The diagrams of S_oP and T_o for the set of 25 Monte Carlo plants show the robustness of the design (Figure 4-39).

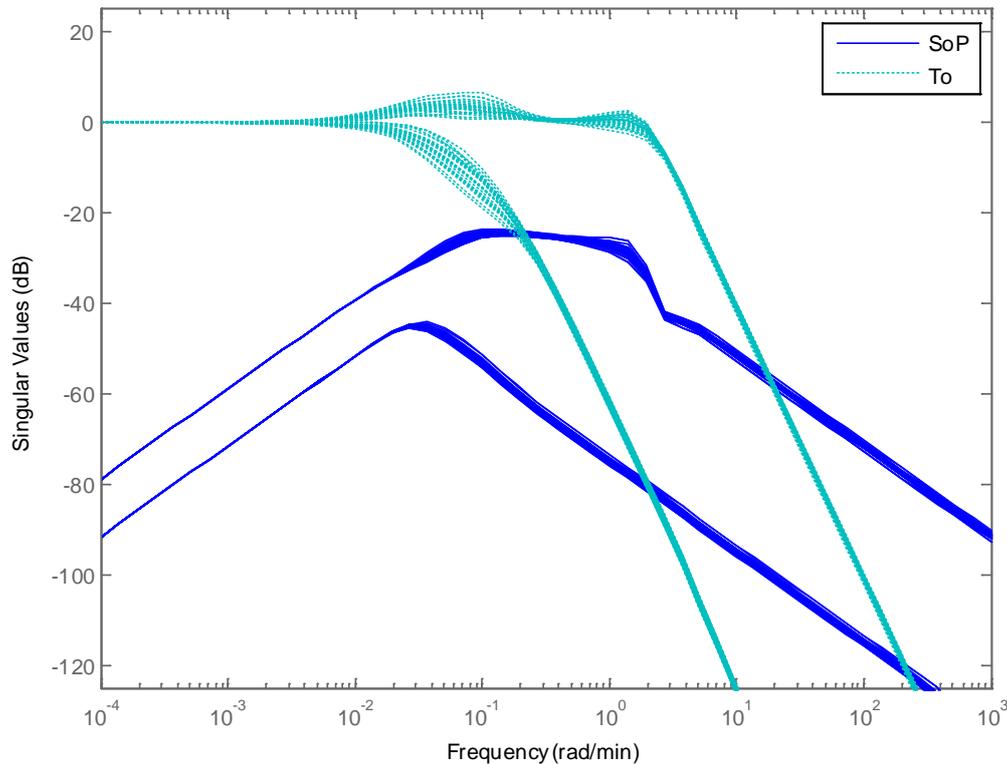


Figure 4-39: CHGE S_oP and T_o

The following points are noted:

- The *CHGE* variant made the control robust to the uncertainties in the plant actuators.
- The controller generated has 14 states (less than half that the μ controllers that has 32-33 states).
- The improved robustness of the *CHGE* method has as prize the reduced performances of the nominal plant g_{22x} or plants with $k_1 = k_2$ (i.e. plants g_{11x} , g_{33x}).
- The performances for g_{313} are still not satisfactory, in particular for the cross-coupling
- Efforts to improve the performances of g_{313} have been not satisfactory:
 - o Design based on the P for g_{313} (instead of g_{223}) produces results inside specs for this plant but deterioration of the performances of all the other plants
 - o It has been also attempted to introduce a bigger variation of Δ (for example instead a 20% a 30%). In particular, allowing a bigger delta value for the second channel decreased the cross-coupling on g_{313} below 1, but this is at cost of decreasing the rise time for all the plants

4.4.6 H-Infinity Structured (Apkarian) controller

We propose in this chapter an H-Infinity Structured controller elaborated from the example provided in the (Gahinet, 2011), ‘Section V, Distillation Column example. (Note that the example is also available on Apkarian website as ‘Decoupling Controller for a Distillation Column’).

We introduce significant modifications in our work:

- Note that the plant proposed in (Gahinet, 2011) is the “normalized” plant of Skogestad, not the original of the CDC benchmark. In order to be fair in the comparison with other controllers we use also here the original plant of the CDC benchmark.
- Note that no time delay is taken into account in the original article. We re-introduce the time delay as required in the CDC benchmark.: the plant with delay is more difficult to control.
- Remove of the weight on the noise (as noise not part of the CDC benchmark)
- Introduction on a weight in the actuator control signal

The H-Infinity structured controller allows imposing a structure for the controller. The structure chosen is a decoupler filter in series with a diagonal PI control, in each input channel.

Then, the following weighting scheme is designed (Figure 4-40):

- A weight W_s , weighting S
- A weight W_d , weighting the disturbance at plant input
- A weight W_u , weighting the control signal has been introduced
- The weight W_n , related to noise has been removed

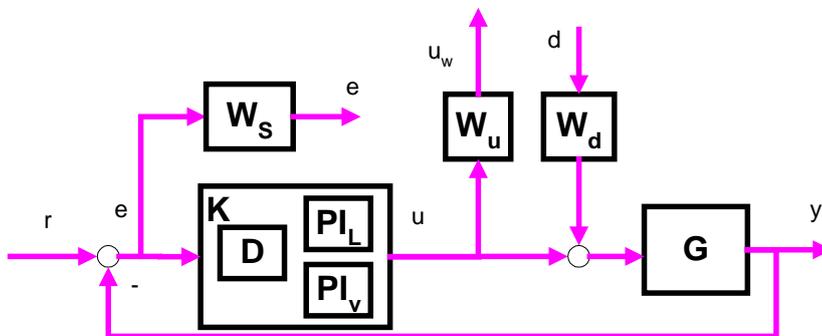


Figure 4-40: Block diagram for the H-infinity Structured controller

The augmented plant is created with the standard Matlab function ‘connect’. The equations are created as follows.

```
% Label block I/Os: Inputs and outputs of the plant
G.InputName = {'L', 'V'};
G.OutputName = {'yD', 'yB'};

% Input to the decoupler is the error signal
DM.u = 'e'; DM.y = {'yDL', 'yDV'};
```

```

% Input to each PID are the decoupler
PI_L.u = 'yDL'; PI_L.y = 'yPL';
PI_V.u = 'yDV'; PI_V.y = 'yPV';

% Error is reference minus output of the plant. Call ref components rD, rB
Sum1 = sumblk('e = %r - %y', {'rD','rB'}, G.y);

% Input to plant G is sum of disturbance and controller output (PID output)
Sum2 = sumblk('%up = %uc + d', G.u, [PI_L.y ; PI_V.y]); % disturbances

CL0 = connect(G,DM,PI_L,PI_V,Sum1,Sum2,
{'rD','rB','d'},{'e','yD','yB','yPL','yPV'});

% We have to build the diagonal as:
% output_weights * plant * inputs_weights
% Output weight is Ws, identity weight for y (eye(2)) and Wu
% Input weight is identity weight for references (eye(2)) and Wd
CL0 = blkdiag(Ws, eye(2), Wu) * CL0 * blkdiag(eye(2), Wd*eye(2));

```

Note how the H-Infinity structured toolbox functions works:

First, the augment plant is defined as:

- Having as inputs the exogenous signals {'rD', 'rB', 'd'} (i.e. the reference and the disturbance)
- Having as outputs the weighted signals {'e', 'yD', 'yB', 'yPL', 'yPV'} (i.e. the error, the measured outputs and the control signals)

Then the diagonal augmented plan with weights is formed as shown in (Figure 4-41):

$$OutputWeights * P_{au} * InputsWeights$$

- Inputs weights are a diagonal transfer matrix composed of an identity matrix for the references (that are not weighted) and W_d .
- Output weights are a diagonal transfer matrix, composed of W_p , an identity matrix for the measured signals (that are not weighted) and W_u .

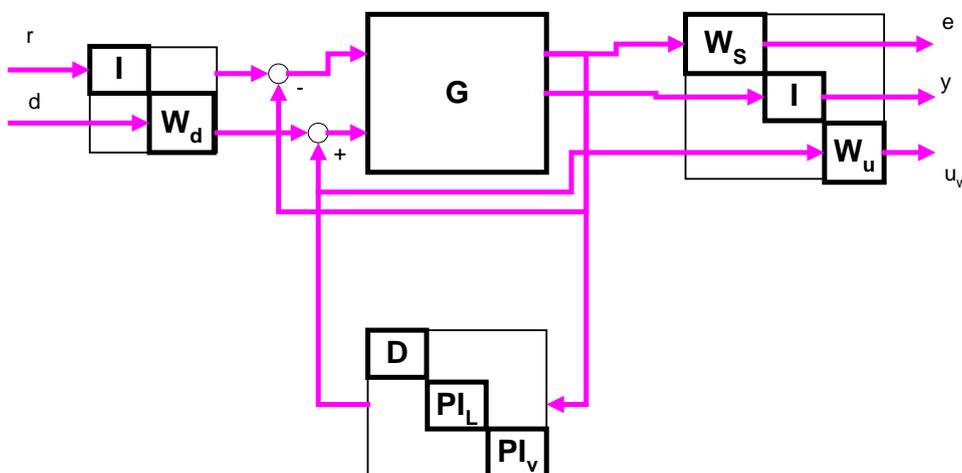


Figure 4-41: The Structured controller in diagonal view

This is equivalent of weighting a diagonal transfer matrix with diagonal elements:

- $W_s P$ (TF r to e)
- $S_o P W_d$ (TF d to y)
- $W_u K S_o$ (TF r to u_w)

Pre-selection of gains in decoupler and PIDs

The decoupler proposed in the original article is free. After one iteration of design we have noted that is better to leave free only the cross-coupled terms (1,2) and (2,1) and instead fix the terms (1,1) and (2,2) to (0.5, -0.5). The justification is that we want to have the overall gain provided by the decoupler near to 1. The sign is justified by the sign of the stationary plant.

Equivalently, we fix the gain term of the PIDs as in the original article. Instead to fix to 1 we fix to 0.8 for the first channel and 1 for the second channel. This mimics the original gains of the open loop plant (second channel has 25% more gain).

Choice of Weight W_s

We follow the strategy used previously, an ‘ideal 2nd order system’ instead of the original weight proposed by Apkarian.

- Choose a 2nd order system that meets the specifications (chose ω_n and delta)
- $T_{o_id} = \omega_n^2 / (s^2 + 2 \text{ delta } \omega_n s + \omega_n^2)$
- Calculate $S_{o_id} = 1 - T_{o_id}$
- Define the weight as the inverse of S_{o_id}

By the physics of the plant we know that the plant has low gain when both inputs increase or decrease simultaneously. So we introduce a small asymmetry in the weights making the inputs u_1 slightly faster than the inputs for u_2 (because the second channel of the open loop plant as a 20% more gain).

We take $\omega_n = 10/75$ for the first channel (i.e. 10 times faster than the open loop response) and $\omega_n = 8/75$ for the second channel. For both channel delta = 1.0.

$$W_s = \text{inv} \left(\begin{array}{cc} 1.0001 - \frac{0.1333^2}{s^2 + 0.2667s + 0.1333^2} & 0 \\ 0 & 1.0001 - \frac{0.1067^2}{s^2 + 0.2133s + 0.1067^2} \end{array} \right)$$

Choice of Weight W_d

The actuators have a 20% uncertainty. This is a multiplicative uncertainty $1*(1+0.2)$ that in closed loop evaluates as 1.2 (or 0.8 if uncertainty is -20%). We change static weight proposed in the original article as:

$$W_d = \begin{pmatrix} 1.2 & 0 \\ 0 & 1.2 \end{pmatrix}$$

Choice of Weight W_u

We want to penalize the actuators effort at high frequencies. We chose the standard form of a weight.

$$W_u = \frac{\frac{1}{M} s + \omega_b}{s + \omega_b A}$$

We choose bandwidth $\omega = 0.06$ rad/min, $A = 316$ (= 50 dB, max gain allowed for the actuators). Again we know by the physics of the plant that the plant has low gain when both inputs increase or decrease simultaneously. So we introduce a small asymmetry in the parameter M limiting the actuator effort (M = 0.75 for first channel and M = 0.25 for second channel).

The weights are:

$$W_u = \begin{pmatrix} \frac{\frac{1}{M} s + \omega_b}{s + \omega_b A} & 0 \\ 0 & \frac{\frac{1}{M} s + \omega_b}{s + \omega_b A} \end{pmatrix} = \begin{pmatrix} \frac{\frac{1}{0.75} s + 0.06}{s + 0.06 * 0.316} & 0 \\ 0 & \frac{\frac{1}{0.25} s + 0.06}{s + 0.06 * 0.316} \end{pmatrix}$$

The results for the criteria S2 are shown in the following tables. In general the control is quite acceptable. The rise time however is generally not satisfied for scenarios 1 and 2. The maximum is generally not satisfied in scenario 1 but the exceedings are small. The cross coupling has 10% exceedings in the scenario 1 but greater in the scenario 2. The control is worst for the plants with $k_1 \ll k_2$ (g_{133} and g_{313}) as usual. Scenario 3 is fulfilled in almost all indicators.

plant ch	set-point tracking		interaction		
Scenario 1					
plant	t=30	max	t=100	max	t=100
g223_u1	0.802	1.113	1.113	0.533	-0.007
g113_u1	0.766	1.096	1.096	0.555	0.072
g133_u1	0.579	1.103	1.103	0.288	0.046
g313_u1	1.046	1.138	1.077	0.907	-0.077
g333_u1	0.835	1.122	1.117	0.582	-0.058
Scenario 2					
plant ch	set-point tracking		interaction		
g223_u2	0.616	1.006	1.006	0.594	-0.089

g113_u2	0.586	0.944	0.944	0.535	-0.075
g133_u2	0.783	1.116	0.964	0.863	-0.080
g313_u2	0.412	1.062	1.062	0.292	-0.060
g333_u2	0.646	1.047	1.047	0.659	-0.092
Scenario 3					
plant	t=30	max	t=100	max	t=100
g223_u3	0.421	0.537	0.392	0.682	0.601
g113_u3	0.425	0.490	0.393	0.622	0.595
g133_u3	0.436	0.598	0.393	0.758	0.597
g313_u3	0.404	0.477	0.395	0.606	0.606
g333_u3	0.417	0.592	0.392	0.751	0.605

Table 4-8: S2 criteria for H-inf Structured controller

The temporal response for a set of 25 Monte Carlo plants for the scenario 1, 2 and 3 is shown in (Figure 4-42), (Figure 4-43) and (Figure 4-44).

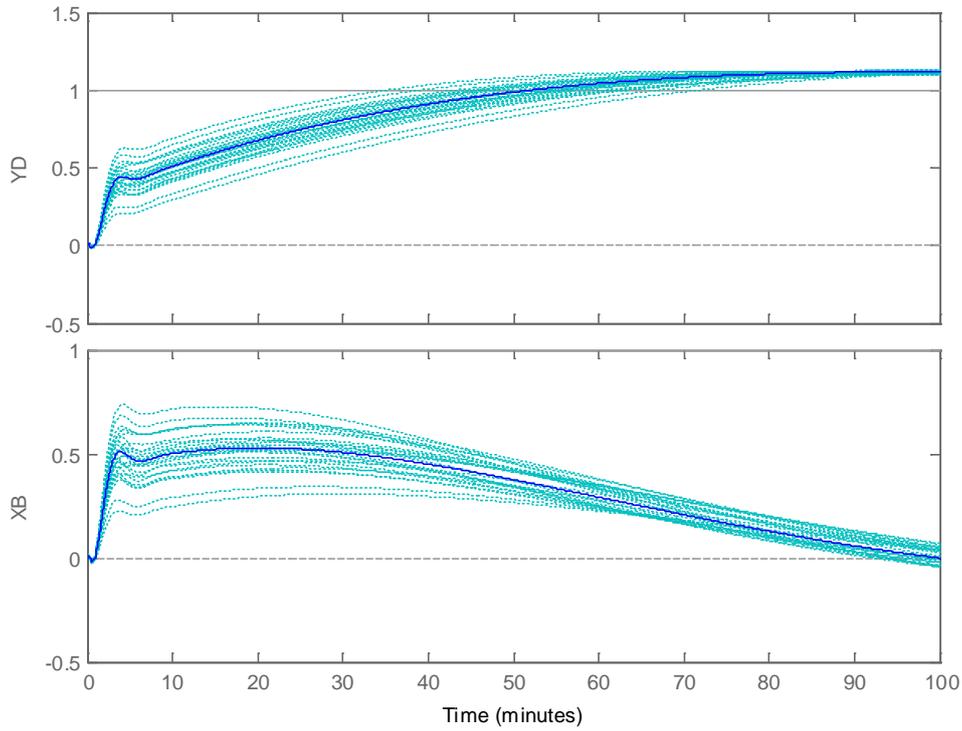


Figure 4-42: H-inf structured controller response to sce1

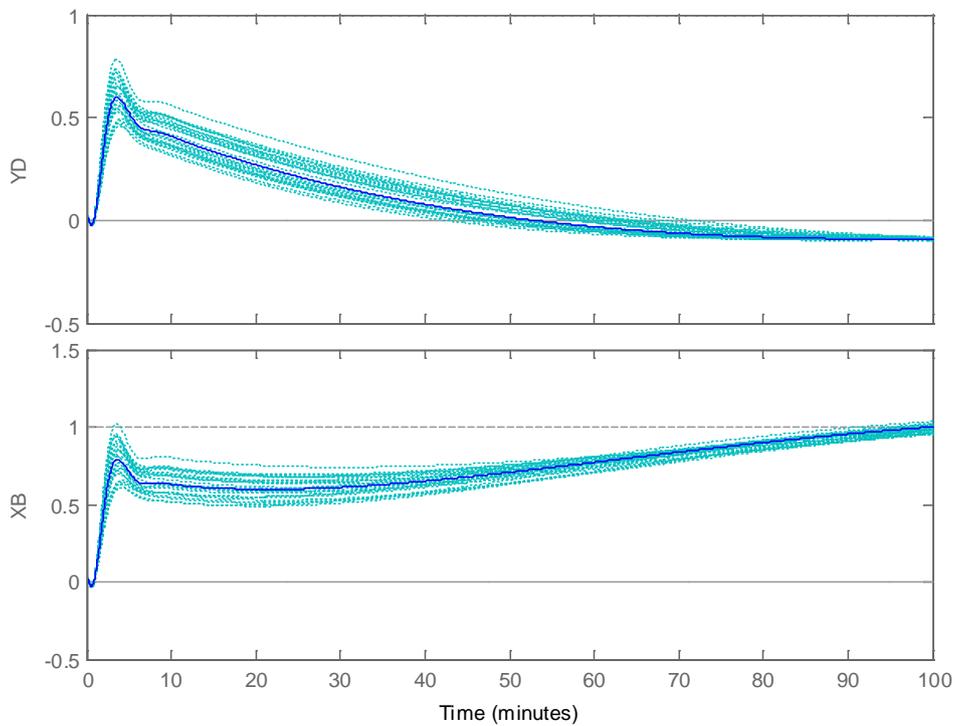


Figure 4-43: H-inf structured controller response to sce2

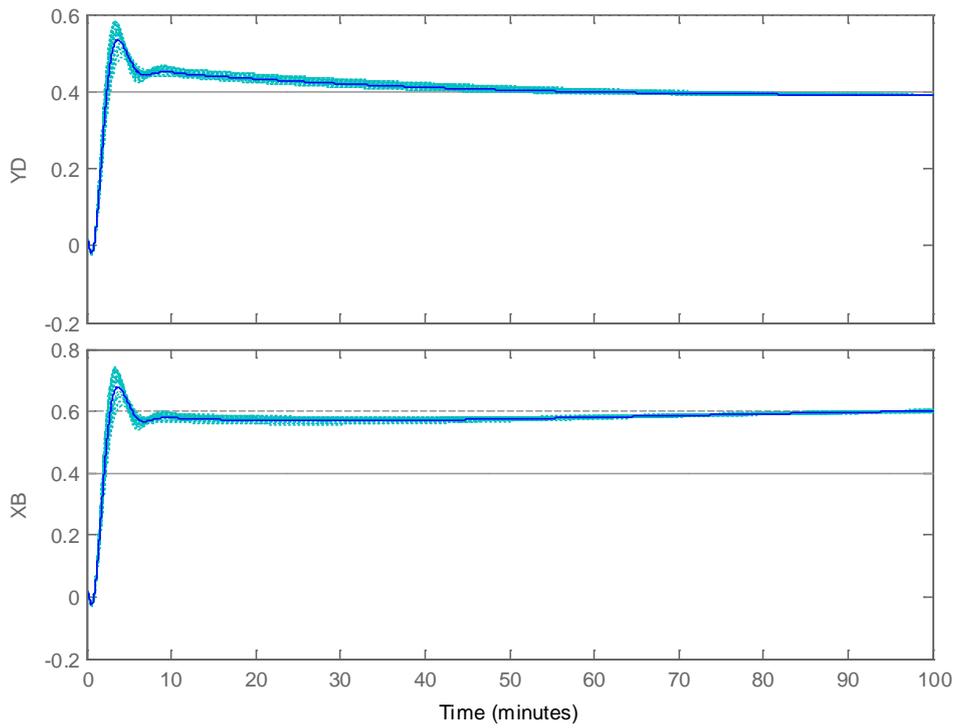


Figure 4-44: H-inf structured controller response to sce3

The singular values diagram of S_oP and T_o demonstrate the robustness of the plant (Figure 4-45). The maximum singular value of S_oP is always well below the maximum singular value of T_o .

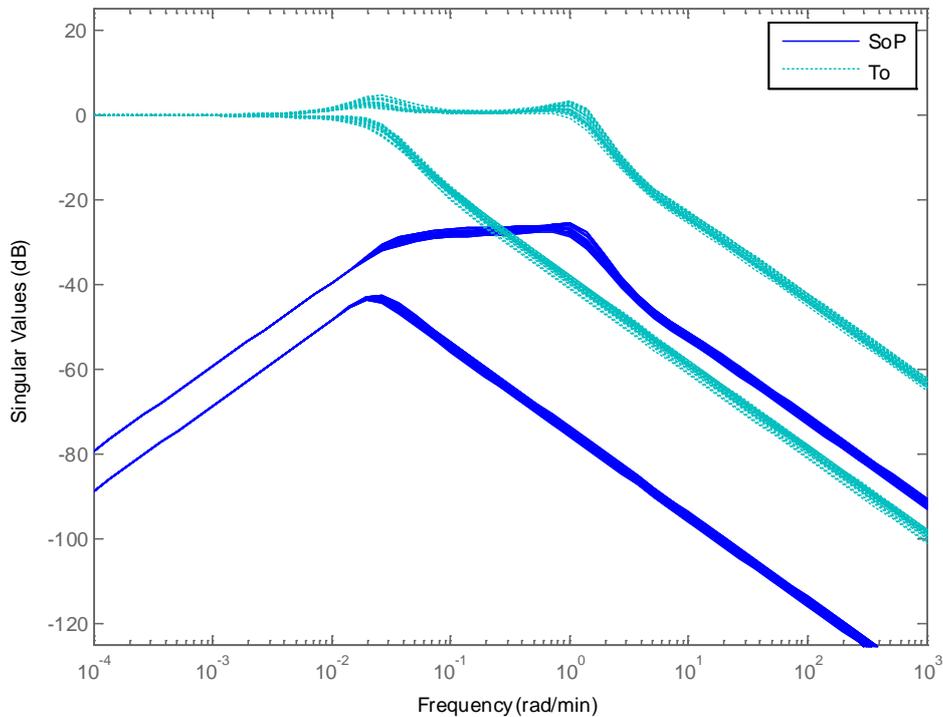


Figure 4-45: H-inf structured controller S_oP and T_o

The specification S3 ($KS_o < 50 \text{ dB}$) is fulfilled. The specification S4 ($KS_o < 1$ for $\omega > 105 \text{ rad/min}$) is not fulfilled (Figure 4-46). Note that K does not roll-off at higher frequencies.

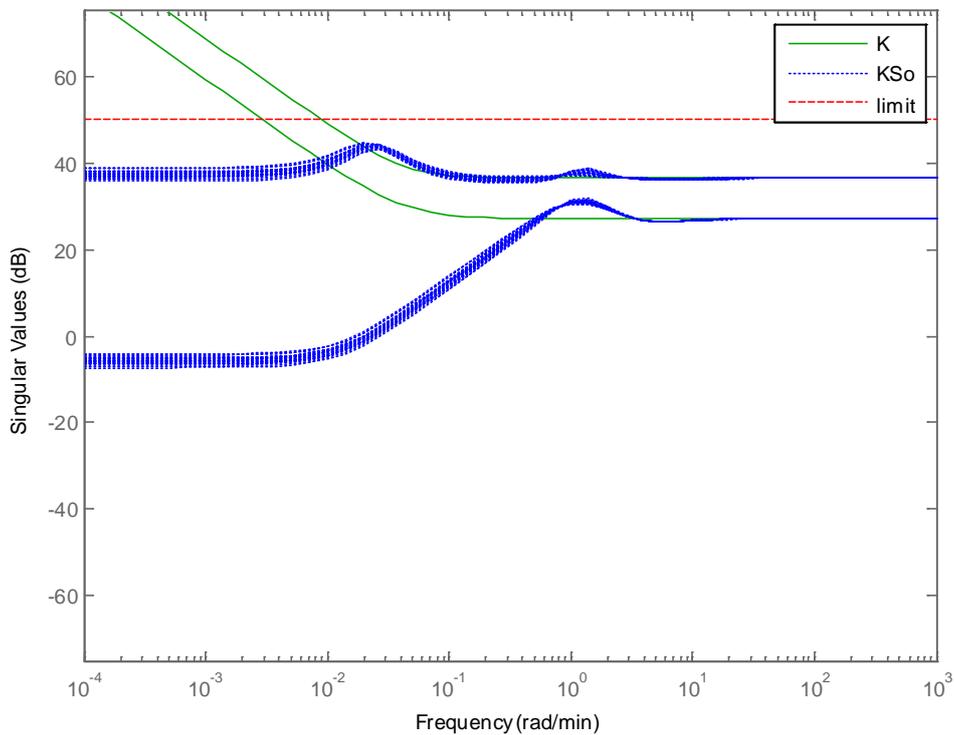


Figure 4-46: H-inf structured controller K and KS_o

Now we can put in place the flexibility of the H-infinity structured approach. As we know the structure of the generated controller we can simply add a roll-off filter for making KS_o to roll-off (Figure 4-47). (This could be introduced also as a weight on T_o , but we prefer to introduce it as a design imposed filter demonstrating the ability of tuning the design).

Choosing a roll off filter:

$$W_{roll} = \begin{bmatrix} \frac{60}{s+60} & 0 \\ 0 & \frac{60}{s+60} \end{bmatrix}$$

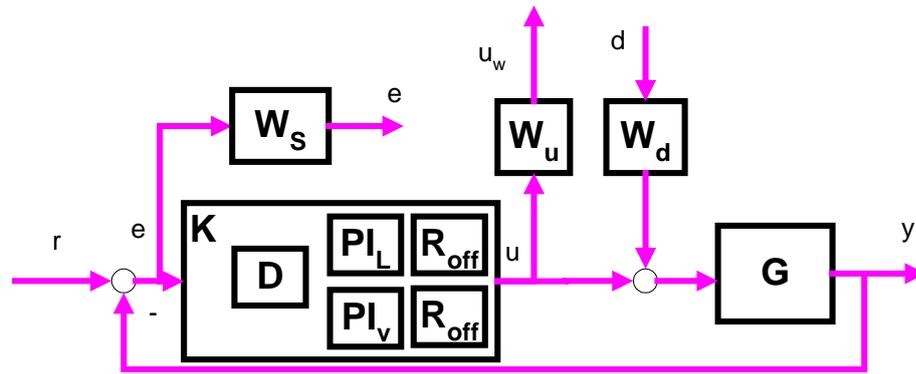


Figure 4-47: Block diagram for the H-infinity Structured controller with roll-off term

The specification S4 ($KS_o = 0.61 < 1$ for $\omega > 105 \text{ rad/min}$) is still not fulfilled ($KS_o = 24$) however we have a control that roll-overs at high frequency (Figure 4-48) (i.e. physically implementable) with no penalty on the performances (Figure 4-49) and (Figure 4-50). The controller generated has only 4 states.

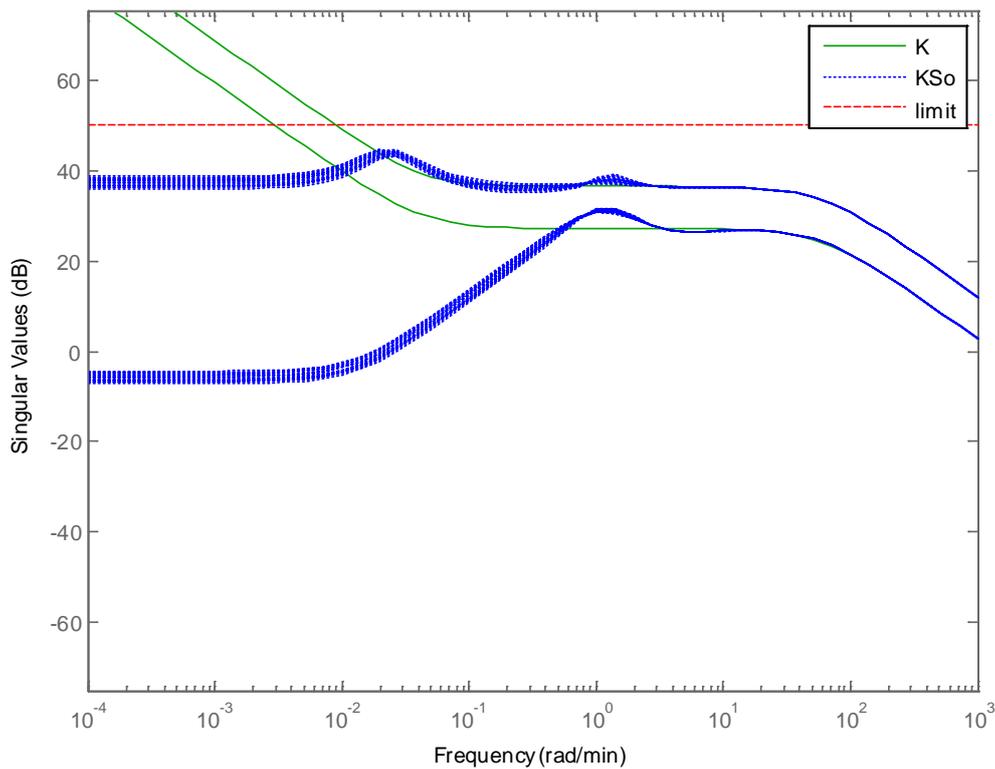


Figure 4-48: H-inf structured controller K and KS_o with roll-off filter

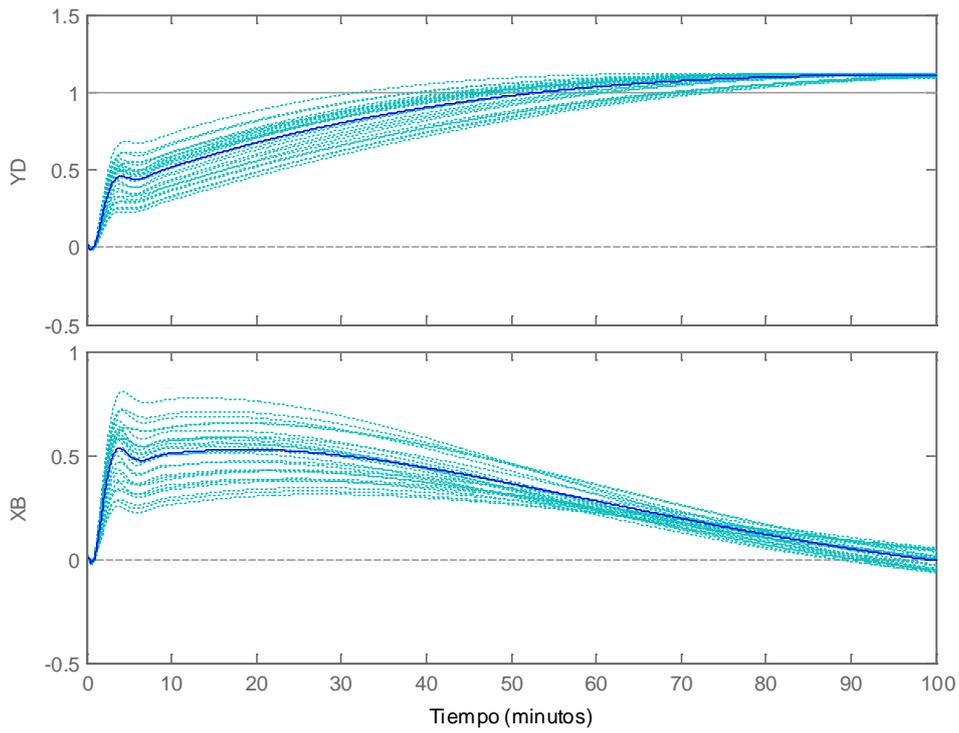


Figure 4-49: H-inf structured controller with roll-off response to sce 1

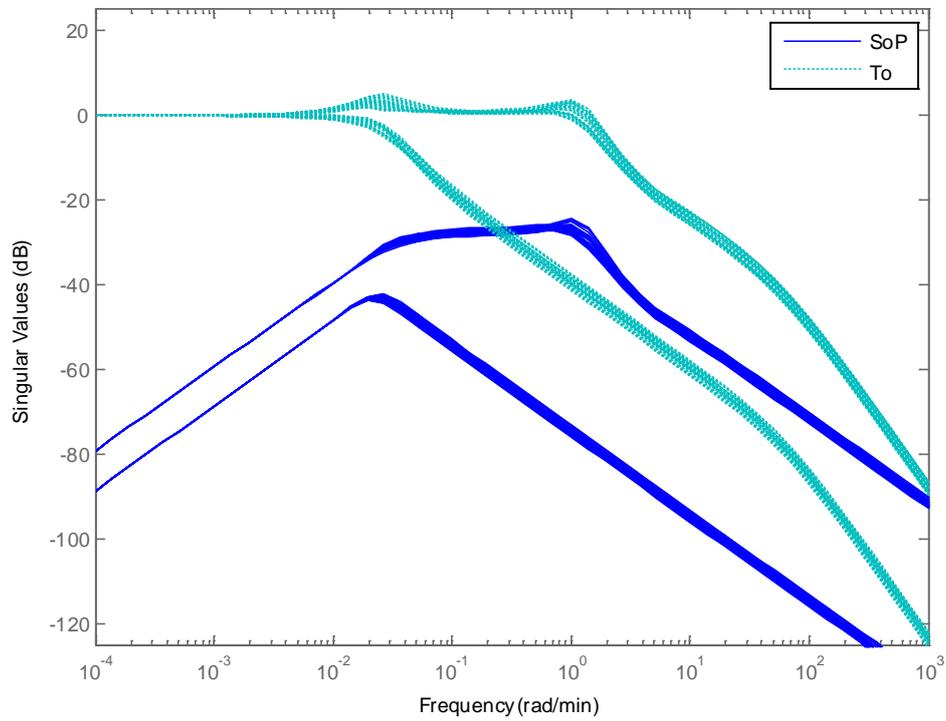


Figure 4-50: H-inf structured controller with roll-off S_oP and T_o

4.5 Conclusions of applying H-Infinity to the CDC benchmark

In this chapter several H-infinity controllers have been compared for the CDC benchmark. For reference these new controllers have been compared with several μ controllers (Skogestad, 1996), (Lundstrom, 1999) and (Balas, 2012). These μ controllers represent well the evolution of μ synthesis along the last 15 years.

The conclusion of (Skogestad, 1996) stating that the H-Infinity method is not appropriate for ill conditioned plants is correct only when referring to the H-infinity Mixed Sensitivity variant. Other variants of H-infinity provide satisfactory results with ill conditioned plants.

The variant KWA (Kwakernaak, 1993) was the first optimization proposed to the H-infinity Mixed Sensitivity for dealing with ill-conditioned plants. It has been demonstrated that this method as first optimized in (Cao, 1997) and further optimized in this thesis (including a pre-filter obtained as a multiplicative perturbation over the open loop plant P) provides a satisfactory control for the benchmark problem. The performances of this KWA controller are better than the ones of the older μ controller and comparable with the most modern ones of (Lundstrom, 1999) and (Balas, 2012).

It has been also demonstrated that the variant of the H-infinity CHGE proposed by Christen Geering in (Christen, 1997) can be also used for control of ill-conditioned plants, achieving results comparable to KWA and the modern μ controllers.

Finally it has been demonstrated that the variant of H-infinity Structured proposed by Apkarian and Gahinet (Gahinet, 2011) as improved in this thesis can be used for control of ill-conditioned plants, achieving results comparable to KWA and the modern μ controllers.

It can be concluded that H-Infinity is a valid method for the control of ill-conditioned plants, providing that one of the new optimized variants is adopted.

The following table summarizes the controller performances and complexity:

Controller	S1 (Stable)	S2 (performance)	S3 (KS ₀)	S4 (KS ₀ bis)	Num States	Tuning effort
μ SKO96	OK	Worst	OK	OK	33	High
μ LUN99	OK	Best	FAIL	FAIL	33	High
μ BALAS 2012	OK	Good	OK	FAIL	32	Medium
H-inf KWA	OK	Good	FAIL	OK	12	Medium
H-inf CHGE	OK	Good	OK	OK	14	Medium
H-inf Structured	OK	Good	OK	OK	4	Low

Table 4-9: Summary of CDC controllers

The advantage of the KWA, CHGE or “H-Infinity Structured” compared with the μ

controllers are:

- Provides a more automatic methods for the design of the controller (although some trial and error is unavoidable in the selection of the weights),
- Generate controllers of less order (12-14 states vs. 32-33 of the μ controllers)

An additional advantage of the H-infinity structured controllers over all the others is the ability to select a particular control structure. This has been demonstrated adding a roll-off term to comply with the specification S4.

After applying the previous methods to the CDC benchmark that is an 'academic example of control' we will proceed to apply it to a real work case: the control of the VEGA Launcher during first stage.

5 Application to a real world application: the VEGA Launcher

On this second part of the thesis we will apply the previous theoretical results to a real example: the VEGA Launch Vehicle of the European Space Agency. The control of the Launch Vehicle during the atmospheric phase of flight (first stage) will be attempted and confronted against the classical controllers implemented in the real Launch Vehicle.

This part of the thesis has the following structure:

- First the VEGA Launcher vehicle is introduced
- Then the problem of controlling the LV during the atmospheric flight in presence of roll is introduced
- The Physical equations of a Launcher in presence of significant roll rate are described. A linear model is deduced
- A minimalistic but 6DoF representative simulator named “*miniVEGA*” is developed
- Finally a comparison of several controls is performed:
 - o The previous existing controllers (PID based) are described
 - o The proposed new H-Infinity controller is designed and synthesized

5.1 Introduction to the VEGA Launcher

The VEGA LV is the new European Small Launcher developed by the European Space Agency. It will be the reference European launcher for the market of the ‘small launchers segment’ in the next decade. It completes the European strategy for independent access to the space from French Guiana based also in Soyuz for the medium launchers segment and Ariane 5 for the heavy launchers segment.

The VEGA Launch Vehicle is a 30 meters tall, four stage launcher able to deliver payloads of up to 1700 kg to low Earth orbits (Figure 5-1). The first 3 stages use SRM (Solid Rocket Motors) technology. The fourth stage uses a LPS (Liquid Propulsion System) motor. The LPS motor can be re-ignited several times along the flight. This is remarkable for a small launcher and made possible to perform multi-payload missions, being each payload released at a different orbit.

The main benefit of VEGA is its economic design, manufacturing and operation. The economic design is due to two facts: reusability of components from Ariane 5 and economic manufacturing, and operation.

The reusability of components already developed and qualified for Ariane 5 reduces the cost of development (VEGA development cost is estimated to be 1/10th of the Ariane 5 development cost).

The economic manufacturing and operation is provided by two facts: the single-string Avionics (i.e. not redundant Avionics except the Safety Subsystem that is redundant) and the presence of Solid Rocket Motors (SRM) instead of the big Cryogenic Liquid Motor (named Vulcain) present on Ariane 5. In effect, the preparation and operation of a cryogenic engine during the launch campaign increases enormously the costs. (The estimated final cost for a VEGA flight it is estimated in the range of 35/40 m€ compared with 150 m€ of Ariane 5 in figures of year 2012).

The VEGA LV performs its maiden flight the 13th of February 2012. The mission was a remarkable success with no major problems detected and nominal injection of the payload into the desired orbit.

A second flight took part on the 7th May 2013. The mission was again an astounding success with nominal injection of two payload into different orbits.

Third flight (30th May 2014) and fourth flight (11th February 2015) have followed with success. The fact of having the first flights fully successful is very remarkable and totally uncommon for a new launcher.

More information can be found at:

http://www.esa.int/Our_Activities/Launchers/Launch_vehicles/Vega2

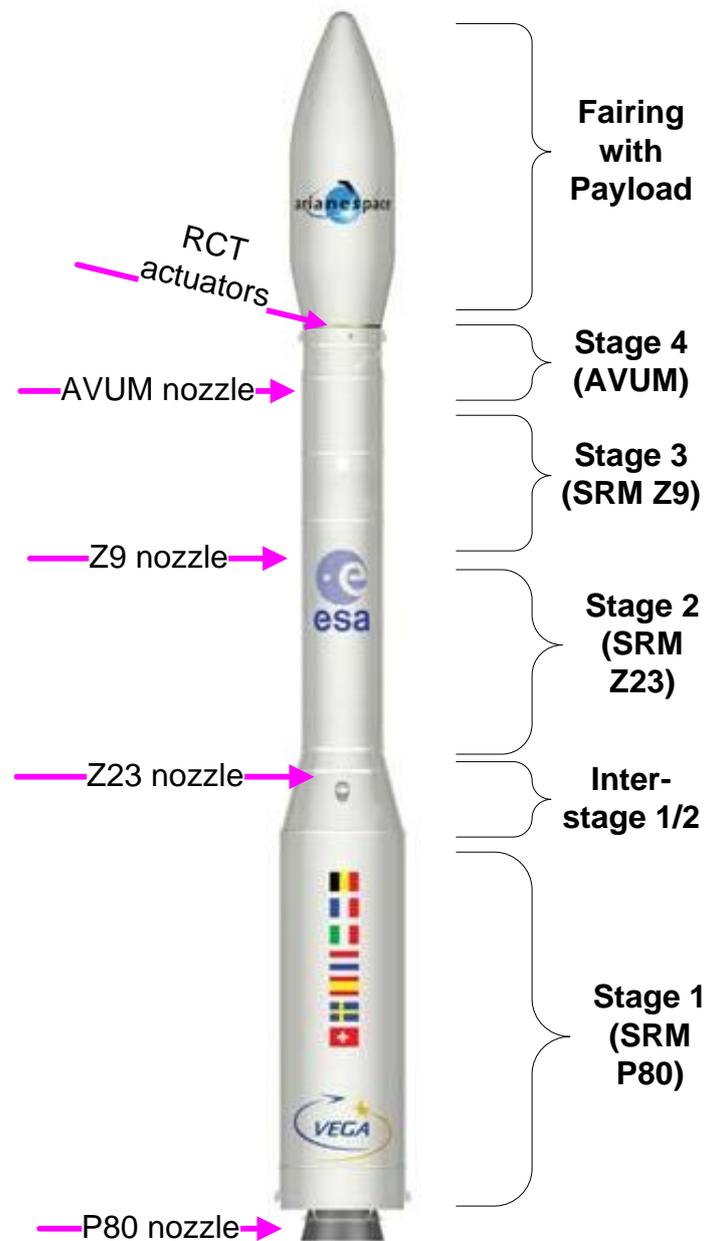


Figure 5-1: Schema of the VEGA Launcher

5.1.1 The Avionics of the VEGA Launcher

The VEGA Avionics is made of several subsystems:

- GNC Avionics subsystem: in charge of the flight management, navigation, guidance and control of the Launcher
- TLM Avionics subsystem: in charge of sending telemetry on real time to ground for safety needs and for post flight analyses
- Safeguard Subsystem: in charge of the auto destruction or commanded destruction under Ground command in case of problems
- Power subsystem: in charge of providing the power to the avionics equipment's

Only the GNC Avionics subsystem is of interest here.

The GNC Avionics Subsystem is composed of the following HW equipment:

- The OBC (On Board Computer), executing the FPS (Flight Program Software) a new development specific for VEGA.
- The IMU (Inertial Management Unit) also known by IRS (Inertial Reference System): reused from the Ariane 5 Launcher
- The MFU (Multifunction Unit). A new development specific for VEGA. In charge of converting the digital commands elaborated by the FPS in orders for the following actuators;
 - o Valves of the Reaction Control Thrusters (RCT): used by the Roll Attitude and Control Subsystem
 - o Valves of the Main Engine AVUM: user for switching on / off of the Liquid Motor engine
 - o Pyro charges activation: used for SRM ignitions, separations of stages, separation of payloads, etc...
 - o Electrical orders, dry-loops: are services offered to the payload
- The TVCs (Thrust Vector Control). Four TVCs are present, one for each stage. A new development specific for VEGA. In charge of deflection of the nozzle according to the digital commands elaborated by the FPS
 - o Each TVC is made of:
 - The Integrated Power and Digital Unit (IPDU): executes the small loop control (described later)
 - Two Electromechanical Actuators (EMA): the physical actuators that moves the nozzle

The (Figure 5-2) describes how the GNC subsystem operates:

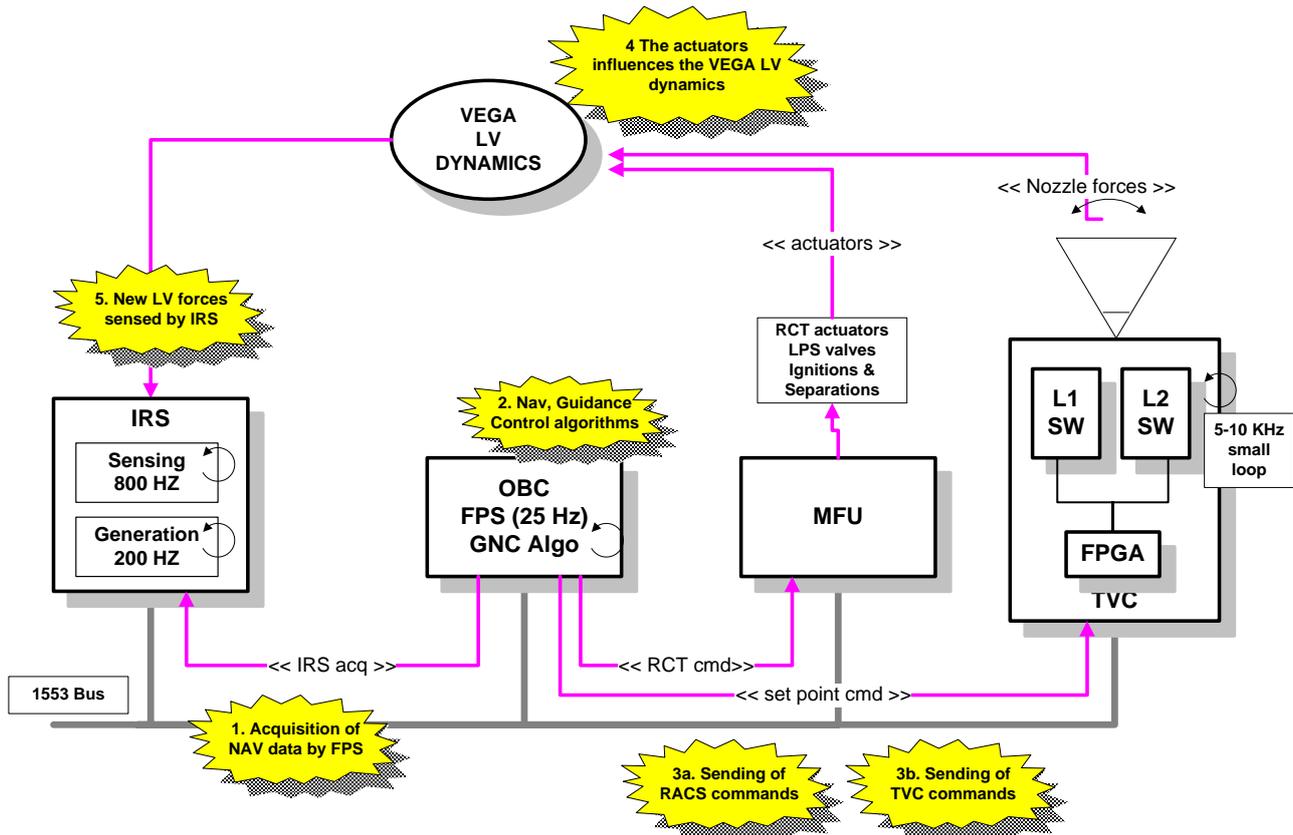


Figure 5-2: The closed loop of the VEGA Launcher

The GNC subsystem operates on a basic cycle of 40 ms (called major cycle)

- The FPS request to the IMU the navigation information (accelerations in IMU body frame, velocities in inertial IMU frame, quaternion of the IMU frame with regard the inertial frame)
- The FPS executes the Guidance, Navigation, Control loop.
 - o First the Navigation algorithms are executed. They compute the LV velocity and attitude based on the raw data provided by the IMU
 - o Then the Guidance navigations are executed. The delta attitude error is computed as difference between the programmed (reference) attitude and the measured one. Same for the position and velocity
 - o Then the Control algorithms are computed. The Control algorithms calculates
 - The 'set point' commands for the TVC control: the desired deflection for the active nozzle is transformed to linear elongations for each EMA actuator
 - The open/close commands for the RCT valves of the Roll and Attitude Control System
 - o By LV controllability reasons, the time from 'get data to the IRS' to the generation of the actuators command is required to be < 12 msec. The rest of the time inside the major cycle the FPS executes other functionality (flight management, FDIR, telemetry, etc...)

- The information is acquired from the sensors (IMU) and transmitted to the actuators electronics through a MIL-STD-1553B bus
- The Integrated Power and Digital Unit (IPDU) of the TVC receive the desired EMA elongations. The TVC SW executes the so called 'small-loop' control, elongating each EMAs up to achieve the desired nozzle deflection, and keeping this elongation (and so the deflection) in face of disturbances until a new set point position is sent by the FPS
- The torque produced by the deflected nozzle motivates a change of attitude of the launcher
- The IMU senses the new attitude and position, and made it available for the FPS, that will start a new cycle 40 ms later

It is important to mention the delays that are present due to the digital processing by the OBC, IMU and TVC computers (the MFU has no digital computer but a FPGA):

- The IMU guarantees that the data provided has a maximum ageing of 10 ms
- The OBC takes the data from the IMU and produces a new pair of TVC commands with a deterministic delay of 12 ms
- The TVC receives the TVC commands and process them initiating the EMAs movement in less than 15 ms
- The MFU receives the MFU commands and process them commanding the RCT valves opening - closing in less than 2 ms

5.1.2 The GNC algorithms of the VEGA Launcher

The algorithms are divided on 3 main blocks:

- Navigation (where I am?)
- Guidance (where I should go?)
- Control (how shall I command my actuators to get there?)

The control in turn is divided on two blocks:

- TVC control: in charge of commanding the TVC and thus the nozzle of the active stage. Only used when one of the motors is active
- RACS control: in charge of commanding the RCT (Reaction Control Thrusters) for roll rate control and fine attitude control during the orbital phases

The (Figure 5-3) shows an overview of the dependencies

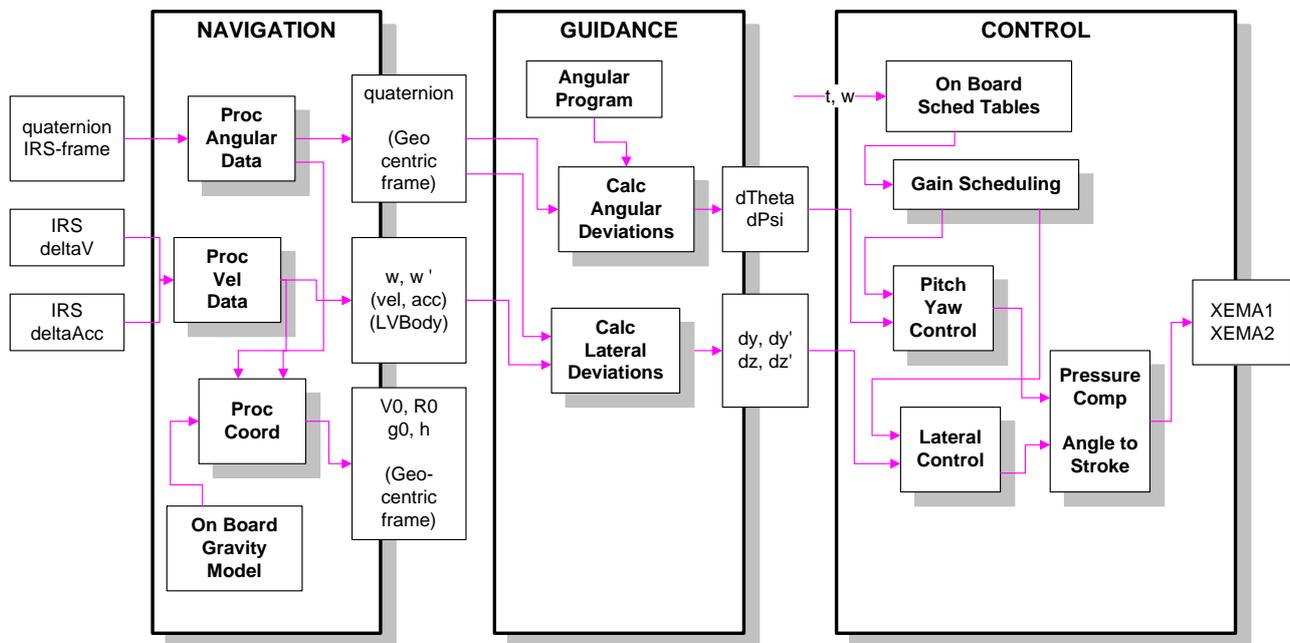


Figure 5-3: Simplified schema of the algorithms

It shall be taken into account that the previous figure is very simplified:

- The Flight Management (FM) algorithm is not shown. The FM is in charge of detection of the events during the flight (ignitions, separations). For example, the SRM separations are detected when the longitudinal acceleration decreases (due to SRM thrust exhaustion) under a threshold (mission defined). In addition, the FM activates or deactivates the other GNC algorithms according to the phase of the flight.
- The Guidance algorithm is implemented by 2 algorithms along the flight.

- An Open Law Guidance is implemented during the atmospheric (P80 and Z23) phases. On these phases the guidance sends the preprogrammed attitude commands to the Launcher on an open loop fashion: there is no intent of correction of the trajectory errors other than to limit the lateral deviations. This is because correcting them would imply a big loss of energy. The philosophy is 'first, get out of the lower atmosphere, when out of the atmosphere, correct the deviations'
 - Both, an Open Loop and a Closed Loop Guidance are implemented during the Z9 and AVUM phases (orbital phases). The predicted orbit is calculated and compared against the target orbit. The TVC are commanded to compensate the deviations
- The RACS control is not shown on the figure
- During the SRM phases, the role of the RACS is limited. Just the roll rate is kept under a threshold by using the RCT actuators
 - During the orbital phase, the RACS performs 3D attitude control of the spacecraft

The approach of the VEGA GNC algorithms is a classical 'gain scheduling approach'. Depending on the phase of the flight and as commanded by the Flight Management, the control strategy is changed, affecting:

- Change of the control strategy (measures selected): example usage or not of lateral feedback
- Change of filters and PID gains
- Usage of open loop or closed loop Guidance

5.1.3 The original TVC control of the VEGA Launcher

The (Figure 5-4) describes in more detail the TVC control that is the object to be studied in the frame of the thesis.

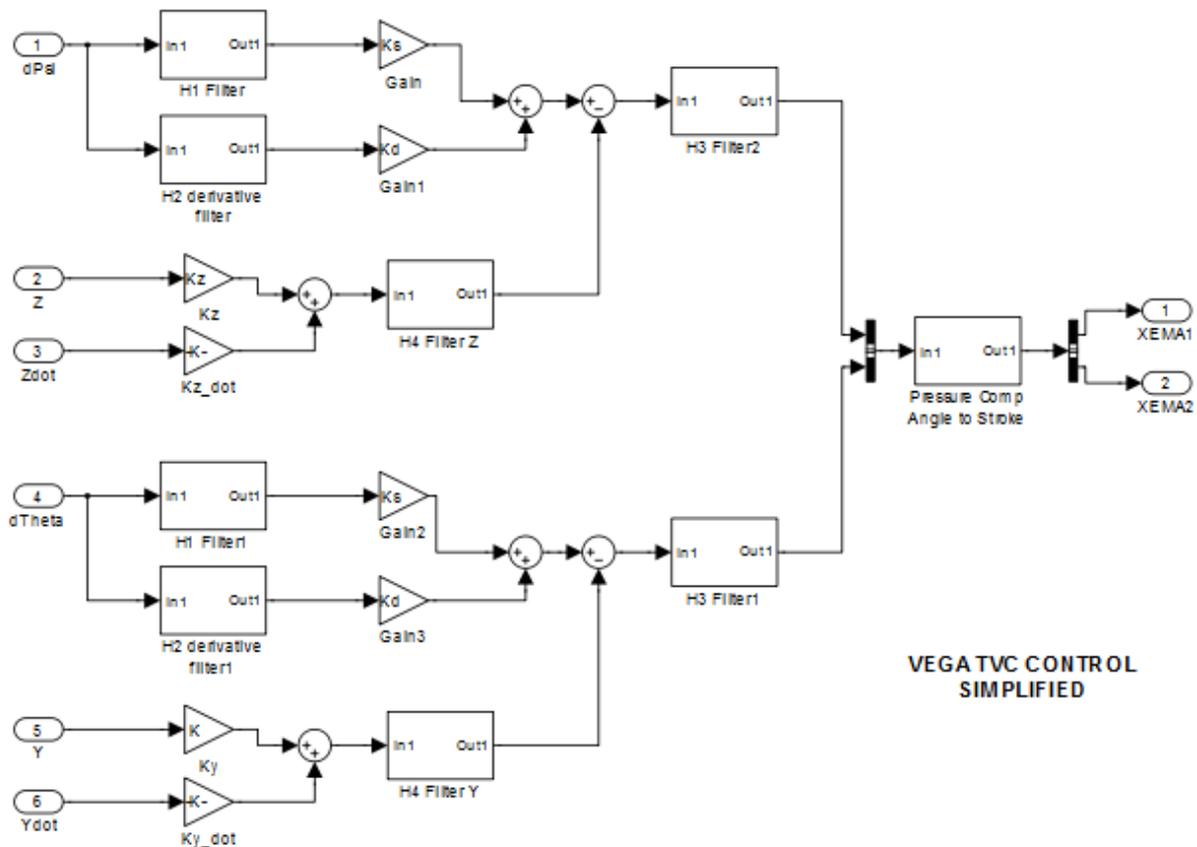


Figure 5-4: The original TVC control

Note that the control is basically divided on 2 decoupled loops (this is only true on absence of LV roll, see later discussion). One loop is the control on the *Psi* plane. The other loop is the control on the *Theta* plane.

Note:
 Reader shall be aware of the different conventions used for the Euler angles in the aerospace field. For VEGA *Phi* is the roll angle (around X), *Psi* is the yaw angle (around Y), *Theta* is the pitch angle (around Z). These conventions are defined formally later.

The *Psi* control takes as inputs the *Psi* angle deviation and the *Z* lateral error and its rate (as computed by the Guidance).

The *theta* control takes as inputs the *theta* angle deviation and the *Y* lateral error and its rate (as computed by the Guidance).

First, the *Psi* error shall be filtered (filter H_1), because the signal elaborated by the Guidance is noisy. This is due to physical limitations of the IRS equipment: the measured angles and velocities increments are noisy.

The Ψ deviation is also processed by the filter H_2 that is a derivative filter that in fact calculates the Ψ error rate. These elaborated signals are the input for a classic PD controller.

The lateral error Z and the lateral error rate \dot{Z} are the inputs of a proportional controller. A low pass filter H_4 is added on the atmospheric phase of flight in order to decrease the activity due to high frequency lateral deviations that shall be not compensated at risk of losing too much energy. In fact, this filter made the control on lateral velocity and position very limited.

The difference between the angular and lateral terms is feed to a third filter, H_3 . This filter is an unstationary notch filter that limits the actuation to a given interval of frequencies. The mission of this filter is to control the bending modes.

The same applies to the other channel (θ and Y).

The previous processing calculates the required deflections of the nozzle on Ψ and θ planes. Several operations have still to be performed:

- Compensation of the nozzle pivot point displacement. This is an effect due to the high pressure that the combustion exercises on the nozzle pivot point (the pressure is so high that the bottom case of the engine is deformed, so the nozzle pivot point descends). The compensation is based on a prediction of the SRM thrust along time. The control algorithm shall elongate the TVC EMAs for compensating this effect and keep the nozzle 'aligned' with the longitudinal axis of the LV
- Conversion to the TVC EMA actuation plane (rotated 45 deg with regard Ψ and θ planes)
- Conversion of angles to EMA elongation (linear displacement of each actuator)

Note that these operations shall be applied independently of the control method used.

It shall be taking into account that all the filters and gains are scheduled during the flight, based on the flight time (for the P80 stage, based on current velocity).

The tuning of such filters to respect the requirements on the entire flight envelope is one of the most time consuming and difficult tasks of the GNC development. The approach is both, analytical and empirical. The GNC experts select initial filters and gains based on its experience that are tuned by means of repetitive Montecarlo simulations.

5.2 VEGA atmospheric flight in presence of roll

5.2.1 Background of the roll problem

The VEGA LV is a symmetrical body around the longitudinal axis.

In principle, the system is decoupled with respect the TVC inputs:

- A deflection on the nozzle on a given plane only changes the orientation of the launcher on that plane (Figure 5-5).
- This is different from an airplane, where a change on one of the inputs is coupled with more than one state: for example, increasing the power of the engine of the airplane not only changes its longitudinal speed, but also its vertical speed because the increment of velocity increments the lift (making the airplane to climb)



Figure 5-5: No coupling in absence of roll

A decoupled system is less interesting for the modern control techniques because a classical SISO approach 'a loop at a time' can be adopted with good results.

This approach (a loop at a time) was the first originally considered for the design of the VEGA LV Thrust Vector Control. The control on the pitch and yaw planes were considered decoupled, and each channel was designed separately.

In an ideal flight, the launcher does not rotate. (In small missiles, auto-rotation can be induced for improving the stability, but this is not the case for a big vehicle as a launcher). But in a real flight, an amount of roll rate will be present.

The causes of roll have been modeled and studied in detail in (Cruciani, 2008)

. These causes are:

- Roll torque induced by geometrical imperfections
 - o Imperfections on the distribution of the propellant inside the SRM case. This made the COG not aligned in the longitudinal axis
 - o Imperfections on the alignment of the nozzle and the longitudinal axis
 - o As result, the SRM thrust axis, the LV longitudinal axis and the COG will be not totally aligned
- Roll torque induced by the combustion
 - o Due to the internal dynamics of the combustion flux and not symmetrical erosion of the nozzle
- Aerodynamics and winds: the LV is not totally symmetrical due to the presence of the 'raceways' (external conducts for the harness). The wind induces an aerodynamic roll when interacts with these protuberances

The most important of the previous causes is the geometrical imperfections. A misalignment of the COG of several centimeters is able to product a torque of thousands

of newtons. These imperfections depend on the manufacturing process of the launcher and can be reduced by strict quality control but never removed.

Under the presence of roll rate, the system becomes coupled. This can be explained intuitively as follows: if we need to change the pitch angle (and only the pitch angle), we command the actuator EMA-i for doing this. If an amount of roll is present, the deflection takes place with some delay (the accumulated of the digital delay due to the computers and the physical delay associated to the EMA elongation). As the LV is rotating, the order will be executed with the actuator rotated from the original position, so giving a main contribution to change the pitch but also a small contribution to the yaw axis.

The VEGA LV control is required to be robust to roll rates as high as 45 deg/s during the first stage (if roll rate is greater than 90 deg/s, the IMU would be out of its qualification domain and disaster may occur).

Note:

Even if high roll rates are needed to destabilize the launcher, the effect is in general undesirable for two reasons:

First reason is that stability margins are decreased under high roll rate. Even if the high roll rate would not de-stabilize the launcher a wind gust in such moment could create problems.

The second reason is a lesson learned from the Ariane 5 second flight. A roll rate greater than expected was present. In principle, the roll rate was under the limits tolerated by the GNC but created a secondary effect: the liquid propellant accumulated against lateral walls of the tank propellant. The sensor that measured the amount of propellant was located in the center of the tank. So it provided to the central computer underestimated measures of available propellant. As result, the On Board SW decided to shut down the main engine when in fact there was still propellant to complete the mission. The result was a high degradation of the orbit (orbit was too low).

Note:

During the first flight the roll during the P80 stage was minor than 10 deg/s. This can be explained by low geometrical imperfections and low winds found. Of course, it is not guaranteed that this value is not surpassed in successive flights.

The coupling under the presence of a roll rate has been taken into account in VEGA design as follows:

- Studies were made for the characterization of the problem, refer to (Roux, 2007).
- The SISO approach has been kept, but a gyroscopic compensation term has been introduced. This approach is described later.

In order to limit the roll rate the control strategy was changed in VEGA. Note that the design team faced a difficult trade-off:

- If a low roll rate is to be achieved during atmospheric flight, powerful RCTs actuators are required to limit the roll rate. But powerful RCT are a disadvantage during the orbital phases, where small thrust pulses are required to achieve a fine orientation of the payload. Basically a powerful RCT is imprecise and vice versa.
- Adding new actuators– i.e. a set of power but low precision RCTs for the atmospheric flight and a second set of small but high precision RCTs for the orbital phases was estimated prohibitive on term of complexity, weight and costs

The alternative chosen by the VEGA designers is that only the roll rate is to be limited (i.e. the RCT actuators only acts when the roll rate is over a threshold). There is no attempt to control the roll angle. Note also that due to the low torque achievable with the implemented RCT when compared to the huge LV inertia of the vehicle during the first stage it is not possible to keep a roll rate near to zero. Instead a roll rate threshold that increases along the stage and reach about 45 deg/s is defined.

5.2.2 The updated TVC control with roll compensation

The control strategy was modified by introducing a gyroscopic compensation term (Figure 5-6). This term computes the angular rates from the quaternion (elaborated by the Navigation algorithms). With the angular rate and a gain a compensation factor is calculated for each channel.

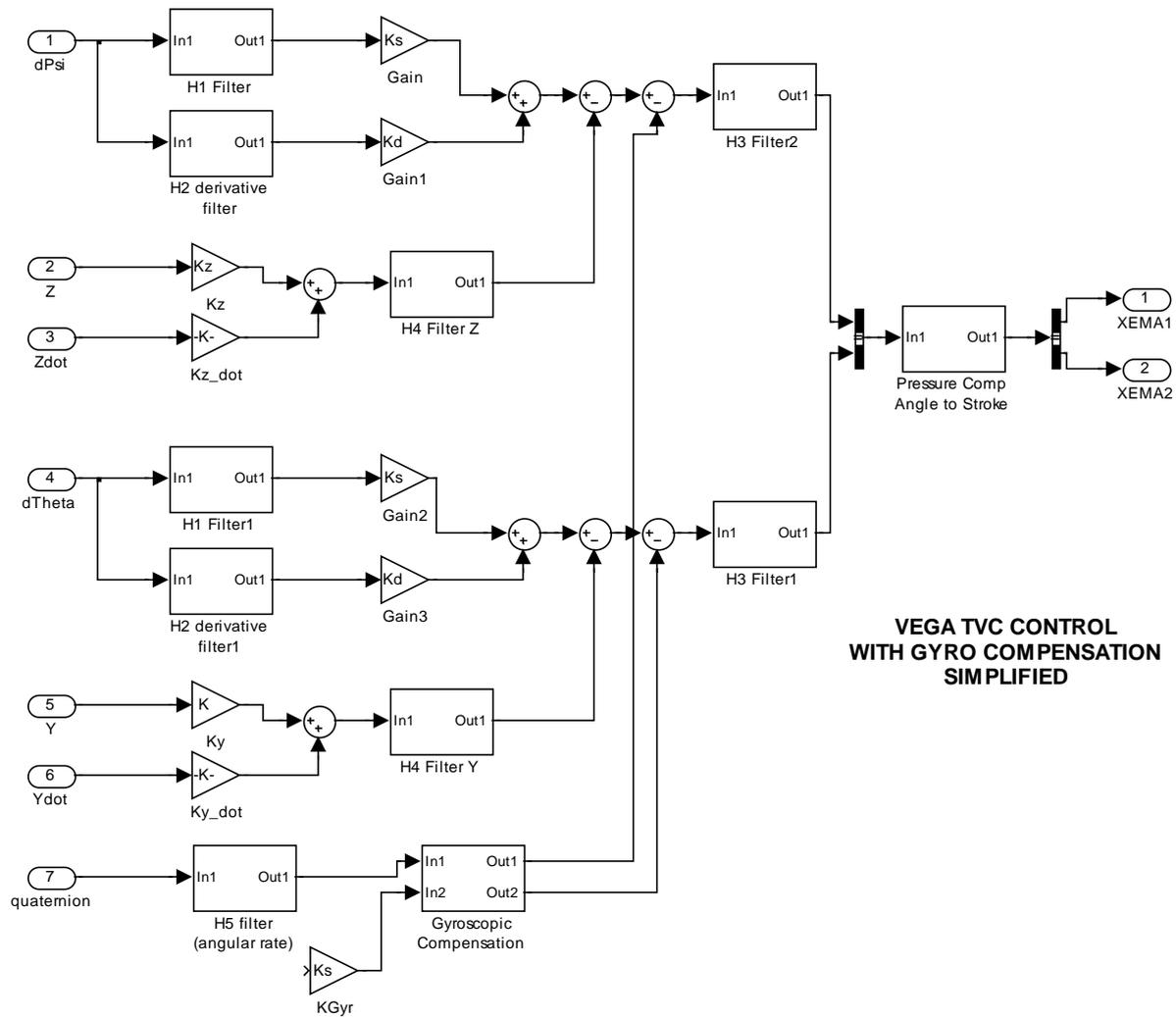


Figure 5-6: TVC control with gyroscopic compensation

The introduction of the gyroscopic compensation requires the inclusion of a new derivative filter H5. This H5 filter computes the derivative of the quaternion (the IMU does not provide angular velocities).

The angular velocity is obtained from the quaternion and quaternion rate as follows. The quaternion rate is obtained with a derivative filter (H5 filter) from the quaternion.

$$\omega = q^{-1} q'$$

And then:

$$p = 2.0 * q(2)$$

$$q = 2.0 * q(3)$$

$$r = 2.0 * q(4)$$

Finally, the compensation is calculated as:

$$\text{ThetaGyro} = \text{KGyro} * p * q$$

$$\text{PsiGyro} = - \text{KGyro} * p * r$$

Also the gain KGyro that shall be scheduled along the flight based on the flight time.

The gyroscopic compensation method has several disadvantages:

- It is based on a nominal inertia model and can produce more problems that benefits if during the flight the real inertia differs significantly from the model.
- Even if gyroscopic compensation is perfect it will cancel only the roll coupling in the rotational dynamics and not on the translational dynamics. This limitation is intrinsic to the method.
- The use of the gyroscopic compensation implies a new set of gains and filters that shall be tuned adding complexity and work

5.3 Physical Equations of the VEGA Launcher

5.3.1 Reference Frames and Conventions

5.3.1.1 Reference Frames

There are a lot of different reference frames defined for the VEGA Launcher. We will define in this work some of them but for the scope of this thesis we will work only with three of them.

LV Body Reference Frame

It is a non-inertial frame, fixed to the Launch Vehicle (Figure 5-7).

Axis X is parallel to LV longitudinal Axis positive pointing versus the tip of the rocket.
Axis Z vs. RCT cluster 1 (EMA 1 is on +Z).

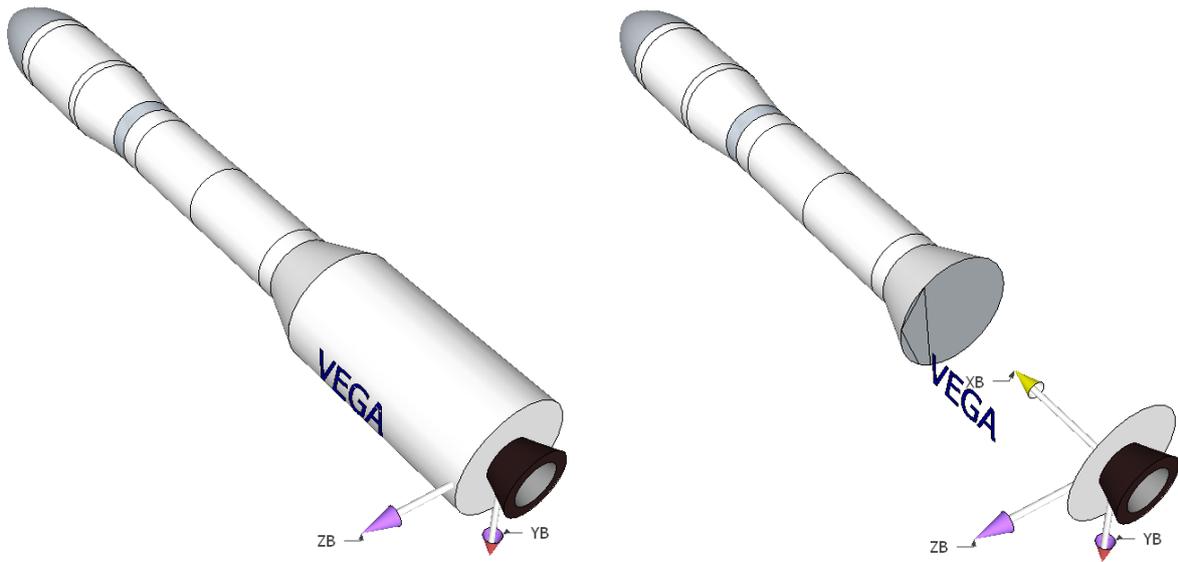


Figure 5-7: LV Body frame

The (Figure 5-8) details the position of the thrusters (the body frame is displayed close to the thrusters just for clarity).

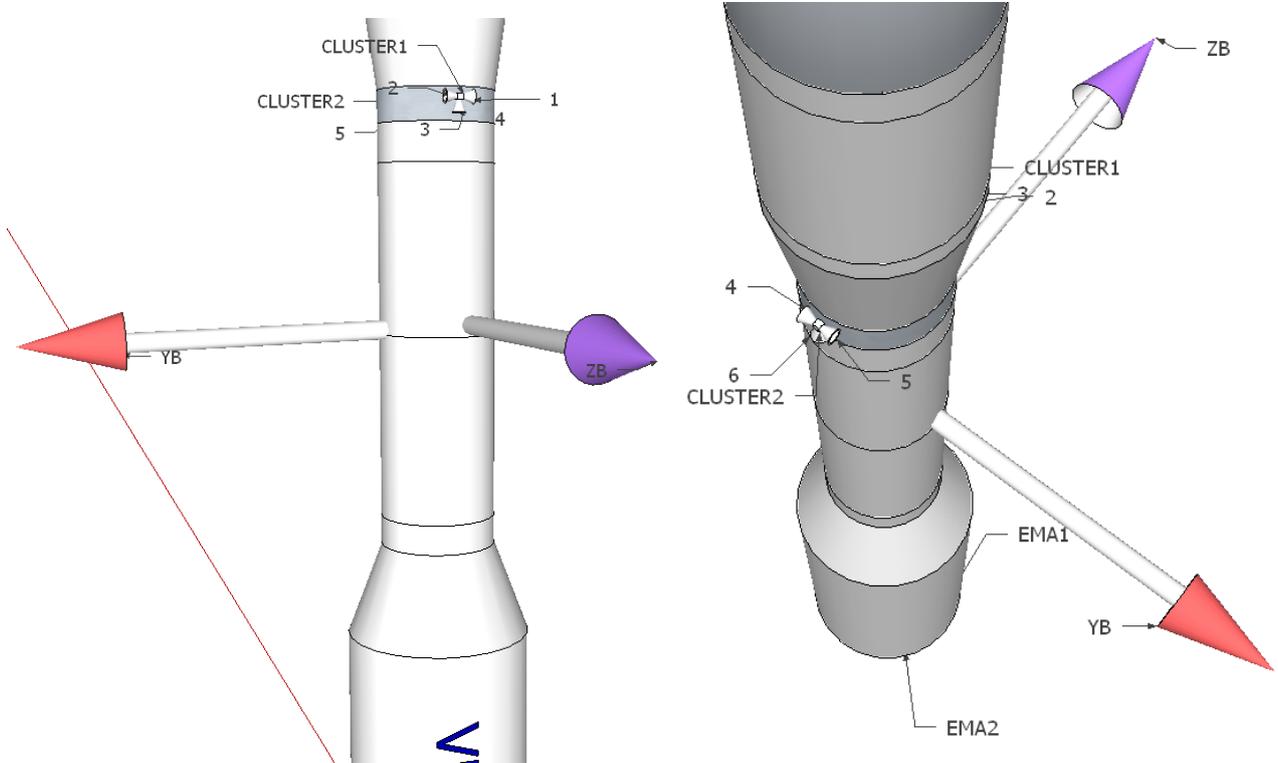


Figure 5-8: View of the RTCs

The (Figure 5-9) details the position of the EMAs.

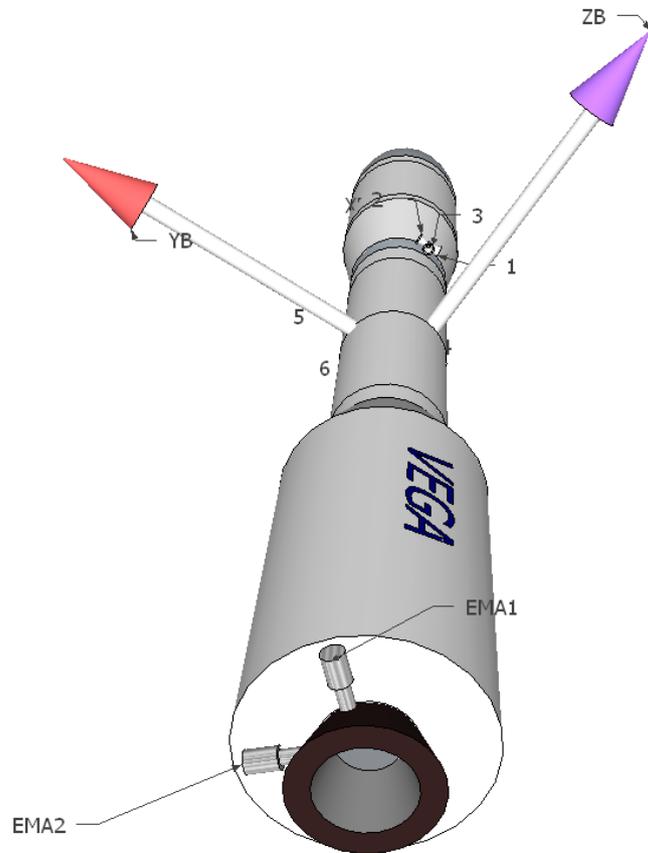


Figure 5-9: Position of the EMA actuators

Initial Starting Reference Frame

It is an inertial RF. Axis Y is aligned with the longitudinal LV Axis (i.e. Y is aligned with the local vertical). X is in the launch azimuth axis (roughly the orbit inclination). Z completes a right handed axis. Origin is at Kourou launch pad (Figure 5-10).

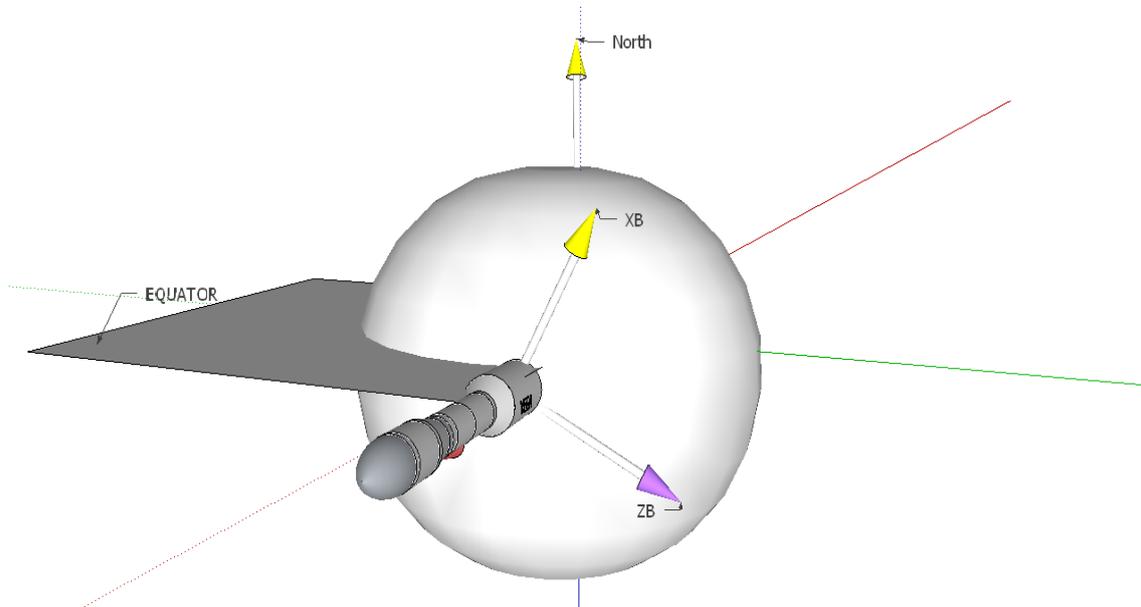


Figure 5-10: LV Initial Starting reference frame

Initial Starting Geocentric RF:

It is an inertial RF, parallel to previous but with origin in the Earth Center.

(Real) Navigation Reference Frame

In the real Launcher the inertial Navigation Reference frame has origin in the IMU equipment. The orientation is the same that the Initial Starting Reference frame. This frame is set when the IMU is configured in 'flight mode' 2 seconds before the Launch.

In the scope of this thesis we will consider just that this frame is inertial and has some initial known position (see next paragraph).

Trajectory Reference Frame

It is a non-inertial frame that moves along the ideal trajectory, being X tangent to the trajectory and Z parallel to local horizon.

Other Reference Frames

More frames are used for the computation of the orbit: the Orbital basic Frame, the Sidereal frame, etc. These are used only for long term guidance and are of no interest here.

5.3.1.2 Simplified Reference Frames

In spite of all the previous frames we only need to consider 3 frames when working with the short term dynamics (attitude control) that is the one addressed in this thesis:

- The Navigation reference frame (F_N). Considered full inertial in a given instant along the trajectory.
- The Guidance reference frame (F_G). Rotates with the vehicle (Y_G parallel to Y_B , (i.e. the Guidance frame see a roll angle zero and a roll rate zero). Plane YZ_G remains parallel with the plane YZ_N inertial (i.e. in some figured sense the plane YZ_G remains inertial)
- Body reference frame (F_B). Attached to the vehicle and moves with it (not inertial)

In an initial instant we consider the origin of the 3 frames coincides and are aligned.

The (Figure 5-11) shows the 3 frames the launch vehicle. (Warning: for clarity in the figures the 3 axis do not have the same origin).

Note that:

- Y_G is parallel to Y_B (i.e. guidance frame rotates with body frame and launch vehicle)
- X_G remains parallel to X_N (plane YZ_G remains parallel with the plane YZ_N)
- X_B, Z_B are not parallel to X_G, Z_G .
 - o The pitch angle is the angle around Z_G between X_B and X_G
 - o The yaw angle is the angle around Y_G between Z_B and Z_G

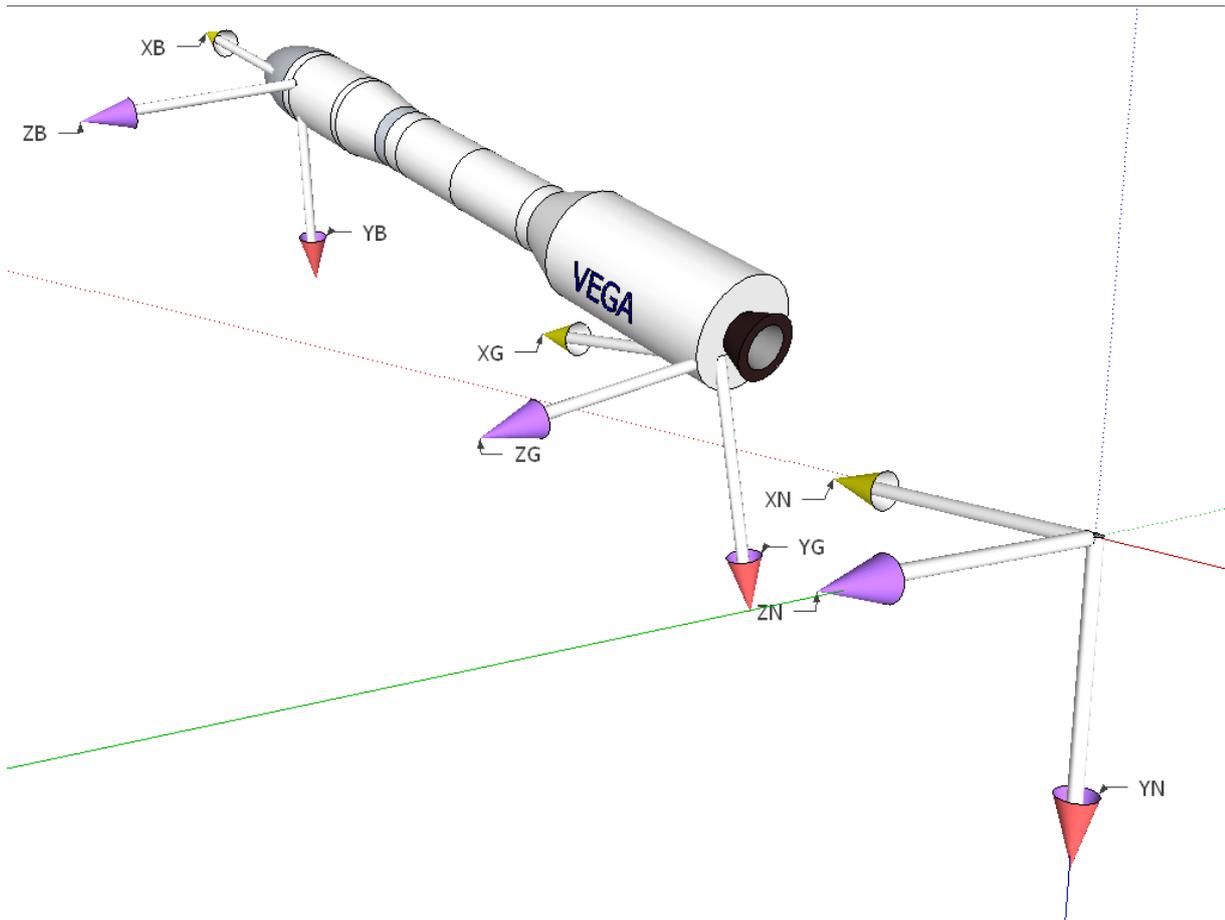


Figure 5-11: Navigation, Guidance and Body RFs

The (Figure 5-12) shows another perspective for clarifications purposes:

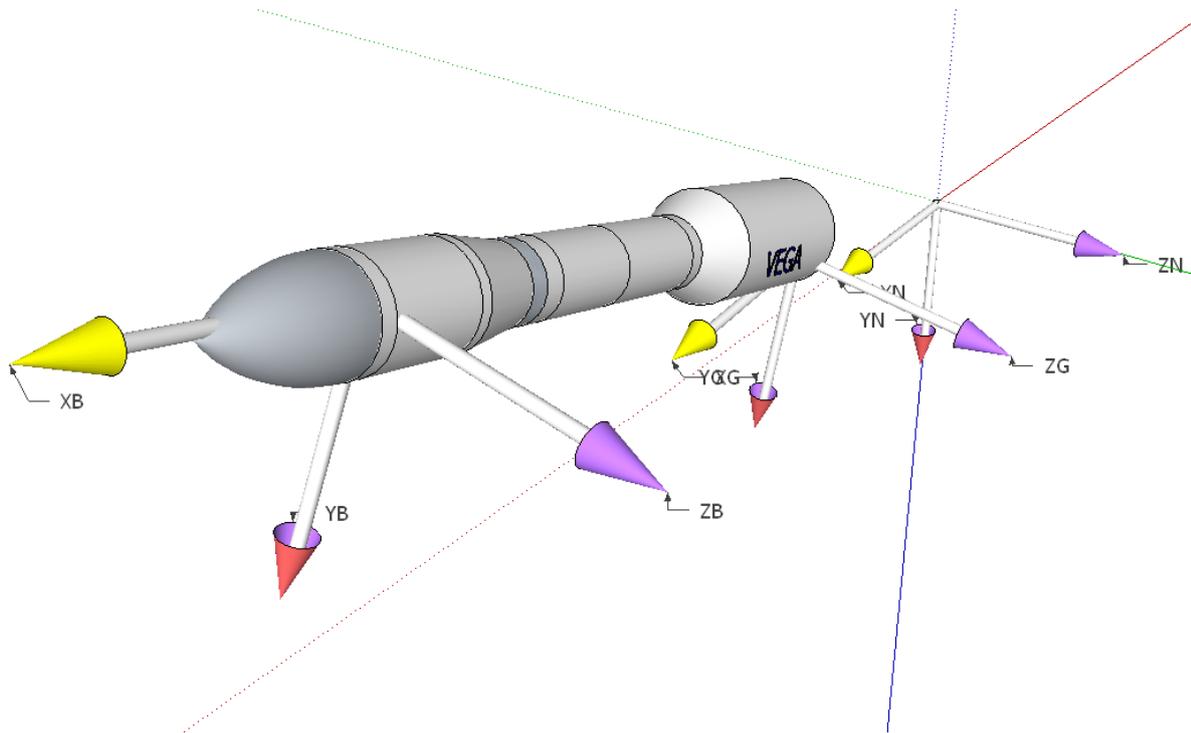


Figure 5-12: Navigation, Guidance and Body RFs (2)

5.3.1.3 Angle conventions

The Euler convention in VEGA is: for conversion from Navigation (inertial) to LV Body RF:

- First rotation about Z_N (theta)
- Second rotation about Y_G (yaw)
- Third rotation about X_B (phi)

So Navigation (inertial) to LV body RF is defined in order (pitch, yaw, roll).
Conversely, LV Body to Navigation is defined in order (roll, yaw, pitch).

The sign of rotation is defined by the “right hand rule”.

5.3.2 Description of the forces and momentums acting on the LV

5.3.2.1 Aerodynamics forces and momentums

The flight of the LV through the atmosphere generates aerodynamics forces and torques. These effects are only significant during the P80 phase and begin of the Z23 phase. At higher altitudes the low density of the atmosphere made aerodynamics contributions negligible.

The aerodynamics forces are obtained as empirical laws function of:

- The square of the modulus of the velocity

- The reference surface (average surface normal to the velocity vector)
- The aerodynamic coefficients C_N , function of the angle of attack and the Mach number
- The Centre of Pressure CP point, also function of the angle of attack and the Mach number
- The air density

For a launch vehicle as VEGA without aerodynamic surfaces and when the vehicle follows a gravity turn maneuver (explained later) the aerodynamics forces do not create lift but only drag.

The aerodynamics forces act in the Centre of Pressure CP point. The aerodynamics momentums depend on the relative position of the CP and the COG .

The coordinates of the CP varies along flight. The position is calculated extracted from tabulated tables extracted from the wind tunnel tests. The position of the COG also varies during flight because the propellant mass is being consumed. For the VEGA LV the COG is always behind the CP .

The air density is tabulated and defined empirically in function of the altitude.

The angular momentum due to aerodynamics is composed of two terms:

- Angular moments due to the aerodynamics forces
- Pure aerodynamic roll

5.3.2.2 Propulsive forces and momentums

The propulsive forces are due to the burning of the propellant in the combustion chamber. These forces act in the Pivot Point of the nozzle (PP). If the nozzle is aligned with the LV longitudinal axis (X axis), the propulsive force is aligned to the LV longitudinal axis. The nozzle can be rotated in the XY or XZ planes. When the nozzle rotates the propulsive force is not aligned with the LV longitudinal axis and creates a torque. This is the main way to control the LV trajectory. See (Figure 5-13) and (Figure 5-14).

It shall be noted that by definition the torque due to a propulsion force has the opposite sign to the torque due to an aerodynamic force of the same sign. This is due to the fact that the torque force acts in the PP (pivot point) that is behind the COG . Instead the aerodynamics force acts in the CP (center of pressure) that is ahead of the COG . Example, a propulsive force with Z component negative and an aerodynamic force with Z component negative will provoke a torque with different sign (i.e. made the vehicle rotate along Y axis on opposite senses).

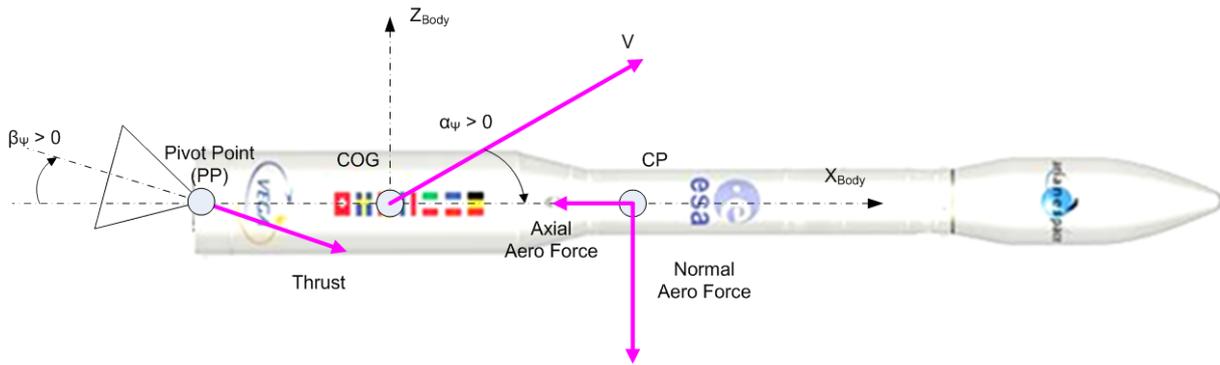


Figure 5-13: Propulsive and aerodynamics forces in XZ (yaw) plane

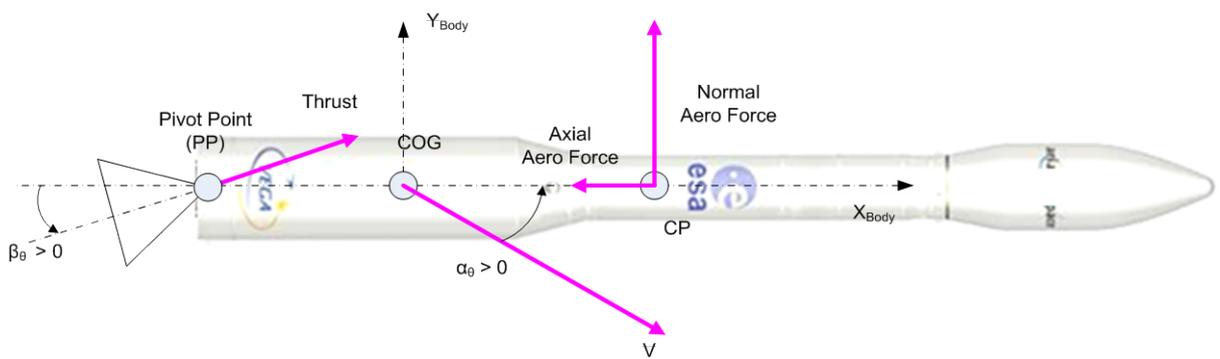


Figure 5-14: Propulsive and aerodynamics forces in XY (pitch plane)

The propulsive force varies along the time. The SRM for the first stage has the typical profile as shown in (Figure 5-15). (The duration of the first stage is about 115 seconds).

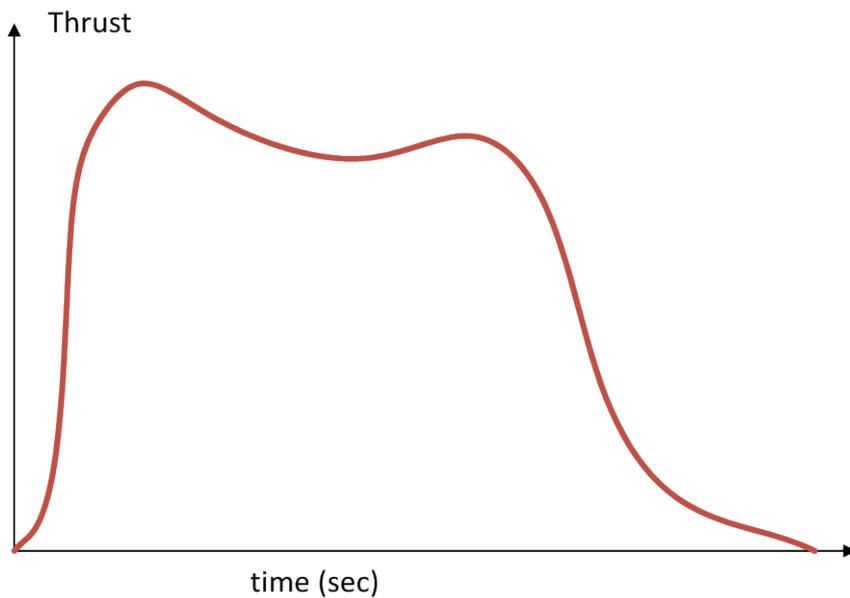


Figure 5-15: Thrust profile for first stage

The thrust increases very fast in the first seconds of burning, then reaches a maximum and decreases to a value where remains almost constant. At the end of the combustion the thrust decreases very fast and finishes with a slow decrease called the “tail-off”. The SRM thrust profile depends on the propellant chemical characteristics and the layout of the propellant inside the stage. In fact the profiles are different for the second and third stages.

5.3.2.3 Nozzle torque

The P80 nozzle has a significant mass and inertia, although it is much minor than the “LV Total Mass” during the first stage.

However in the tail-off the relation “Nozzle Mass” to “LV Total Mass” increases (as “LV Total Mass” decreases). Also the TVC is commanded with higher angles during the tail-off in order to compensate the lack of thrust.

As this work refers mainly to the region of maximum dynamic pressure ($t = 55$ seconds) where the relation “Nozzle Mass” to “LV Total Mass” is small, the nozzle torque has not been modeled in *miniVEGA*.

5.3.2.4 Gravity force

The Earth Gravity field is function exclusively of the position with regard the Earth center. As the Earth is not a perfect sphere, the gravity on the surface depends on the concrete latitude and longitude.

The applicable model to VEGA is the WGS-84.

During the first stage the LV follows a “Gravity turn” (or zero-lift) maneuver. This is a trajectory where the LV uses the gravity force for changing slowly the trajectory from the vertical to the desired inclination. The utility of the gravity turn is double: first by using the gravity instead of deflecting the nozzle all the thrust is dedicated to increase the longitudinal velocity. Second, the no deflection of the nozzle allows to flight with nearly zero angle of attack minimizing the structural loads.

The gravity is used for changing the trajectory but it can be demonstrated that for steady changes of the flight path angle the effect on the LV acceleration is minor. A demonstration is given in the appendix 8.4.1.

Of course, this approximation is only valid for the atmospheric phase, for small time periods and in the scope of short term attitude control (but this is indeed the scope of this thesis).

5.3.3 Non-linear equations (6DoF rigid body)

We first recall the problem of the rotational and translational dynamics of a rigid body with six degrees of freedom when observed from an inertial reference frame.

5.3.3.1 Need of the transport theorem

Suppose a rocket that is rotating with regard a fixed frame. Let's define a body frame F_B attached to the rocket. Let's define an external frame F_I . F_B is rotating with regard F_I with an angular velocity ω (Figure 5-16).

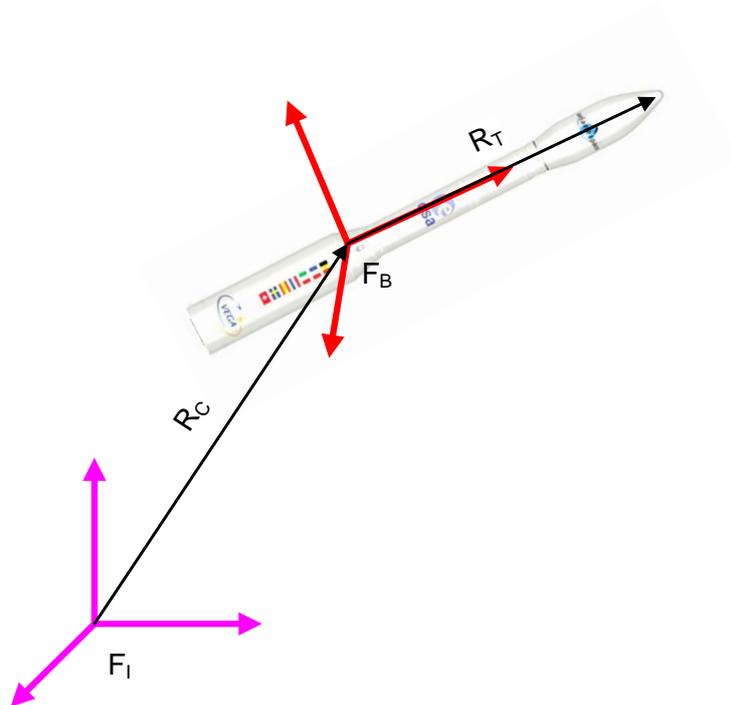


Figure 5-16: Inertial and rotating frames

R_C is the position of the center of mass of the rocket in F_I .

R_T is the position of the tip (nose) in F_B .

We want to compute the acceleration and torque of the tip of the nose in F_I assuming we know it in body frame. We must be careful: the Newton equations are valid only in inertial reference frames.

$$F_I = ma = m \frac{d^I v}{dt} \quad \text{ONLY in inertial frame!}$$

However, we deduce the expression of the acting forces usually in body frame. Any of the following would be incorrect:

$$F_B \neq m \frac{d^I v}{dt} \quad \text{or} \quad F_I \neq m \frac{d^B v}{dt}$$

We have to apply the ‘Transport Theorem’ that relates any vector in inertial frame knowing it in body frame (where it is supposed that the body frame is rotating wrt the inertial frame). See appendix 8.4.2 for a demonstration of the Transport theorem. For example the expression that relates inertial velocity with body velocity is:

$$\frac{d^I \vec{r}}{dt} = \frac{d^B \vec{r}}{dt} + \omega_{BinI} \otimes \vec{r}_{(inB)}$$

The transport theorem is used extensively on the following sections.

5.3.3.2 Rotational equation

The angular momentum L is given by:

$$L = I\omega$$

Let’s assume we take a frame F_B with origin at the center of mass of the body. In this reference frame the inertia value does not change. Let’s assume also a constant mass.

$$\begin{aligned} \frac{d^B I}{dt} &= 0, \quad \text{in } F_B \\ \frac{d^B L}{dt} &= \frac{d^B (I\omega)}{dt} = \frac{d^B (I)}{dt} \omega + I \frac{d^B (\omega)}{dt} = 0 + I \frac{d^B (\omega)}{dt} \end{aligned}$$

The Torque T is the temporal derivative of the angular momentum. We can express the Torque T in an inertial reference system F_I knowing it in body applying the transport theorem:

$$M^I = \frac{d^I (L)}{dt} = \frac{d^B (L)}{dt} + \omega \otimes L = I \frac{d^B (\omega)}{dt} + \omega \otimes (I\omega)$$

This is the general form of the rotational dynamics.

Let’s now take the body frame such that their axes are oriented in the principal axis of the body. By doing this the inertia matrix in body reference frame becomes a diagonal matrix. With this choice of orientation of the body frame we can obtain the Euler Equations of rotational motion:

$$M^I = I \frac{d^B (\omega)}{dt} + \omega \otimes (I\omega)$$

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \omega \otimes \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \omega \otimes \begin{bmatrix} I_{xx}\omega_x & 0 & 0 \\ 0 & I_{yy}\omega_y & 0 \\ 0 & 0 & I_{zz}\omega_z \end{bmatrix}$$

Developing the vector product:

$$\begin{bmatrix} i & j & k \\ \omega_x & \omega_y & \omega_z \\ I_{xx}\omega_x & I_{yy}\omega_y & I_{zz}\omega_z \end{bmatrix} = \begin{bmatrix} \omega_y I_{zz} \omega_z - \omega_z I_{yy} \omega_y \\ -(\omega_x I_{zz} \omega_z - \omega_z I_{xx} \omega_x) \\ \omega_x I_{yy} \omega_y - \omega_y I_{xx} \omega_x \end{bmatrix}$$

We arrive to the Euler rotational equation:

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \begin{bmatrix} (I_{zz} - I_{yy})\omega_y \omega_z \\ (I_{xx} - I_{zz})\omega_x \omega_z \\ (I_{yy} - I_{xx})\omega_x \omega_y \end{bmatrix} \quad (\text{Eq 5-1})$$

The equation is coupled and not linear. In general the equation can be solved analytically only if some simplifications are assumed:

- Inertia varies slowly
- Mass varies slowly
- Roll rate is constant

5.3.3.3 Translational equation (Tsiolkovsky formula)

The translational equation **for a 3DOF rocket** is the Tsiolkovsky formula. Consider a rocket that is exhausting propellant (Figure 5-17):



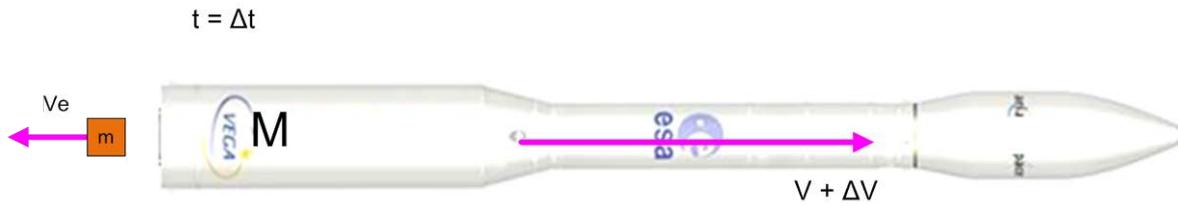


Figure 5-17: Deduction of Tsiolkovsky formula

In a inertial frame of an external observer, at $t = 0$, the momentum is

$$P_1 = (m + \Delta m) v$$

And at $t = 0 + \Delta t$ is,

$$P_2 = m (v + \Delta v) + \Delta m * v_e$$

Where v_e is the exhaust velocity of the particles in the inertial frame

We can express v_e in the body frame as $v_{eb} = v_e - v$

$$P_2 = m (v + \Delta v) + \Delta m * (v_{eb} + v)$$

$$\begin{aligned} P_2 - P_1 &= m (v + \Delta v) + \Delta m * (v_{eb} + v) - ((m + \Delta m) v) = \\ &= m \Delta v + \Delta m v_{eb} \end{aligned}$$

Then, taking the limit

$$\sum F_i = \lim_{\Delta t \rightarrow 0} \frac{P_2 - P_1}{\Delta t}$$

Having the translational formula:

$$\sum F_i = m \frac{dv}{dt} + \frac{dm}{dt} v_{eb}$$

The Tsiolkovsky formula is useful when computing the orbit of the rocket. However when we are studying the control of the LV as a 6DoF body (short term attitude dynamics) we are not interested on the longitudinal dynamics (X axis). We cannot control the thrust level but only control the LV attitude by deflecting the nozzle. We also consider that the mass and inertia are constant (or change very slow) in the time interval considered.

5.3.3.4 Translational equation

We have seen that the Transport theorem relates the velocity of a point in an inertial frame knowing the velocity on a body frame.

$$v_I = \frac{d^I r}{dt} = \frac{d^B r}{dt} + \omega \otimes r_B$$

We apply the transport theorem again for computing the acceleration of a point in the inertial frame:

$$a_I = \frac{d^I v_I}{dt}$$

Substituting previous expression for v_I we have:

$$a_I = \frac{d^I \left(\frac{d^B r}{dt} + \omega \otimes r_B \right)}{dt}$$

Applying derivative of a sum:

$$a_I = \frac{d^I \left(\frac{d^B r}{dt} \right)}{dt} + \frac{d^I (\omega \otimes r_B)}{dt}$$

Applying the transport rule to each term:

$$a_I = \left(\frac{d^{2B} r}{dt} + \omega \otimes \frac{d^B r}{dt} \right) + \left(\frac{d^B (\omega \otimes r_B)}{dt} + \omega \otimes (\omega \otimes r_B) \right)$$

$$a_I = \left(\frac{d^{2B} r}{dt} + \omega \otimes \frac{d^B r}{dt} \right) + \left(\frac{d^B (\omega)}{dt} \otimes r_B + \omega \otimes \frac{d^B r}{dt} \right) + (\omega \otimes (\omega \otimes r_B))$$

We arrive to the general formula:

$$a_I = \frac{d^{2B} r}{dt} + 2\omega \otimes \frac{d^B r}{dt} + \frac{d^B \omega}{dt} \otimes r_B + (\omega \otimes (\omega \otimes r_B)) \quad (\text{Eq 5-2})$$

That says that the acceleration that an inertial observer sees in a point of the body subject to rotation and acceleration is composed of the following four terms:

Term 1: is the usual expression for the acceleration (the one an observer attached to the body frame would see)

Term 2: is the Coriolis term $\left(2\omega \otimes \frac{d^B r}{dt} \right)$

This term is due to the fact that the rotating axis is moving.

E.g. suppose a free particle is thrown from the center of a disk that is rotating (we suppose no friction). From the point of view of the inertial observer, the particle follows a straight line (no acceleration). From the point of view of an observer attached to the disk the particle moves away from the axis (so is accelerating).

Term 3: is the variable angular velocity $\left(\frac{d^B \omega}{dt} \otimes r_B \right)$

This term only exist if the angular velocity of the rotating frame is changing (increasing or decreasing)

Term 4: is the centrifugal force $(\omega \otimes (\omega \otimes r_B))$

5.3.4 Derivation and solution of the Linear equations for a rocket

In this chapter we apply the previous general results to the specific case of a rocket. It is considered in this thesis that the rocket is a rigid body.

5.3.4.1 Solution of the Rotational equation

We start by the general Euler equation (Eq 5-1) in presence of external torques derived in 5.3.3.2:

$$\sum \begin{bmatrix} M_{Ex} \\ M_{Ey} \\ M_{Ez} \end{bmatrix} = \begin{bmatrix} I_{xx} \frac{d}{dt} \omega_x \\ I_{yy} \frac{d}{dt} \omega_y \\ I_{zz} \frac{d}{dt} \omega_z \end{bmatrix} + \begin{bmatrix} (I_{zz} - I_{yy}) \omega_y \omega_z \\ (I_{xx} - I_{zz}) \omega_x \omega_z \\ (I_{yy} - I_{xx}) \omega_x \omega_y \end{bmatrix}$$

As seen in chapter 5.3.2, the external torque is due to aerodynamics and propulsion.

Deduction of propulsive torques

The propulsive force in body axis is the Thrust as deflected by the nozzle in theta and psi planes:

$$F_{propX} = T \cos(\beta_\psi) \cos(\beta_g)$$

$$F_{propY} = T \cos(\beta_\psi) \sin(\beta_g)$$

$$F_{propZ} = -T \sin(\beta_\psi) \cos(\beta_g)$$

The propulsive torque applies at nozzle pivot point (r_{PP}). The torque is:

$$M_{Prop} = (r_{PP} - r_{COG}) \otimes F_{PROP}$$

If we consider that r_{PP} is behind of the r_{COG} and that a positive propulsive force creates a torque in opposed sense to a positive aerodynamic force. If we consider the arm of the torque from r_{COG} to r_{PP} we have instead to consider:

$$M_{Prop} = -(r_{COG} - r_{PP}) \otimes F_{PROP}$$

Let's name

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} x_{PP} - x_{COG} \\ y_{PP} - y_{COG} \\ z_{PP} - z_{COG} \end{bmatrix}$$

Making the vector product and taken only moments around y and z, we have:

$$M_y = -(r_x F_z - r_z F_x) = -r_x (-T \sin(\beta_\psi) \cos(\beta_g)) + r_z (T \cos(\beta_\psi) \cos(\beta_g))$$

$$M_z = r_x F_y - r_y F_x = r_x (T \cos(\beta_\psi) \sin(\beta_g)) - r_z (T \cos(\beta_\psi) \cos(\beta_g))$$

Now let's assume that we have small angles for the actuators. In this case we can approximate:

$$\cos(\beta_\psi) \approx 1,$$

$$\cos(\beta_g) \approx 1$$

$$\sin(\beta_\psi) \approx \beta_\psi$$

$$\sin(\beta_g) \approx \beta_g$$

Due the long shape of the LV the distance between the X coordinates of the r_{COG} and the r_{PP} (i.e. the distance along the longitudinal axis) is of the order of meters. It is much bigger that the distance between the Y, Z coordinates of COG and PP that is of the order of centimeters. This is because the LV mass is distributed almost symmetrically in the plane normal to the X axis.

$$(r_x \gg r_y) \text{ and } (r_x \gg r_z)$$

This allows the simplification of the equations as follows:

$$M_y \approx -r_x (-T \sin(\beta_\psi)(1)) + r_z (T(1)(1))$$

$$M_z \approx r_x (T(1) \sin(\beta_g)) - r_z (T(1)(1))$$

$$M_y \approx (x_{PP} - x_{COG}) T \beta_\psi$$

$$M_z \approx (x_{PP} - x_{COG}) T \beta_g$$

Or if we consider r_{COG} to r_{PP} :

$$\begin{aligned} M_y &\approx -(x_{COG} - x_{PP})T\beta_\psi \\ M_z &\approx -(x_{COG} - x_{PP})T\beta_\theta \end{aligned} \quad (\text{Eq 5-3})$$

Deduction of aerodynamics torques

The empiric laws for the aerodynamics forces in body frame are:

$$\begin{aligned} F_{Aerox} &= -\frac{1}{2}\rho V^2 S_R C_{X(\alpha, M)} \\ F_{Aeroy} &= -\frac{1}{2}\rho V^2 S_R C_{N(\alpha, M)} y_{comp} \\ F_{Aeroz} &= -\frac{1}{2}\rho V^2 S_R C_{N(\alpha, M)} z_{comp} \end{aligned}$$

Being:

ρ = air density (function of the altitude)

V = modulus of the relative velocity (taking into account wind effects)

α = angle of attack

M = Mach number

S_R = reference surface (roughly the surface of the transversal area)

C_X = axial coefficient (function of the angle of attack and the Mach number)

C_N = normal coefficient (function of the angle of attack and the Mach number)

$$y_{comp} = \frac{v_y}{\sqrt{v_y^2 + v_z^2}} = \frac{v_y}{v_T}$$

$$z_{comp} = \frac{v_z}{\sqrt{v_y^2 + v_z^2}} = \frac{v_z}{v_T}$$

The minus sign is justified because the aerodynamics forces in a gravity turn trajectory are drag forces, i.e. in opposite direction that the velocity of the LV.

The y_{comp} , z_{comp} are simply the normalization of the lateral velocities with regard the modulus of the total transversal velocities.

As said the aerodynamic coefficients depends on the angle of attack on each plane and the Mach number and are based in empirical measures (wind tunnel tests). However, in a first approximation the dependency can be expressed as product of a constant C_N (value of the $C(\alpha, M)$ coefficient at that Mach and angle) by the angle of attack.

$$C_N(\alpha, M) = C_N \alpha$$

The dynamic pressure is defined as:

$$p_{dyn} = \frac{1}{2}\rho V^2$$

With these simplifications we have:

$$F_{Aerox} = -P_{dyn} S_R C_{X(\alpha, M)}$$

$$F_{Aero y} = -P_{dyn} S_R C_N \alpha_g \frac{v_y}{v_T}$$

$$F_{Aero z} = -P_{dyn} S_R C_N \alpha_\psi \frac{v_z}{v_T}$$

The general expression for the aerodynamic torques is:

$$M_{Aero} = (r_{CP} - r_{COG}) \otimes F_{Aero}$$

Because the aerodynamics forces acts on the *CP* (center of pressures) and creates a torque that rotates the LV around the *COG*.

Let's name

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} x_{CP} - x_{COG} \\ y_{CP} - y_{COG} \\ z_{CP} - z_{COG} \end{bmatrix}$$

Making the vector product and taken only moments around y and z, we have:

$$M_y = -(r_x F_z - r_z F_x) = -r_x \left(-P_{dyn} S_R C_N \alpha_\psi \frac{v_z}{v_T} \right) + r_z \left(-P_{dyn} S_R C_X \right)$$

$$M_z = r_x F_y - r_y F_x = r_x \left(-P_{dyn} S_R C_N \alpha_g \frac{v_y}{v_T} \right) - r_y \left(-P_{dyn} S_R C_X \right)$$

Again, due the long shape of the LV the distance between the X coordinates of the *r_{COG}* and the *r_{PP}* and symmetry on planes perpendicular to X axis we take:

$$(r_x \gg r_y) \text{ and } (r_x \gg r_z)$$

This allows neglecting the second terms in *r_y* and *r_z* and the simplification of the equations as follows:

$$\begin{aligned} M_y &= (x_{CP} - x_{COG}) \left(P_{dyn} S_R C_N \alpha_\psi \frac{v_z}{v_T} \right) \\ M_z &= (x_{CP} - x_{COG}) \left(-P_{dyn} S_R C_N \alpha_g \frac{v_y}{v_T} \right) \end{aligned} \quad (\text{Eq 5-4})$$

Putting all together

Coming back to the Euler equation (Eq 5-1):

$$\begin{bmatrix} M_{Propx} + M_{Aerox} \\ M_{ProPy} + M_{AeroY} \\ M_{Propz} + M_{Aeroz} \end{bmatrix} = \begin{bmatrix} I_{xx} \frac{d}{dt} \omega_x \\ I_{yy} \frac{d}{dt} \omega_y \\ I_{zz} \frac{d}{dt} \omega_z \end{bmatrix} + \begin{bmatrix} (I_{zz} - I_{yy}) \omega_y \omega_z \\ (I_{xx} - I_{zz}) \omega_x \omega_z \\ (I_{yy} - I_{xx}) \omega_x \omega_y \end{bmatrix}$$

And naming:

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

We have

$$\begin{bmatrix} M_{Propx} + M_{Aerox} \\ M_{ProPy} + M_{AeroY} \\ M_{Propz} + M_{Aeroz} \end{bmatrix} = \begin{bmatrix} I_{xx} \dot{p} \\ I_{yy} \dot{q} \\ I_{zz} \dot{r} \end{bmatrix} + \begin{bmatrix} (I_{zz} - I_{yy})qr \\ (I_{xx} - I_{zz})pr \\ (I_{yy} - I_{xx})pq \end{bmatrix}$$

We use now the previously deduced propulsion torque (Eq 5-3) :

$$\begin{aligned} M_y &= -(x_{COG} - x_{PP})I\beta_\psi \\ M_z &= -(x_{COG} - x_{PP})I\beta_\theta \end{aligned}$$

And the aerodynamics torque (Eq 5-4):

$$\begin{aligned} M_y &= (x_{CP} - x_{COG}) \left(p_{dyn} S_R C_N \alpha_\psi \frac{v_z}{v_T} \right) \\ M_z &= (x_{CP} - x_{COG}) \left(-p_{dyn} S_R C_N \alpha_\theta \frac{v_y}{v_T} \right) \end{aligned}$$

By substitution of all previous expressions into the Euler equation:

$$\begin{aligned} &(-(x_{COG} - x_{PP})I\beta_\psi) + \left[(x_{CP} - x_{COG}) \left(p_{dyn} S_R C_N \alpha_\psi \frac{v_z}{v_T} \right) \right] = I_{yy} \dot{q} + (I_{xx} - I_{zz})pr \\ &(-(x_{COG} - x_{PP})I\beta_\theta) + \left[(x_{CP} - x_{COG}) \left(-p_{dyn} S_R C_N \alpha_\theta \frac{v_y}{v_T} \right) \right] = I_{zz} \dot{r} + (I_{yy} - I_{xx})pq \end{aligned}$$

We arrive to:

$$\begin{aligned} & \left(-(x_{COG} - x_{PP})T\beta_\psi \right) + \left[(x_{CP} - x_{COG}) \left(p_{dyn} S_R C_N \alpha_\psi \frac{v_z}{v_T} \right) \right] - (I_{xx} - I_{zz})pr = I_{yy}\dot{q} \\ & \left(-(x_{COG} - x_{PP})T\beta_g \right) + \left[(x_{CP} - x_{COG}) \left(-p_{dyn} S_R C_N \alpha_g \frac{v_y}{v_T} \right) \right] - (I_{yy} - I_{xx})pq = I_{zz}\dot{r} \end{aligned}$$

We want to express the previous equation only in variables on the guidance reference frame removing the dependence on the angles of attack on body frame. The angle of attack expressed on inertial variables is given by the following formula (see demonstration in appendix 8.4.3) where θ_G and ψ_G are the orientation of the body frame in the inertial frame:

$$\begin{aligned} \alpha_g &= -\theta_G + \frac{\dot{Y}}{V} \\ \alpha_\psi &= \psi_G + \frac{\dot{Z}}{V} \end{aligned}$$

Substituting in the previous equation:

$$\begin{aligned} I_{yy}\dot{q} &= \left(-(x_{COG} - x_{PP})T\beta_\psi \right) \\ &+ \left[(x_{CP} - x_{COG}) \left(p_{dyn} S_R C_N \frac{v_z}{v_T} \right) \right] \psi \\ &+ \left[(x_{CP} - x_{COG}) \left(p_{dyn} S_R C_N \frac{v_z}{v_T} \right) \right] \frac{\dot{Z}}{V} - (I_{xx} - I_{zz})pr \end{aligned}$$

$$\begin{aligned} I_{zz}\dot{r} &= \left(-(x_{COG} - x_{PP})T\beta_g \right) \\ &+ \left[(x_{CP} - x_{COG}) \left(-p_{dyn} S_R C_N \frac{v_y}{v_T} \right) \right] (-\theta) \\ &+ \left[(x_{CP} - x_{COG}) \left(-p_{dyn} S_R C_N \frac{v_y}{v_T} \right) \right] \left(+\frac{\dot{Y}}{V} \right) - (I_{yy} - I_{xx})pq \end{aligned}$$

The following simplifications are usually taken:

- If v_y and v_z are of similar magnitude, we can say (with an error of 1 over square root of 2) that

$$v_y \approx v_z \rightarrow \frac{v_y}{\sqrt{v_y^2 + v_z^2}} = \frac{v_y}{\sqrt{2v_y^2}} \approx \frac{1}{\sqrt{2}} \approx 1$$

- By symmetry of the LV, the transversal inertia moments are similar. We can substitute I_{YY} and I_{ZZ} by I_T

$$I_{yy} \approx I_{zz}, I_{yy} = I_T, I_{zz} = I_T$$

- Assuming that the roll rate is constant
 $p \approx p_0$

So we have:

$$I_T \dot{q} = -(x_{COG} - x_{PP}) I \beta_\psi + [(x_{CP} - x_{COG}) (p_{dyn} S_R C_N)] \psi + [(x_{CP} - x_{COG}) (p_{dyn} S_R C_N)] \frac{\dot{Z}}{V} + (I_T - I_{xx}) p r$$

$$I_T \dot{r} = -(x_{COG} - x_{PP}) I \beta_\vartheta + [(x_{CP} - x_{COG}) (-p_{dyn} S_R C_N)] (-\vartheta) + [(x_{CP} - x_{COG}) (-p_{dyn} S_R C_N)] \left(+ \frac{\dot{Y}}{V} \right) - (I_T - I_{xx}) p q$$

The following definitions are commonly used in the aerospace field:

$$A_6 = \frac{1}{I_T} (x_{CP} - x_{COG}) (p_{dyn} S_R C_N)$$

$$K_1 = -\frac{T}{I_T} (x_{COG} - x_{PP})$$

$$\lambda = \frac{I_T - I_{xx}}{I_T} = 1 - \frac{I_{xx}}{I_T}$$

A_6 is the stability parameter. It is defined as positive. K_1 is the controllability parameter.

Finally we obtain a more simplified expression of the Euler rotational equation that relates the angular acceleration in function of the current angular rates, the LV attitude (aerodynamics) and the actuator positions:

$$\begin{aligned} \dot{q} &= (K_1 \beta_\psi) + A_6 \psi + \left(\frac{A_6}{V} \right) \dot{Z} + \lambda p_0 r \\ \dot{r} &= (K_1 \beta_\vartheta) + A_6 \vartheta - \left(\frac{A_6}{V} \right) \dot{Y} - \lambda p_0 q \end{aligned} \tag{Eq 5-5}$$

Note: by symmetry of the Launcher, we know that under a positive turn of 90 deg on the roll axis, the dynamics shall be identical (i.e. the selection of the Y and Z axis is just a convention). This symmetry shall be observed on the rotational equation. Refer to appendix 8.4.5 for such demonstration.

5.3.4.2 Solution to the Translational equation

We want to find the expression of the translational formula in the Guidance frame, i.e. we want to know how sees the rocket an inertial observer knowing the forces in body frame.

For small deviations from the steady state and small angles this Guidance frame is inertial (in Y and Z axis) and aligned with the Trajectory reference frame. In the next instant the Guidance frame follows the body frame in rotation around X (roll).

We start from the equation (Eq 5-2) expressed as force instead of acceleration. It is assumed a constant (or slow varying) mass.

$$F_I = m \frac{d^{2B}r}{dt} + 2m\omega \otimes \frac{d^B r}{dt} + m \frac{d^B \omega}{dt} \otimes r_B + m(\omega \otimes (\omega \otimes r_B))$$

The following simplifications are also assumed:

The angular velocity of the Body frame with regard the Guidance (inertial) frame is constant, so its time derivative is zero. This made the third term of previous formula zero.

The centrifugal force does not play any role, as:

- In the roll axis, it is compensated by the structure of the launcher (i.e. the LV preserves its integrity)
- In the pitch and yaw axis, the angular velocity are negligible against the roll axis

With these simplifications we have arrived to:

$$F_I = m \frac{d^{2B}r}{dt} + 2m\omega \otimes \frac{d^B r}{dt}$$

Developing the vector product we have

$$2m\omega \otimes \frac{d^B r}{dt} = 2m \begin{bmatrix} i & j & k \\ p & q & r \\ v_x & v_y & v_z \end{bmatrix} = 2m \begin{bmatrix} qv_z - rv_y \\ -pv_z + rv_x \\ pv_y - qv_x \end{bmatrix}$$

And we obtain the following equation for the forces in inertial frame

$$F_I = m \begin{bmatrix} \dot{v}_x + 2(qv_z - rv_y) \\ \dot{v}_y - 2(pv_z - rv_x) \\ \dot{v}_z + 2(pv_y - qv_x) \end{bmatrix} \quad (\text{Eq 5-6})$$

Expression of the forces

First, we have to find the expression of the forces in body frame. We start with the propulsive forces:

$$\begin{aligned} F_{propX} &= T \cos(\beta_\psi) \cos(\beta_g) \\ F_{propY} &= T \cos(\beta_\psi) \sin(\beta_g) \\ F_{propZ} &= -T \sin(\beta_\psi) \cos(\beta_g) \end{aligned}$$

We should ignore the F_{propX} as the thrust on axis X is not to be controlled. Assuming that we have small actuation angles we can approximate:

$$\begin{aligned} \cos(\beta_\psi) &\approx 1, \quad \cos(\beta_g) \approx 1 \\ \sin(\beta_\psi) &\approx \beta_\psi, \quad \sin(\beta_g) \approx \beta_g \end{aligned}$$

$$\begin{aligned} F_{propY} &= T(1) \sin(\beta_g) = T\beta_g \\ F_{propZ} &= -T(1) \sin(\beta_\psi) \cos(\beta_g) \approx -T\beta_\psi \end{aligned}$$

We continue with the aerodynamics forces with the same simplifications taken in the rotational dynamics:

- If v_y and v_z are of similar magnitude, we can say (with an error of $\sqrt{2}$) that

$$v_y \approx v_z \rightarrow \frac{v_y}{\sqrt{v_y^2 + v_z^2}} = \frac{v_y}{\sqrt{2v_y^2}} \approx \frac{1}{\sqrt{2}} \approx 1$$

Arriving to:

$$\begin{aligned} F_{AeroX} &= -p_{dyn} S_R C_{X(\alpha, M)} \\ F_{AeroY} &= -p_{dyn} S_R C_N \alpha_g \\ F_{AeroZ} &= -p_{dyn} S_R C_N \alpha_\psi \end{aligned}$$

In order to express the previous forces in inertial frame we need to express the body variables in inertial variables. The body variables are the angles of attack that can be expressed in Guidance frame variables as follows (see demonstration in appendix 8.4.4)

$$\begin{aligned} \alpha_g &= -\theta_G + \frac{\dot{Y}}{V} \\ \alpha_\psi &= \psi_G + \frac{\dot{Z}}{V} \end{aligned}$$

We arrive to the expressions for the forces:

$$\begin{aligned}
 F_{propy} + F_{Aeroz} &= (T\beta_g) + (-p_{dyn}S_R C_N)(-g) + (-p_{dyn}S_R C_N) \left(\frac{\dot{Y}}{V} \right) \\
 F_{propz} + F_{Aeroz} &= (-T\beta_\psi) + (-p_{dyn}S_R C_N)\mu + (-p_{dyn}S_R C_N) \left(\frac{\dot{Z}}{V} \right)
 \end{aligned}
 \tag{Eq 5-7}$$

Expression of the accelerations

We start from the equation (Eq 5-6) deduced previously:

$$F_I = m \begin{bmatrix} \dot{v}_x + 2(qv_z - rv_y) \\ \dot{v}_y - 2(pv_z - rv_x) \\ \dot{v}_z + 2(pv_y - qv_x) \end{bmatrix}$$

Now let's express the velocities in body (v_y, v_z) in function of variables in the (inertial) guidance frame.

For XY axis we have that the angle of the total velocity vector V with regard the X axis in guidance frame is the sum of the angle theta of the body frame with regard the guidance frame plus the angle of the vector V in the body frame. We have an analogous relation for the XZ axis. See demonstration on appendix 8.4.4.

$$\begin{aligned}
 v_y &= -g_G V + \dot{Y} \\
 v_z &= \psi_G V + \dot{Z}
 \end{aligned}$$

We can derive the previous expression 'normally' as the variables are all inertial.

$$\begin{aligned}
 \dot{v}_y &= -\dot{g}_G V - g_G \dot{V} + \ddot{Y} \\
 \dot{v}_z &= +\dot{\psi}_G V + \psi_G \dot{V} + \ddot{Z}
 \end{aligned}$$

We want to derive now a formula for expressing the angular velocity of the body frame in function of inertial angles in guidance. The expression is (refer to demonstration in 8.4.3):

$$\begin{aligned}
 q &= \dot{\psi}_G - p g_G \\
 r &= \dot{g}_G + p \psi_G
 \end{aligned}$$

Let's substitute in F_y and F_z the variables in body frame

$$\begin{bmatrix} F_Y \\ F_Z \end{bmatrix} = m \begin{bmatrix} \dot{v}_y - 2pv_z + 2rv_x \\ \dot{v}_z + 2(pv_y - qv_x) \end{bmatrix}$$

By their equivalences in guidance frame (we remove from here all the explicit reference to the sub-index that indicates guidance frame):

$$\begin{aligned} v_y &= -\mathcal{G}V + \dot{Y} & \dot{v}_y &= -\dot{\mathcal{G}}V - \mathcal{G}\dot{V} + \ddot{Y} \\ v_z &= \psi V + \dot{Z} & \dot{v}_z &= +\dot{\psi}V + \psi\dot{V} + \ddot{Z} \\ q &= \dot{\psi} - p\mathcal{G} \\ r &= \dot{\mathcal{G}} + p\psi \end{aligned}$$

Also, due to the physics of the problem, V is similar to v_x and much greater than v_y and v_z (almost all the velocity of the rocket is along the X axis and the angles are small)

We have:

$$\begin{bmatrix} F_Y \\ F_Z \end{bmatrix} = m \begin{bmatrix} (-\dot{\mathcal{G}}V - \mathcal{G}\dot{V} + \ddot{Y}) - (2p)(+\psi V + \dot{Z}) + 2(\dot{\mathcal{G}} + p\psi)V \\ (+\dot{\psi}V + \psi\dot{V} + \ddot{Z}) + 2p(-\mathcal{G}V + \dot{Y}) - 2(\dot{\psi} - p\mathcal{G})V \end{bmatrix}$$

Developing and grouping terms:

$$\begin{aligned} \begin{bmatrix} F_Y \\ F_Z \end{bmatrix} &= m \begin{bmatrix} (-\dot{\mathcal{G}}V - \mathcal{G}\dot{V} + \ddot{Y}) - 2p\psi V - 2p\dot{Z} + 2\dot{\mathcal{G}}V + 2p\psi V \\ (+\dot{\psi}V + \psi\dot{V} + \ddot{Z}) - 2p\mathcal{G}V + 2p\dot{Y} - 2\dot{\psi}V + 2p\mathcal{G}V \end{bmatrix} \\ \begin{bmatrix} F_Y \\ F_Z \end{bmatrix} &= m \begin{bmatrix} (\dot{\mathcal{G}}V - \mathcal{G}\dot{V} + \ddot{Y}) - 2p\dot{Z} \\ (-\dot{\psi}V + \psi\dot{V} + \ddot{Z}) + 2p\dot{Y} \end{bmatrix} \end{aligned}$$

Also we know that the acceleration in x body axis is similar to \dot{V} and is given by the expression:

$$\dot{v}_x = \frac{T - D}{m} = \frac{T - p_{dyn}S_R C_X}{m}$$

Replacing in previous:

$$\begin{bmatrix} F_Y \\ F_Z \end{bmatrix} = m \begin{bmatrix} \left(\dot{\mathcal{G}}V - \mathcal{G} \left(\frac{T - p_{dyn}S_R C_X}{m} \right) + \ddot{Y} \right) - 2p\dot{Z} \\ \left(-\dot{\psi}V + \psi \left(\frac{T - p_{dyn}S_R C_X}{m} \right) + \ddot{Z} \right) + 2p\dot{Y} \end{bmatrix}$$

And restoring in the equation the angular rates (p, q, r)

$$\begin{bmatrix} F_Y \\ F_Z \end{bmatrix} = m \begin{bmatrix} \left((r - p\psi)V - \mathcal{G} \left(\frac{T - p_{dyn}S_R C_X}{m} \right) + \ddot{Y} \right) - 2p\dot{Z} \\ \left(-(q + p\mathcal{G})V + \psi \left(\frac{T - p_{dyn}S_R C_X}{m} \right) + \ddot{Z} \right) + 2p\dot{Y} \end{bmatrix} \quad (\text{Eq 5-8})$$

Putting all together

Now let's put together the expressions for the forces (Eq 5-7) and the expressions for the accelerations (Eq 5-8):

$$\begin{aligned} (T\beta_g) + (-p_{dyn}S_R C_N)(-\mathcal{G}) + (-p_{dyn}S_R C_N) \left(+ \frac{\dot{Y}}{V} \right) &= m \left(\ddot{Y} - 2p\dot{Z} - \mathcal{G} \left(\frac{T - p_{dyn}S_R C_X}{m} \right) + V(r - p\psi) \right) \\ (-T\beta_\psi) + (-p_{dyn}S_R C_N)\psi + (-p_{dyn}S_R C_N) \left(\frac{\dot{Z}}{V} \right) &= m \left(\ddot{Z} + 2p\dot{Y} + \psi \left(\frac{T - p_{dyn}S_R C_X}{m} \right) - V(q + p\mathcal{G}) \right) \end{aligned}$$

And then reordering

$$\begin{aligned} \ddot{Y} &= \frac{T}{m}\beta_g + \frac{1}{m}(p_{dyn}S_R C_N)\mathcal{G} - \frac{1}{m}(p_{dyn}S_R C_N)\frac{\dot{Y}}{V} + 2p\dot{Z} + \left(\frac{T - p_{dyn}S_R C_X}{m} \right)\mathcal{G} - V(r - p\psi) \\ \ddot{Z} &= -\frac{T}{m}\beta_\psi - \frac{1}{m}(p_{dyn}S_R C_N)\psi - \frac{1}{m}(p_{dyn}S_R C_N)\frac{\dot{Z}}{V} - 2p\dot{Y} - \left(\frac{T - p_{dyn}S_R C_X}{m} \right)\psi + V(q + p\mathcal{G}) \\ \ddot{Y} &= \frac{T}{m}\beta_g + \frac{1}{m}(T - p_{dyn}S_R C_X + p_{dyn}S_R C_N)\mathcal{G} - \frac{1}{m}(p_{dyn}S_R C_N)\frac{\dot{Y}}{V} + 2p\dot{Z} - V(r - p\psi) \\ \ddot{Z} &= -\frac{T}{m}\beta_\psi - \frac{1}{m}(T - p_{dyn}S_R C_X - p_{dyn}S_R C_N)\psi - \frac{1}{m}(p_{dyn}S_R C_N)\frac{\dot{Z}}{V} - 2p\dot{Y} + V(q + p\mathcal{G}) \end{aligned}$$

Grouping terms we arrive to the final expression for the lateral accelerations in function of the actuator angles, of the angles of the velocity vector in guidance frame and of the lateral velocities:

$$\begin{aligned} \ddot{Y} &= \frac{T}{m}\beta_g + \frac{T - p_{dyn}S_R(C_N - C_X)}{m}\mathcal{G} - \frac{p_{dyn}S_R C_N}{mV}\dot{Y} + 2p\dot{Z} - Vr + Vp\psi \\ \ddot{Z} &= -\frac{T}{m}\beta_\psi - \frac{T - p_{dyn}S_R(C_N - C_X)}{m}\psi - \frac{p_{dyn}S_R C_N}{mV}\dot{Z} - 2p\dot{Y} + Vq + Vp\mathcal{G} \end{aligned} \tag{Eq 5-9}$$

Note: by symmetry of the Launcher, we know that under a positive turn of 90 deg on the roll axis, the dynamics shall be identical (i.e. the selection of the Y and Z axis is just a convention). This symmetry shall be observed in the translational equation. Refer to appendix 8.4.5 for such demonstration.

5.3.5 State space model of the LV

The linear state space model chosen for the open loop plant is the following:

States:

$$\vec{x} = [\dot{y} \quad \mathcal{G} \quad r \quad \dot{z} \quad \psi \quad q]$$

Measured outputs:

$$\vec{y} = [\mathcal{G} \quad \psi]$$

Control inputs:

$$\vec{u} = [\beta_{\mathcal{G}} \quad \beta_{\psi}]$$

Being the state space representation:

$$A = \begin{bmatrix} -A_{vlat} & A_{theta} & -V & 2p_0 & Vp_0 & 0 \\ 0 & 0 & 1 & 0 & -p_0 & 0 \\ -\frac{A_6}{V} & A_6 & 0 & 0 & 0 & -\lambda p_0 \\ -2p_0 & Vp_0 & 0 & -A_{vlat} & -A_{psi} & V \\ 0 & p_0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \lambda p_0 & \frac{A_6}{V} & A_6 & 0 \end{bmatrix}, B = \begin{bmatrix} \frac{T}{m} & 0 \\ 0 & 0 \\ K_1 & 0 \\ 0 & -\frac{T}{m} \\ 0 & 0 \\ 0 & K_1 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$A_{theta} = A_{psi} = \frac{T - p_{dyn} S_{ref} (C_N - C_x)}{m}, \quad A_{vlat} = \frac{p_{dyn} S_{ref} C_N}{mV}$$

The choice of measures and states is justified by the requirements. The more important requirements during the first stage are:

- Maximum “ $p_{dyn} * \alpha$ ” (according to a gabarit on Mach number)
- Lateral deviations: first flight < 500 m, second flight < 1000 m
- Lateral velocity errors: first flight < 15 m/s, second flight < 35 m/s
- Transversal rate at stage separation (not applicable in the paper as simulation is performed before separation)
- TVC actuator: max deflection 4.9 deg, max deflection rate 10 deg/s

The more important requirement is that the load “ $p_{dyn} * \alpha$ ” cannot surpass a gabarit because the structural integrity of the vehicle would be compromised. As p_{dyn} cannot be controlled (because depends on the aerodynamics and longitudinal velocity that cannot be controlled) the requirement is imposed on the angle of attack.

However due to a variety of technological reasons VEGA has no angle of attack sensors. The angle of attack can be expressed in inertial components as done previously. This justifies the choice of θ and ψ as part of the state and as the measured outputs.

The second and third requirement on lateral deviations and lateral velocity errors justify the choosing of the lateral deviations and velocities as states. Note that during the atmospheric flight this second requirement is of lower importance with regard the one in angle of attack. We do not impose a tight control of these and we will not include as measured outputs. This choice will be justified by the results obtained later.

The choice of the actuators is obvious: the deflection of the nozzle.

We add delays to the 6DoF plant. These delays are due to digital processing by the IMU (10 ms), the OBC (12 ms) and TVC (15 ms). The delays are modeled with a Padé transform of second order at plant input.

5.3.5.1 Study of the open loop plant in the frequency domain

Taken the Laplace transform in the rotational equation (Eq 5-5), neglecting the translational dynamics and setting a zero roll rate we have:

$$\begin{aligned}\dot{q} &= (K_1 \beta_\psi) + A_6 \psi \\ \dot{r} &= (K_1 \beta_g) + A_6 \mathcal{G}\end{aligned}$$

The Laplace transform is:

$$\begin{aligned}\frac{\psi}{\beta_\psi} &= \frac{K_1}{s^2 - A_6} \\ \frac{\mathcal{G}}{\beta_g} &= \frac{K_1}{s^2 - A_6}\end{aligned}$$

The transfer function has a double pole in $s = \pm\sqrt{A_6}$. This means that the Launcher in open loop is not stable. It is needed to describe the evolution of the parameters K_1 and A_6 during the atmospheric phase.

At low velocities A_6 is very small. It increases very fast with the square of the velocity but start decreasing after a given altitude due to low air density. K_1 is increases slowly during the first phase of the flight while the thrust is the dominant effect (because thrust is almost constant but the distance between the X_{PP} and X_{COG} increases as propellant is being consumed). At the end of the stage the thrust descent very fast and so does K_1 . See (Figure 5-18).

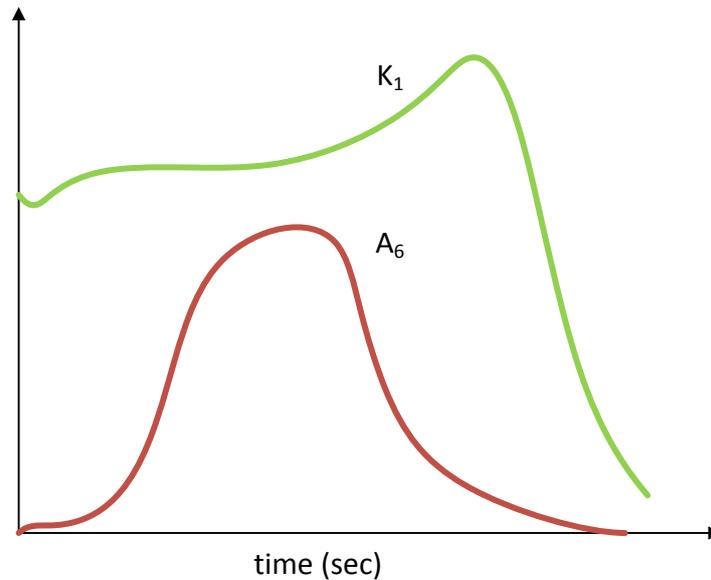


Figure 5-18: Evolution of K_1 and A_6 along the first stage

In order to be able to control the Launcher the torque provided by the TVC when deflecting the nozzle (related to K_1) shall be always greater than the aerodynamic torque (related to A_6). If the previous is not true the vehicle becomes unstable. The more dangerous region is the region at about 55-60 seconds after launch (called the “maximum dynamic pressure” region) where the margin between the torques is dangerously small.

The influence of the roll rate in the stability margin has been deeply studied in (Cruciani, 2008). Basically there is a double effect. First effect is direct, due to the coupling introduced by the term “ λ_{por} ”. Second effect is indirect due to the feedback lack. The TVC angle commanded will be computed with some control law based on the measures taken in inertial frame by the IMU that shall be transformed to the Guidance frame. Expressed in Guidance frame these measures depend on the roll angle.

5.3.5.2 Study of the model uncertainty for the plant

The sensitivity of the open loop plant with regard variations of parameters as *mass*, *thrust*, X_{COG} , p_{Dyn} is not strong. We plot in (Figure 5-19) a set of plants where the previous parameters have been scattered by a 5%.

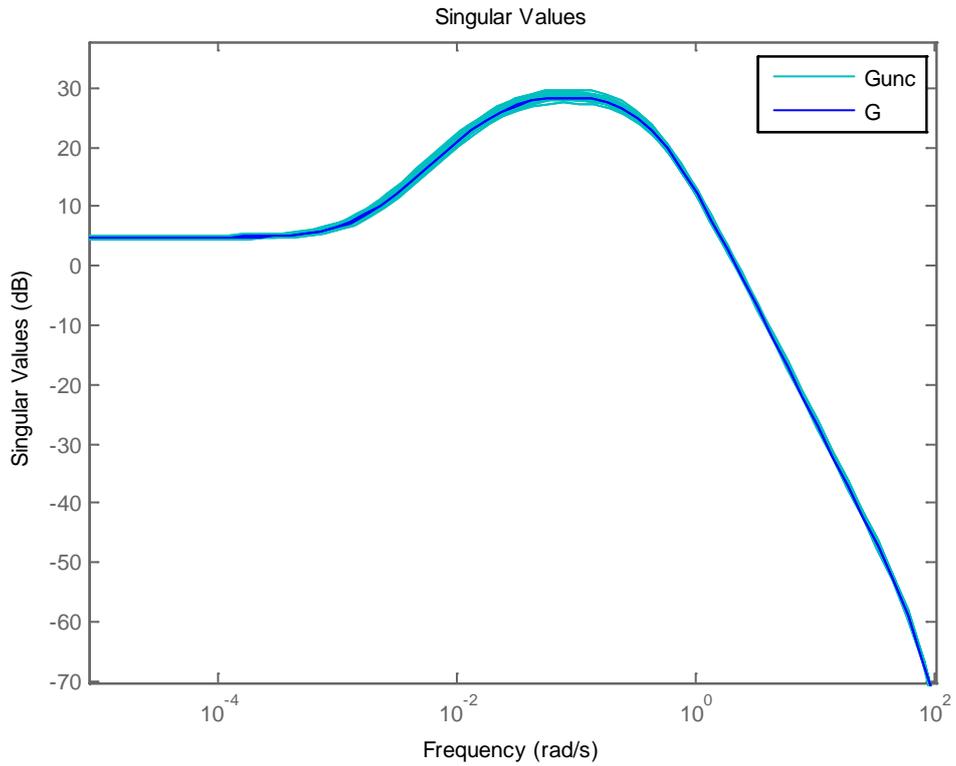


Figure 5-19: Scattering of open loop (except roll)

The sensitivity is much higher against variations in the roll rate.

The (Figure 5-20) shows the sigma values for a roll rate of 0, +5 and +45 deg/s. Basically we are introducing coupling between the pitch and yaw channels. In the sigma diagram this is manifested as a big difference between the major and minor singular values at frequencies at around 1-3 Hz. This is the coupling. Under roll rate the plant becomes ill conditioned.

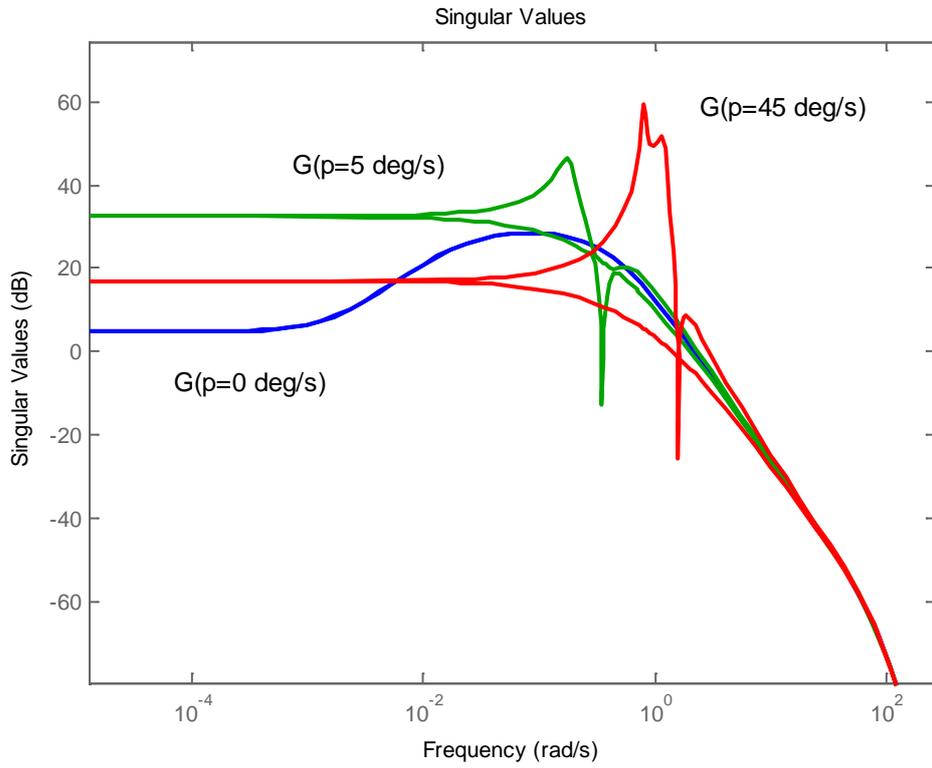


Figure 5-20: Influence of roll rate in open loop plant

The (Figure: 5-21) shows the sigma values of a set of uncertainty plants in a range -45 to +45 deg/s.

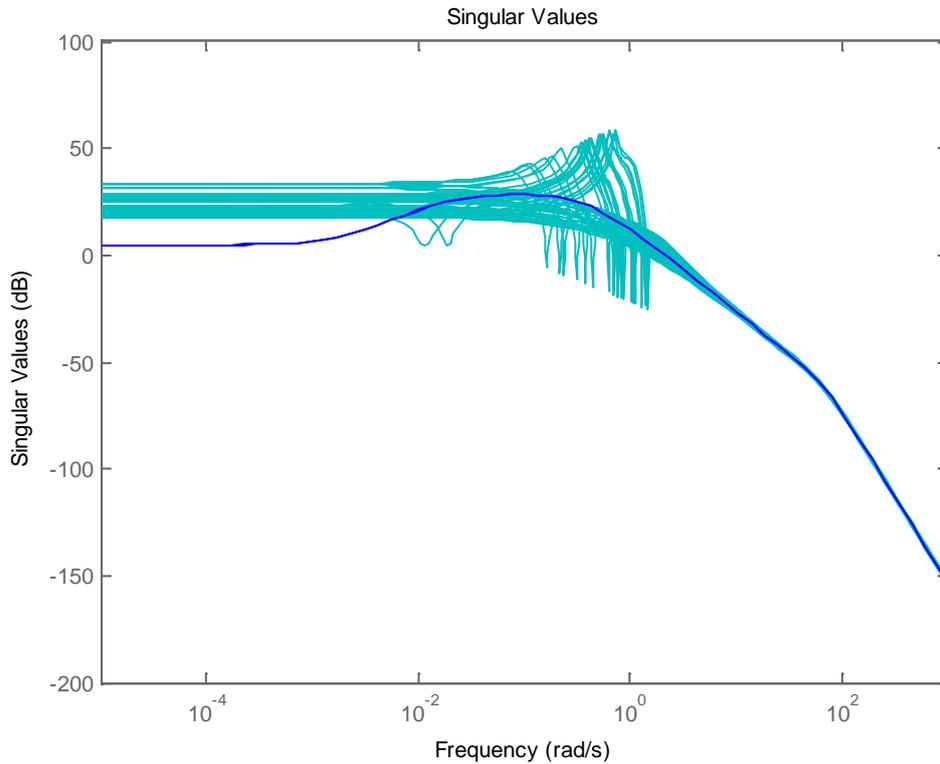


Figure: 5-21: Sigma values variation with roll rate

5.4 miniVEGA: a non Linear Simulator for the VEGA Launcher

In any control problem is fundamental to have a good model of the plant. This chapter explains the design and synthesis of a minimalistic (but 6DoF realistic) model of the VEGA Launcher.

5.4.1 Introduction to the VEGA Launcher simulators

The reference simulator of the VEGA Launcher project is VEGAMATH. This is an advanced and fully representative simulator of the VEGA Launch Vehicle. The representativity of the simulator has been demonstrated by the success of the VEGA first and second flights and the good according between the simulated and real flight found in the post flight analysis. VEGAMATH is written with Simulink.

VEGAMATH is a proprietary product and technical details cannot be disclosed. We will just enumerate the features of the simulator:

- Simulation of the four LV stages with one or two payloads
- Simulation of the stage ignitions, separations
- Simulation of the 6DoF dynamics
- Propulsive model for the solid stages and liquid engine motor
- RACS detailed model

- Atmospheric model (winds and gusts)
- Gravity model
- Bending modes
- Sloshing modes
- Full capacity to introduce uncertainties and scatterings
- Full representative models of the TVC, nozzle and IMU (with noise, drift and scatterings)
- Ability to execute the GNC algorithms either as prototype on C or the real flight Ada code
- Ability to run Montecarlo simulations

A second reference simulator used on VEGA is VESAT, developed for ESA at ESTEC. VESAT is a work in progress and has been cross-checked with VEGAMATH for the first mission. VESAT is also written with Simulink.

Also VESAT is too complex for using in the scope of our thesis, but will serve as a model to derive a more simplified simulator. We will call it “*miniVEGA*” that stands for “mini simulator for VEGA”.

“*miniVEGA*” contains a number of simplifications with regards VEGAMATH and VESAT. The following are the main simplifications introduced:

- Only first stage is modeled (no lift-off, no separation)
- No bending modes. No sloshing models.
- Simplified support uncertainties and scatterings
- Ideal IMU
- TVC model is a non-linear second order model (with actuation and rate limiters)
- GNC algorithms:
 - o Navigation and guidance are perfect.
 - o Control is implemented by a linear controller
- The simulation is representative only during short times

These simplifications are justified and detailed on the next chapters.

5.4.2 Structure of mini-VEGA

5.4.2.1 Overall structure

The top level of the simulator (Figure: 5-22) follows the typical layout of a feedback system:

- The ‘Launch Vehicle’ subsystem models the Launch Vehicle dynamics behavior
- The ‘TVC’ subsystem models the Thrust Vector Control dynamics
- The ‘GNC’ subsystem implements the GNC algorithms
- The ‘Measurement’ subsystem models the measurement instruments
- The ‘Winds’ subsystem models the external winds

- Auxiliary subsystems as 'Figures', 'Logging'

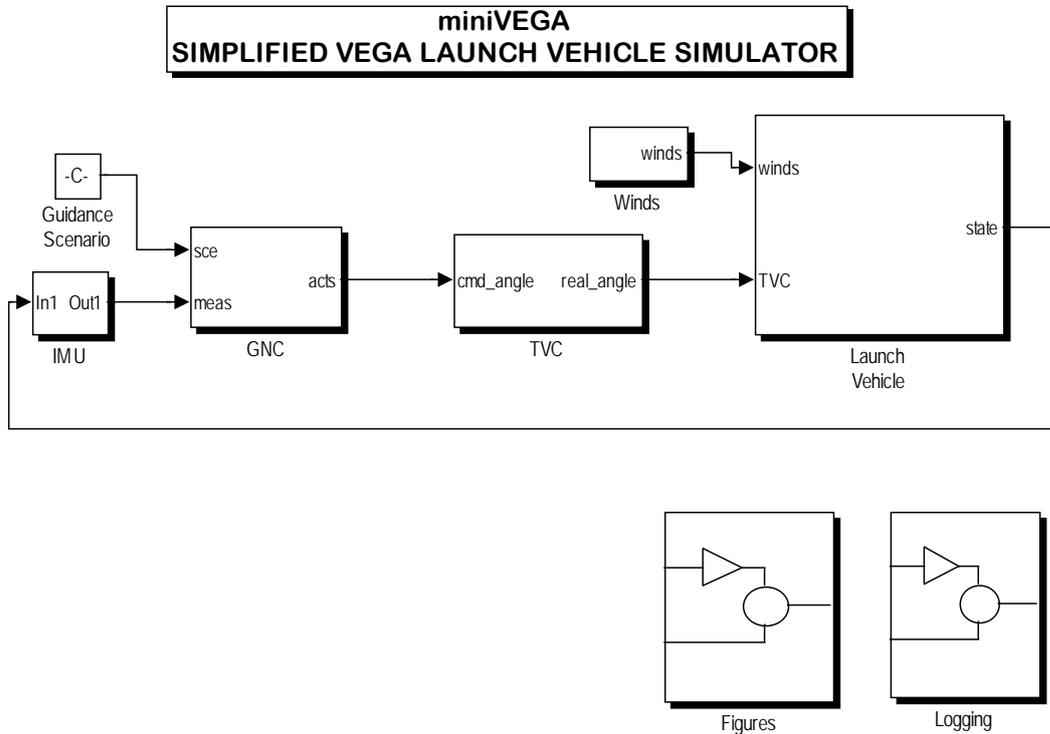


Figure: 5-22: Top level of miniVEGA

5.4.2.2 The Measurement subsystem

In the real LV, the Measurement subsystem is implemented by the IMU (Inertial Measurement Unit). This Unit includes laser gyro meters and accelerometers that sense the forces and torques acting on the LV. The IMU embeds an ERC32 processor that executes the algorithms to transform these sensed forces and torques measured on IMU body frame to acceleration and velocities measures on the IMU inertial reference system. The OBC computer requests this information from the IMU at 40Hz.

VEGAMATH includes a full model of the IMU that has been cross-compared with the real equipment. This model elaborates the accelerations and velocities (linear and angular) based on the forces sensed by the IMU sensors. Effects as misalignments, noise, drifts, etc... are modeled. The operative modes of the IMU (alignment, flight mode) are also modeled.

VESAT includes a simplified model of the IMU. Only the flight mode is implemented.

miniVEGA implements an ideal IMU model. The measurements are assumed to be taken at the COG and to be perfect: i.e. the IMU model just receives the accelerations and velocities of the 6DoF model and propagates them.

A delay block models the time that takes the computation on the real IMU that is 10 ms in the worst case.

5.4.2.3 The GNC subsystem

We will provide first an overview of the real GNC algorithms. Then we will comment the simplifications performed in our work.

The GNC subsystem is implemented by the central OBC (On Board Computer). The OBC includes an ERC32 processor and a special chip (COCOS) for management of 1553 communications with the instruments. During the flight the OBC executes the Flight Program Software (FPS) that is in charge of:

- Execute the GNC algorithms
- Receive measurements from the IMU
- Send actuators commands to the TVC and to the RACS
- Flight Management: command the stages separation, PL separation, etc...
- Scheduling
- Telemetry: send TLM to Ground for Safety (location of the LV) and post-flight analysis
- FDIR (Failure Detection Isolation and Recovery)

The FPS is divided on 3 main layers:

- LN1 in charge of management of the OBC HW (i.e. drivers)
- LN2 in charge of scheduling
- LN3 in charge of GNC, FDIR, Telemetry, Flight Management

The LN3 contains the Navigation, Guidance and Control algorithms. The algorithms are implemented in Ada.

The Navigation Algorithm

The Navigation algorithm receives the data of the IMU: current velocities and angular position with regard the initial IMU reference frame (set when the IMU changes to flight mode 2 seconds before lift-off). The navigation algorithms shall compute the current inertial velocities and angular position with regard the ECI (Earth Centered Inertial) reference frame. It shall be taken into account also that the IMU is not mounted in the COG of the LV but in the upper stage and that the IMU has an angle of 45 degrees with regard the LV body frame.

The Guidance Algorithm

The Guidance algorithm during the P80 atmospheric phase has several modes. The first mode is active from lift-off until the Launch Pad is cleared (a few seconds). In this mode

the Guidance commands a vertical flight (the nozzle is kept vertical) to avoid collision with the Launch Pad.

The second mode (pitch over maneuver) and third mode (gravity turn) are described together. The objective of the Guidance on these modes is to achieve a flight with zero angle of incidence. This minimizes the load imposed to the LV structure and maximizes the velocity reached. In absence of wind this is achieved by keeping the nozzle angle close to zero.

However, the objective is not to perform an exact vertical climb. In fact, to get advantage of initial velocity provided by the Earth rotation the trajectory has to be made progressively horizontal to the Earth surface. Both objectives are compatible (kept the nozzle to zero and change the inclination of the trajectory) by using the Earth gravity. The pitch over maneuver deflects slightly the nozzle to achieve some deviation from the vertical. From this instant the gravity made the LV to rotate slowly.

The Control Algorithm

The TVC control algorithm receives the error in the controlled variables and generates the appropriate commands for the TVCs. In the real algorithms the following has to be taken into account:

- The compensation of the pivot point offset due to the SRM pressure
- The conversion from angles in pitch, yaw planes to linear elongations of the EMAs. It shall be taken into account that the EMAs are not aligned on pitch and yaw planes but have an angle of 135 deg
- The scheduling of the control law (gain and filters) during the flight

The RACS control algorithm only limits the roll rate of the LV under a threshold. If the roll does not reach 45 deg/s the thrusters are no activated. If the roll rate exceeds this margin the thrusters are activated until roll is decreased.

The implantation of the GNC into the simulators

VEGAMATH and VESAT are able to execute the complete GNC algorithms but not the complete FPS. It shall be noted that the FPS is designed to run in top of an ERC32 processor and the LN1 and LN2 layers are dependent on that specific HW. These layers cannot run on a standard PC. Instead the code of the GNC algorithms is portable and can be executed in Windows with minor adaptations.

The GNC algorithms are so compiled and called from the simulators using the standard Simulink capabilities to call external code (MEX files).

miniVEGA follows a much more simplified approach:

- Navigation is ideal:
 - o We have full access to all the LV state vector and so the measurements are perfect
 - o The measures are performed at the COG of the LV

Application to a real world application: the VEGA Launcher

- Guidance is simplified:
 - o i.e. basic orders as steps, ramps, given on the LV body frame
- Control is simplified:
 - o A TVC linear controller (in the continuous domain)
 - o The pivot point offset and conversion to EMA elongation is not managed
 - o No RACS control is performed: it is assumed that if roll rate is above 45 deg/s the RACS would act and decrease it below 45 deg/s

As summary the full GNC subsystem is modeled as *miniVEGA* as (Figure 5-23):

- A Simulink 'Controller' block
- A delay block, modeling the time that takes the computation in real OBC (12 ms)

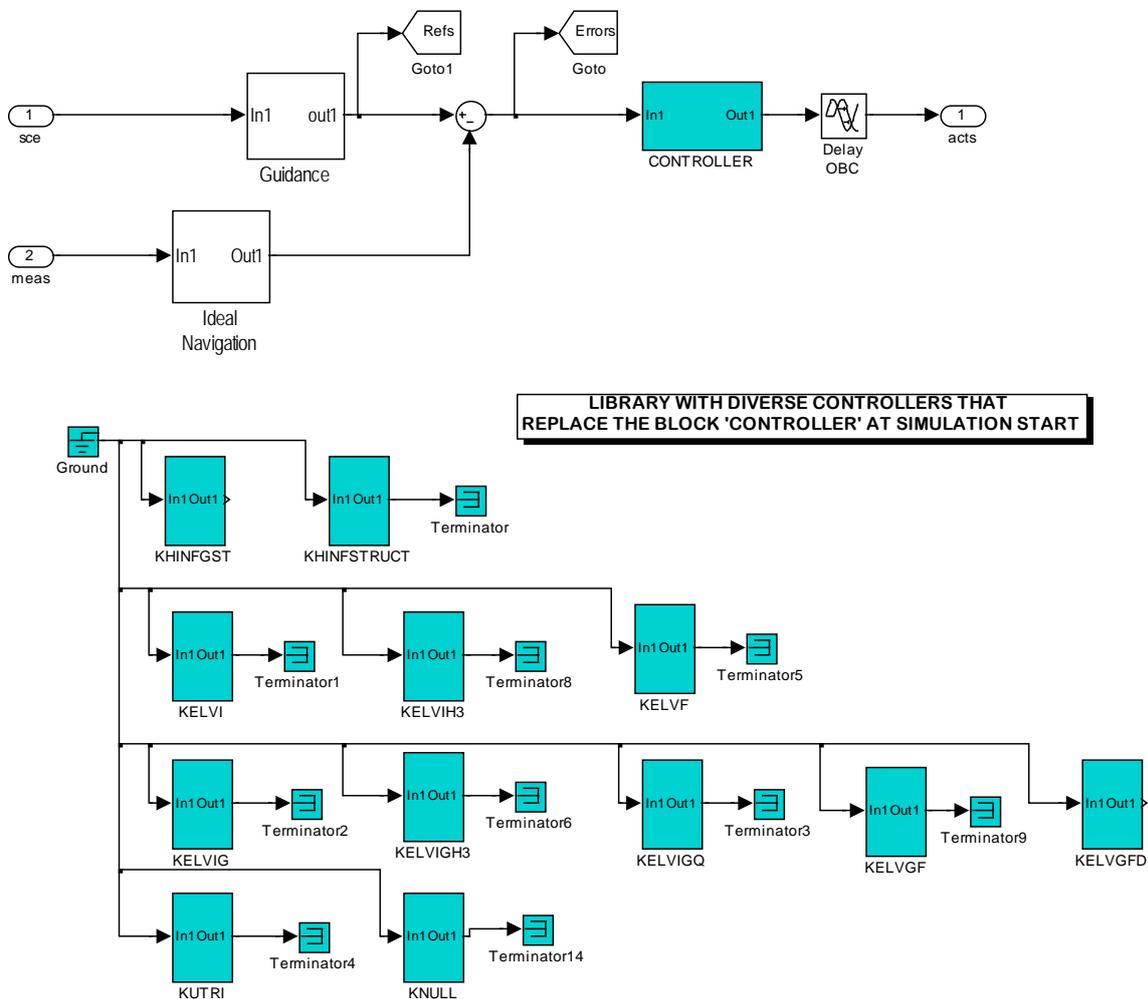


Figure 5-23: Model of the GNC in miniVEGA

The name of the desired controller is passed as parameter to the simulation. Some Matlab code replaces the block 'controller' with the desired block before the start of the simulation.

5.4.2.4 The TVC subsystem

The Thrust Vector Control Subsystem is made of the IPDU (Integrated Power and Dynamics Unit), the EMA (Electro Mechanical Actuator) and batteries, harness, etc. The TVC is a full new development with regard the hydraulic actuators used in Ariane 5. The usage of electromechanical actuators allows a decrease of costs in maintenance and simplifies the preparation of the Launch.

The IPDU includes an specifically designed digital computer that is able to perform 'closed loop control' of the nozzle position with extreme performances (the control loop is made of 3 feedback loops in electrical current, velocity and position, being the frequency of the inner loop (the current loop) of the order of kHz).

VEGAMATH incorporates several TVC models that can be selected depending on the needs of the simulation:

- A high representative model provided by the TVC manufacturer
- A second order function model (with saturations of position and velocity) and delays

VESAT implements only the second order model.

miniVEGA implements also only the nonlinear second order model. The full TVC subsystem is modeled as (Figure 5-24):

- A second order model or
- A second order nonlinear actuator (with limits in actuation angle and rate). This is used by default
- A delay block, modeling the time that takes the computation in real IPDU (15 msec)

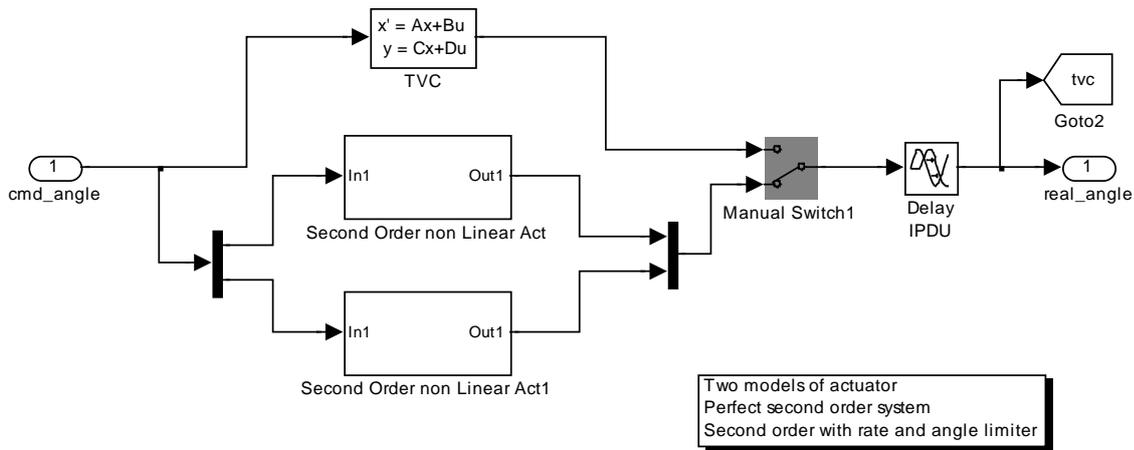


Figure 5-24: Model of the TVC

Each second order nonlinear actuator is as follows (Figure 5-25):

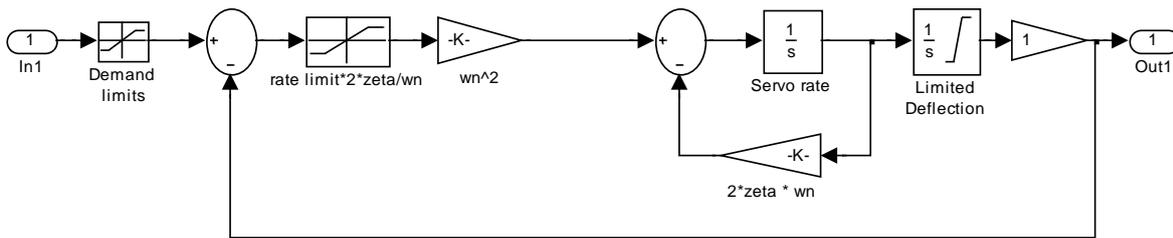


Figure 5-25: Second order model with position and rate limiters

5.4.2.5 The Wind model

The wind models in VEGA are based on:

- Real Winds: models generated from real measurements (observations made with balloons along many years in French Guyana)
- Synthetic models: typical scenarios generated from previous

Winds are significant only during the first stage and at the beginning of the second stage. At higher altitudes the air density decreases fast and the wind effect is negligible.

The wind data in VEGAMATH and VESAT are tables that are interpolated versus the altitude of the LV.

Usually the winds are described as intensity (m/s) of North-South wind and East-West wind for a given position (longitude, latitude) and altitude with regard the sea level. In addition to steady winds, wind gusts are also defined.

In VEGAMATH and VESAT the wind vector is interpolated based on the current LV position (longitude, latitude and altitude). This wind vector is transformed into ECI coordinates and then into LV Body coordinates, where it contributes to calculate the relative velocity of the LV with regard the surrounding air.

In *miniVEGA* we perform a small simplification. Basically assume that the wind model described in the interpolation tables is already described on ECI coordinates system.

5.4.2.6 The Launch Vehicle model

This is the most important part of the simulator. The overall structure is shown in (Figure 5-26):

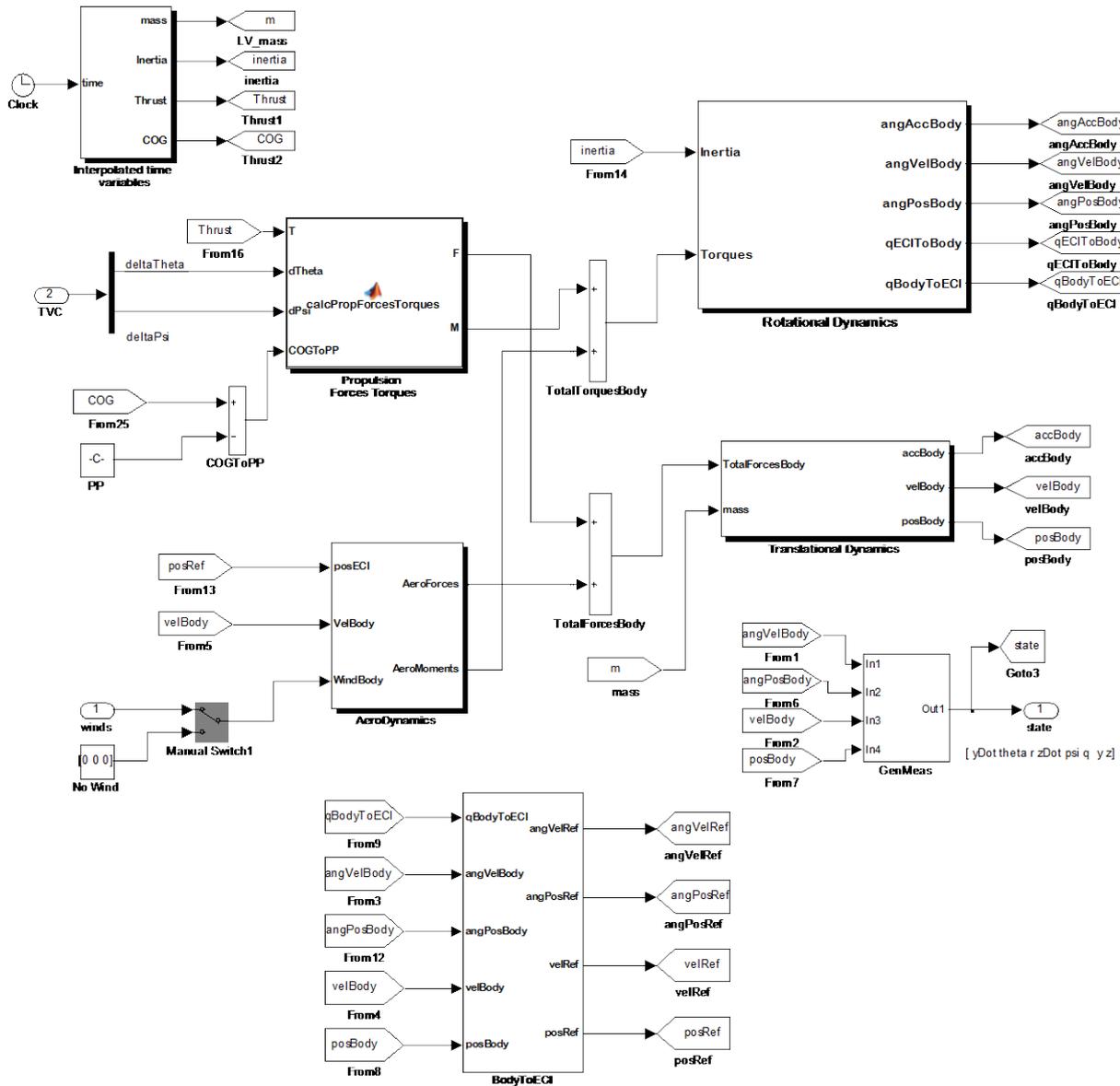


Figure 5-26: The Launch Vehicle model

5.4.2.6.1 The time interpolated variables subsystem

VEGAMATH and VESAT integrate four different propulsive models, one for each stage of the vehicle. The first 3 stages are solid rocket motors (SRM) and the fourth stage is a Liquid Propulsion System (LPS). We will describe here only the SRM first stage.

VEGAMATH and VESAT integrate a complex SRM model that computes the SRM parameters based on interpolation tables. These variables are mass consumption, thrust (nominal and at nozzle exit), thrust perturbations (oscillations of thrust called chuffing). From the SRM mass consumption is calculated the overall LV mass, inertia and COG position.

These SRM interpolation tables are predicted tables provided by the SRM manufacturer. The mass consumption and nominal thrust depend on the specific layout of the solid propellant inside of the SRM case. In addition these parameters depend also on the specific batch of production. For each SRM, the manufacturer takes a sample of the propellant and performs a dedicated test to fine tune the predicted propulsive tables. For our purposes we only have to know that the SRM manufacturer provides these tables (usually a set of 3 curves: nominal, upper and lower limit) and that these tables only depend on the combustion time.

The SRM model of VEGAMATH and VESAT takes more factors into account: chuffing (thrust oscillations), pressure at the nozzle exit that depends on the external atmospheric pressure that in turn depends on the altitude.

As the SRM tables are proprietary information that cannot be disclosed, *miniVEGA* follows a much more simplified approach. The following variables are a linear interpolation of the data of a nominal flight see (Figure 5-27).

- The SRM Thrust
- The LV Mass and Inertia
- The COG position

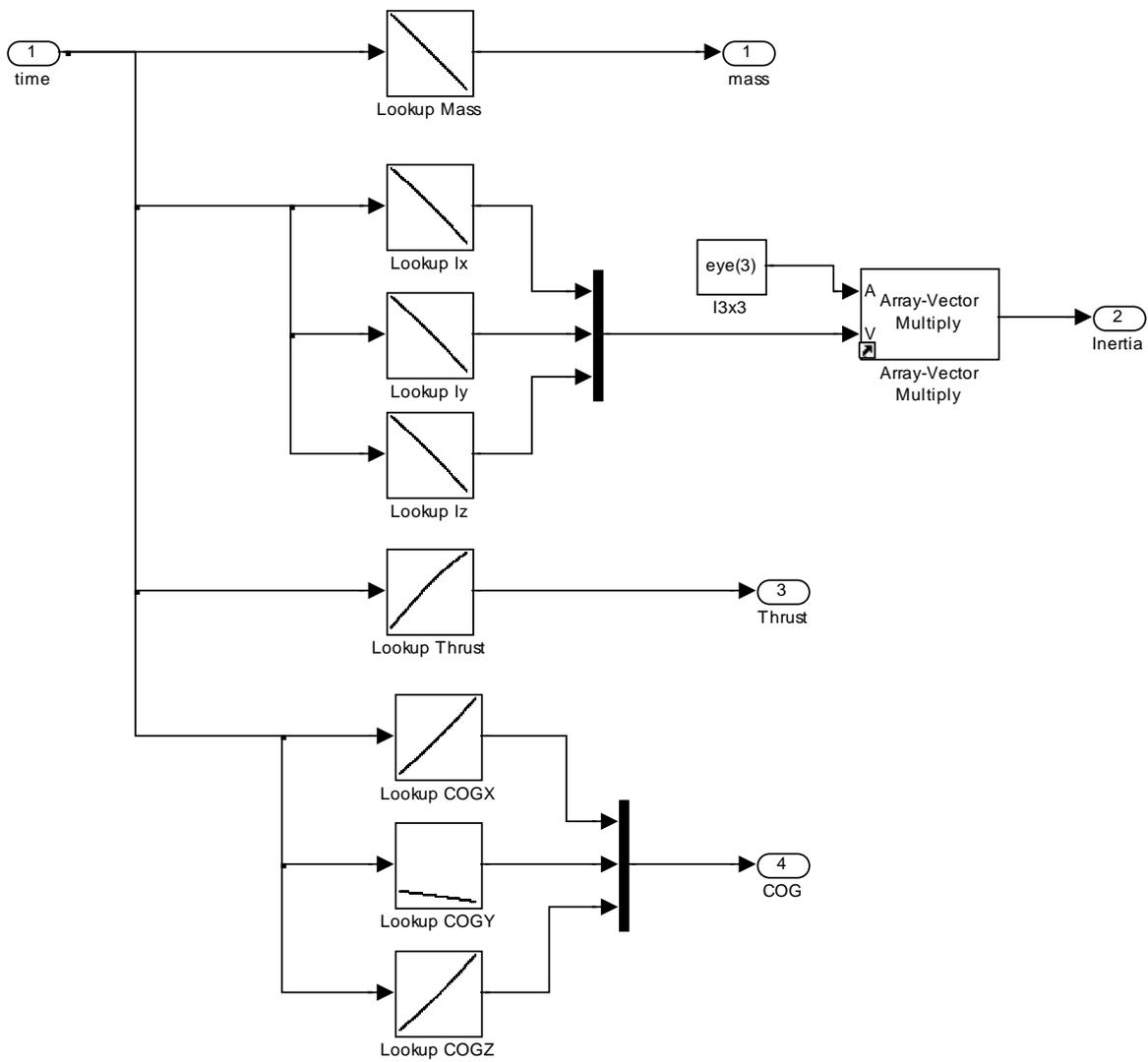


Figure 5-27: The interpolation variables model

5.4.2.6.2 The Propulsion Forces and Torques

This subsystem takes as inputs:

- The SRM Thrust
- The TVC actuators position
- The distance between the COG and the PP (actuator pivot point)

And calculates:

- The propulsive forces
- The propulsive torques

The block is implemented with embedded Matlab. Basically the propulsive force (that is aligned to the nozzle longitudinal axis) is projected into LV Body axis and the cross vector product of $-(COG-PP) \times F$ gives the torques

```
function [F, M] = calcPropForcesTorques(T, dTheta, dPsi, COGToPP)
% #codegen

% Calculates propulsion forces and torques
% Inputs
%   T: SRM Thrust
%   dTheta, dPsi: TVC angles
%   Vector PP to COG

F = [0 0 0]';
M = [0 0 0]';

% Fx, Fy, Fz
F(1) = T * cos(dPsi) * cos(dTheta);
F(2) = T * cos(dPsi) * sin(dTheta);
F(3) = -T * sin(dPsi) * cos(dTheta);

M = cross_product(-COGToPP, F);

function c = cross_product(a, b)

% cross vector product
c = [0 0 0]';

c(1) = a(2)*b(3) - a(3)*b(2);
c(2) = -(a(1)*b(3) - a(3)*b(1));
c(3) = a(1)*b(2) - a(2)*b(1);
```

5.4.2.6.3 The Aerodynamics Forces and Torques

This subsystem takes as inputs:

- The position in ECI
- The velocity in Body
- The winds in Body

And calculates:

- The aerodynamic forces
- The aerodynamic torques

It is implemented with the following Simulink diagram (Figure 5-28).

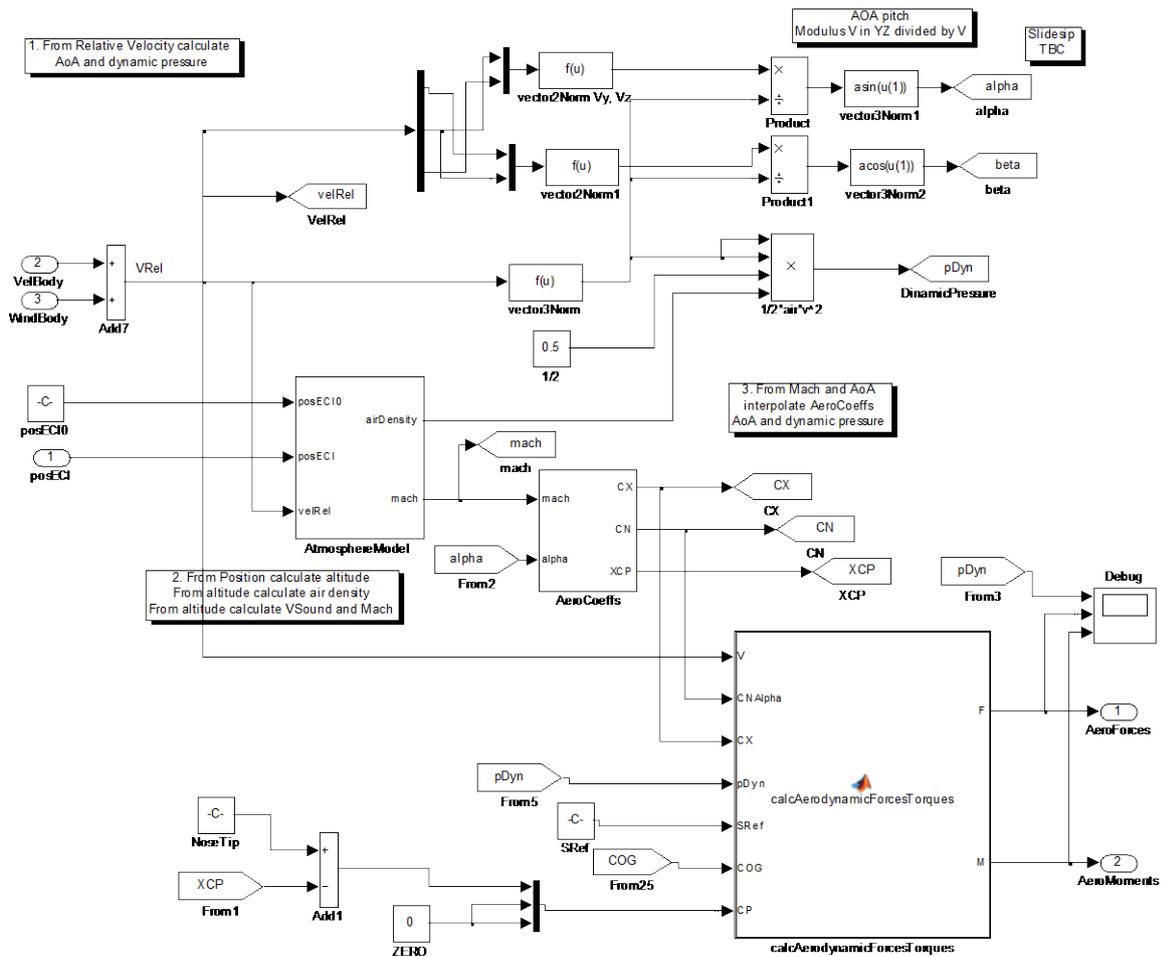


Figure 5-28: The aerodynamics model

The calculus is performed in 4 steps:

- From relative air velocity compute the angle of attack and dynamic pressure
- From position calculate altitude and from altitude calculate air density and Mach number
- From Mach and angle of attack calculate aerodynamic coefficients CN, CX and XCP
- From the previous calculate aerodynamics forces and torques

Calculus of angle of attack and dynamic pressure

The relative velocity (with regard the air flow) is Vel Body minus Wind Body. From this we calculate the angles of attack as

$$Alpha = asin(norm(VLat)/norm(V)), \quad \text{being } VLat = sqrt(vy^2+vz^2)$$

The dynamic pressure is calculated as

$$pDyn = 1/2 * airDensity * V^2$$

Calculus of altitude, air density and Mach number

The position in ECI is used by the Atmosphere model. This model receives as input the current ECI position and the ECI position at start of the simulation. The sum of both minus the normalized radius of Earth is the altitude over the sea level. The altitude is used to lookup into interpolation tables the air density and the Mach number. The Mach number is the velocity of sound at a given altitude divided by the module of the velocity in LV body coordinates

Calculus of aerodynamics coefficients

These are taken from interpolated tables based on Mach and angle of attack (alpha). The data of the tables are result of extensive test campaign into wind air tunnels performed specifically for VEGA.

The calculated aerodynamics coefficients are:

- CX : axial coefficient (i.e. into the longitudinal axis of the LV, X coordinate in LV Body)
- CN : normal coefficient (i.e. into the normal axis of the LV, Y, Z coordinates in LV Body)
- X_{CP} : position of x coordinate of center of pressure

Calculus of the aerodynamics torques and forces

This subsystem takes as inputs:

- The modulus of the relative velocity
- The CN , CX coefficients
- The dynamic pressure
- The reference surface (a constant)
- The COG position
- The CP (Center of pressure) position

And calculate as outputs:

- The aerodynamic forces
- The aerodynamics torques

This calculus is performed with embedded Matlab as follows:

```
function [F, M] = calcAerodynamicForcesTorques(V, CNAl pha, CX, pDyn, SRef, COG, CP)
% #codegen
% Calculates propulsion forces and torques
% Inputs
% T: SRM Thrust
% dTheta, dPsi: TVC angles
% COG
F = [0 0 0]';
M = [0 0 0]';
```

```

% Fx, Fy, Fz
F(1) = -CX * pDyn * SRef;

if V(2) == 0 && V(3) == 0
    F(2) = 0;
    F(3) = 0;
else
    F(2) = CNAl pha * ( -V(2)/ sqrt(V(2)^2+V(3)^2) ) * pDyn * SRef;
    F(3) = CNAl pha * ( -V(3)/ sqrt(V(2)^2+V(3)^2) ) * pDyn * SRef;
end

M = cross_product(CP - COG, F);
    
```

5.4.2.6.4 The Rotational Dynamics

This subsystem takes as inputs:

- The inertia
- The total torques (aerodynamics minus propulsive)

And calculates as outputs:

- The angular acceleration, position and velocity
- The quaternion Body to ECI
- The quaternion ECI to Body

It is implemented with the following Simulink diagram (Figure 5-29).

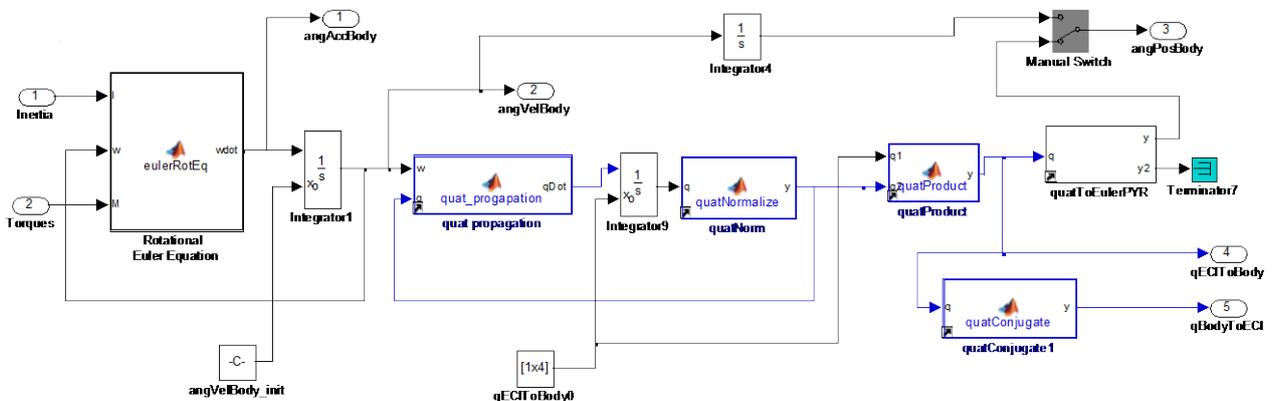


Figure 5-29: The rotational dynamics

The subsystem performs the calculus on 2 steps:

- Resolve the rotational Euler Equation, finding the angular acceleration into body
- Integrate the acceleration to find the angular velocity and angular position both in ECI and in LV Body

The rotational Euler equation is implemented with an embedded Matlab block. Basically, we have to resolve:

$$M = I\dot{\omega} + \omega \otimes (I\omega)$$

As:

$$\dot{\omega} = I^{-1}(M - \omega \otimes (I\omega))$$

```
function wdot = eulerRotEq(I, w, M)
% #codegen
% Given M = I * wdot + w x ( I * w )
% resolves
% wdot = 1/I * ( M - w x ( I * w ) )
aux = cross_product( w, I * w );
wdot = inv(I) * ( M - aux );
function c = cross_product(a, b)
% cross vector product
c = [0 0 0]';
c(1) = a(2)*b(3) - a(3)*b(2);
c(2) = -(a(1)*b(3) - a(3)*b(1));
c(3) = a(1)*b(2) - a(2)*b(1);
```

With this we obtain the angular acceleration that can be integrated twice for obtaining the angular velocity and position into Body. However a ‘serious’ simulator should not follow this approach as may cause inconsistencies due to the discontinuities inherent to the Euler angles.

Instead, the ‘propagation of the quaternion’ method is preferred. The acceleration is integrated to give the angular velocity. Then, the quaternion ECI to Body is calculated as:

$$\dot{q} = 0.5 \ q_{\omega} q$$

Were q_{ω} is a ‘fake’ quaternion build with ω as:

$$q_{\omega} = [0 \ \omega(1) \ \omega(2) \ \omega(3)]';$$

Refer to appendix 8.4.6 for a demonstration of the quaternion derivative.

The quaternion rate is integrated for calculating quaternion. Note that this quaternion rate is ECI to Body. We can integrate it to find the quaternion ECI to Body (i.e. how to rotate ECI to arrive to LV Body).

We can obtain the Euler angles using the utility function *quatToEulerPYR()*. We have to use the order (pitch, yaw, roll) because this is the order when changing from ECI to Body. The quaternion Body to ECI is just the quaternion conjugate of ECI to Body.

Notes:

- The constant term for the integration of the angular acceleration (i.e. the angular velocity at the start at the simulation) will be defined according to our simulation

needs (example, we will select [roll, yaw, pitch] = [20, 0, 0] deg/s if our intention is to simulate with an initial roll rate of 20 deg/s.

- The constant term for the integration of the quaternion is [1, 0, 0, 0] that is 'LV body frame is not rotated'. This is a license we take. We impose that at the beginning of the simulation the reference frames ECI and LV Body are aligned. This made easier the interpretation of the simulations.

5.4.2.6.5 The Translational Dynamics

This subsystem takes as inputs:

- The mass
- The total forces (aerodynamics plus propulsive)

And calculates as outputs:

- The linear acceleration, position and velocity

All of this is implemented with the following Simulink diagram (Figure 5-30).

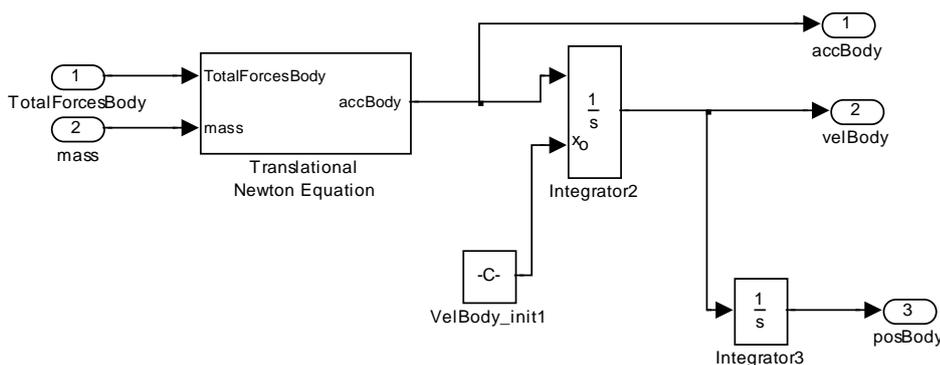


Figure 5-30: The translational dynamics

The process is very simple. The acceleration is calculated applying the Newton formula. Then it is integrated (LV body) to find the velocity and position. Note that we take as constant integrator term of the velocity the initial velocity at the beginning of the simulation (usually is a high velocity on the X coordinate and zero or near to zero in Y, and Z coordinates). For the position, we take [0, 0, 0] as initial position at the start of the simulation.

5.4.2.6.6 Body to ECI Blocks

This auxiliary subsystem just transforms parameters from Body to ECI. It takes as inputs:

- The quaternion Body to ECI
- The angular velocity and positions in LV Body
- The velocity and positions in LV Body

And calculates as outputs:

- The angular velocity and positions in ECI
- The velocity and positions in ECI

5.4.2.6.7 GenMeas Blocks

This auxiliary subsystem just generates the LV state. From this LV state we take the individual measures needed by the controllers. It takes as inputs:

- The angular velocity and positions in LV Body
- The velocity and positions in LV Body

and calculates as outputs:

- The angular velocity components (p, q, r)
- The angular position components (ϕ, ψ, θ)
- The velocity components (v_x, v_y, v_z)
- The position components (x, y, z)

5.4.2.7 The mathematical library

This is a Simulink library with mathematical functions as:

- Quaternion operations
 - o product
 - o norm
 - o conjugate
 - o quaternion propagation
 - o quaternion to Euler and Euler to quaternion conversions
- Norm of a vector

The library is coded in “Embedded Matlab”.

5.4.2.8 Non modeled parts in miniVEGA

The following LV features are not modeled

- Second, third and fourth stages
- Stages ignition and separation
- Bending modes
- Sloshing modes
- Nozzle inertia dynamics
- Gravity (Gravity is considered constant)

A justification of why the gravity model is not included follows. First, the P80 stage is separated at the altitude of about 60 km. At this altitude g is about 9.62 m/s^2 . At $t = 55$ sec (time of maximum dynamic pressure) the altitude is about 12 km and the gravity is about 9.77 m/s^2 . So for the first stage and in a first approximation we can consider the gravity constant.

5.5 Comparison of controllers for the VEGA Launcher

5.5.1 The reference ELV controllers

5.5.1.1 The original ELV PID controller

The structure of the controller has been described in 5.1.3.

The original ELV controller is designed as a discrete PID controller plus a number of discrete filters. For the P80 phase the tuning is composed of a set of 12 groups that are scheduled based on the non-gravitational velocity. At a time $t = 50$ s this corresponds to the group number 7.

The original filters are described on the digital domain. The conversion to the continuous domain is achieved with the function $d2c()$ using a step of 0.001 ms and the method 'Tustin'.

It has been noted that the simulation on the continuous domain has problem with the derivative filter H_2 . We have replaced it with a perfect derivative filter, i.e. $H_2 = tf([1 \ 0], 1)$

The Matlab commands to build the controller are:

```
sysPsi = series( H.P80.H1.tf, H.P80.gains.ks ) + series( H2, H.P80.gains.kd );
sysPsi = series( sysPsi, H.P80.H3.tf );
sysTheta = sysPsi;
K = append(sysTheta, sysPsi);
```

Note also that the purpose of the filter H_3 is to control the bending modes. In our exercise we do not incorporate bending modes so the behavior with and without H_3 filter are almost identical (with H_3 , small oscillations in the TVC actuators are filtered out).

5.5.1.2 The ELV PID controller with gyroscopic compensation

The structure of the controller with gyroscopic compensation has been described in 5.2.2.

The controller is very similar to the previous but the gyroscopic filter H_5 is introduced.

It shall be noted that the gyro controller is a non-linear controller (due to the products 'p*q' and 'p*r'). Due to this a linear controller cannot be fully implemented with Matlab commands only: instead, we define the components of the controller in Matlab and compound the controller with Simulink.

The Matlab commands to build the first component of the controller are:

```

sysPsi = series( H. P80. H1. tf, H. P80. gains. ks ) + series( H2, H. P80. gains. kd );
sysPsi = series( sysPsi, H. P80. H3. tf );

sysTheta = sysPsi;

H5 = H. P80. H5. tf;
    
```

The H_5 controller is a derivative filter able to compute the p, q, r from the quaternion. As we are supposing perfect navigation, we have access to p, q, r directly and the filter H_5 is not used. (Note: as separate exercise we have synthesized the gyro controller also using the H_5 filter and the results are similar, except that the H_5 as any derivative filter is noisy and made the TVC actuation to be noisier).

As with the previous controller, the only purpose of the filter H_3 is to control the bending modes. In our simulation we do not use them, so the behavior with and without H_3 filter are almost identical (with H_3 , small oscillations on the TVC actuators are filtered out).

The two parts of the controller are composed in Simulink as follows (Figure 5-31):

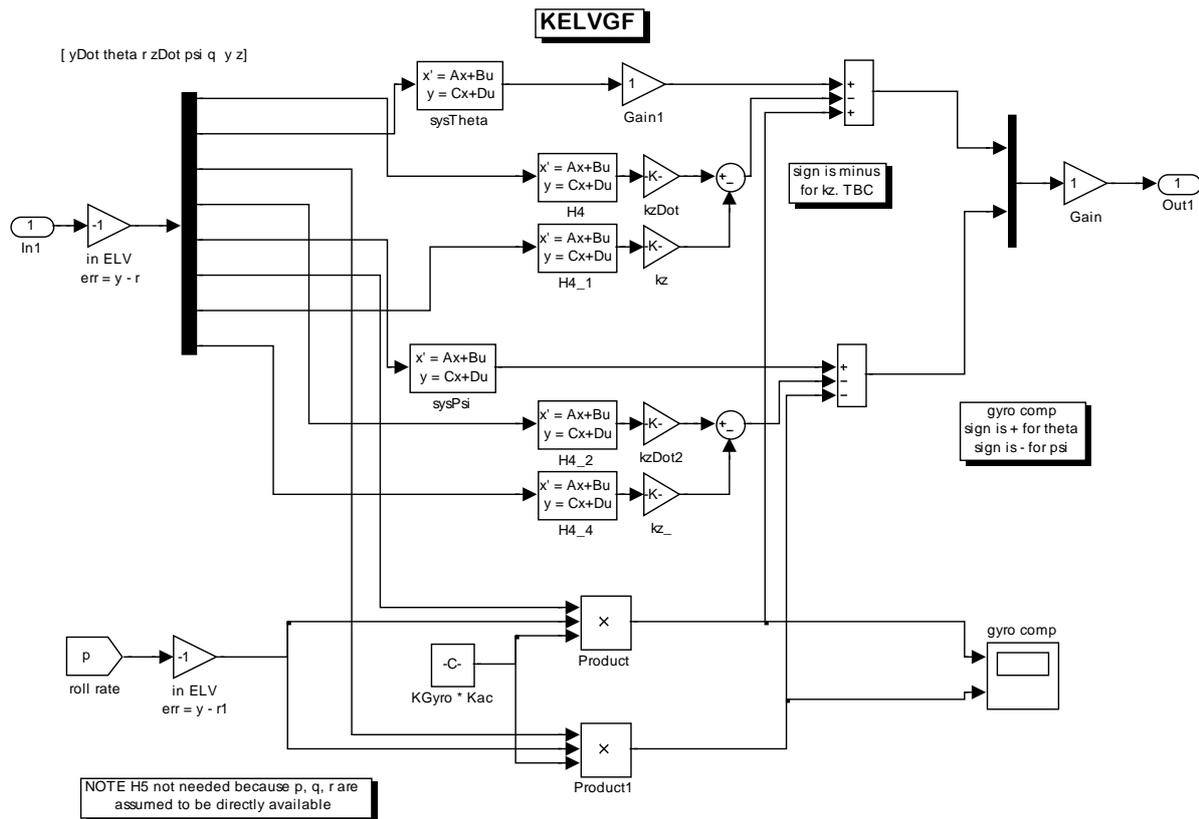


Figure 5-31: Schema of the ELV controller with Gyroscopic compensation

5.5.2 The proposed H-infinity controllers

Two H-Infinity controllers are proposed in this thesis: one based on the CHGE method and other based on the Structured method.

5.5.2.1 The proposed H-infinity controller (CHGE)

The approach of 'Christen-Geering' described in previous chapters has been followed for the design and synthesis of the controller.

Note: A paper based on this controller has been presented during the ACA 2013, IFAC Symposium on Automatic Control in Aerospace (Sanchez, 2013).

The controlled variables chosen are theta and psi. The y and z states are removed from the lineal plant. We have decided not to control the lateral deviations. This is justified because the original requirements states the lateral control is of less importance during the atmospheric phase. In fact, the previous ELV controllers set low gains for the lateral channels. If the result is not satisfactory the control of the lateral deviations would be implemented.

With this election of measures (theta and psi) the plant does not need input-output scaling. (It would have been different if the lateral deviations were included in the control). Note however that the CHGE methods needs scaling of the transfer functions (see later).

A total delay of 37 ms (IMU = 10 ms, OBC = 12 ms, TVC=15 ms) is introduced as a Padé second order transform.

Choice of Weight W_d

A weight $W_d \ll 1$ is taken. This made that KS_o and S_o does not contribute too much to the norm, so practically we are weighting only T_i and S_oP .

Choice of Weight W_p for S_oP

For the S_o part we select ideal second order systems with:

- No overshoot
- Settling time of 0.400 ms (10 times the period of control of 40 ms)

One of the possible second order systems with these performances is given by:

- $\omega_n = 20$; $\delta = 1$

Due to the symmetry of the plant for both channels (pitch and yaw), the same weight is chosen for them.

For the P_o part of the weight, we take just the nominal steady gain of the open loop plant:

- $A = \max(\sigma(G)) = 13.35$

$$W_p = \text{inv}(A) \text{inv} \begin{pmatrix} 1 - \frac{20^2}{s^2 + 20s + 20^2} & 0 \\ 0 & 1 - \frac{20^2}{s^2 + 20s + 20^2} \end{pmatrix}$$

Choice of Weight W_u for T_i

W_u weights both T_i and KS_o in the CHGE formulation:

$$\begin{bmatrix} dy \\ z \end{bmatrix} = \begin{bmatrix} -W_u T_i & -W_u K S_o W_d \\ W_p S_o P & W_p T_o W_d \end{bmatrix} \begin{bmatrix} du \\ dw \end{bmatrix}$$

We presented in the previous chapters the guidelines for selection of the weight for T_o .

Note that on MIMO, T_i is in general not equal to T_o , but in a first approach they are similar. We can follow the same strategy for the S_o part of W_p : choose a weight T_i as the inverse of an ideal system T_{o_id} . The weight needs to be a proper transfer function so we have to add zeros.

An alternative strategy is to think that W_u is weighting KS_o (i.e. references to actuators). In this case the weight shall be a high pass filter so its inverse limits the actuation at high frequencies.

In any of the alternatives, the fact that the transfer functions being weighted shall have a similar magnitude have been explained in (4.3.3). A factor similar to $1/P_o$ has been introduced in the weight W_p . We should introduce a similar factor in the weight for W_u , restoring the situation where both weighted TF contribute equivalently to the norm.

In this concrete problem we chose a minimalistic W_u weight that is simply the identity weight plus the scalar factor 'A' (equal to the maximum singular value of P_o).

Summarizing, W_u is chosen as:

- Choose the weight as $\text{inv}(A * \text{eye}(2))$

$$W_u = \text{inv} \left(A \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right)$$

The following figures show the singular values diagrams of the controller and closed loop plant. We have introduced uncertainty on the plant that as was discussed in (5.3.5.1) depends mostly on the roll rate.

The synthesized controller K and KS_o are shown in (Figure 5-32).

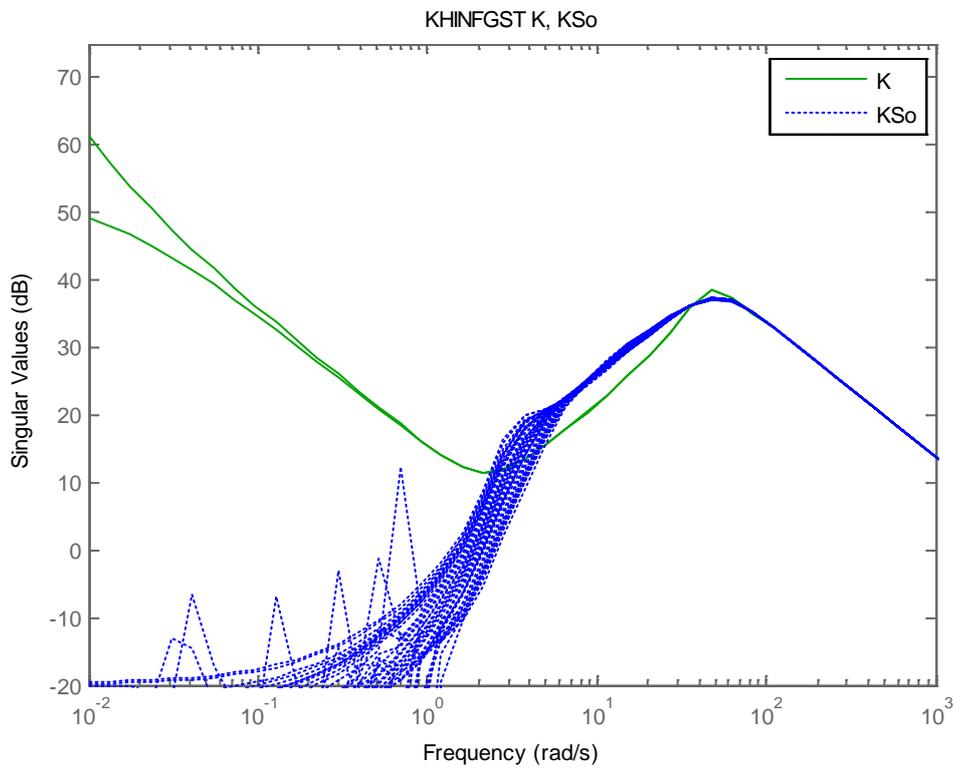


Figure 5-32: K and KS_o for the CHGE controller

The figures of S_oP and T_o (Figure 5-33) for a set of uncertainty plants shows that the magnitude of S_oP is always below T_o . This means that the system will have a good degree of robustness and will be in principle robust to input uncertainties (uncertainty at plant inputs: actuators and winds)

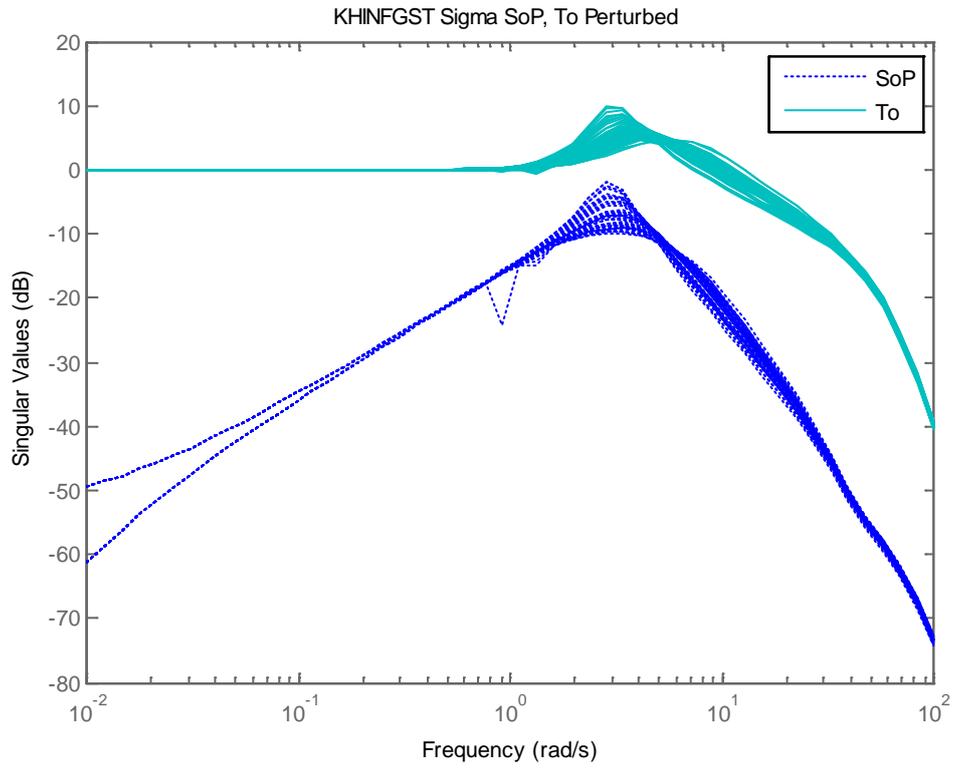


Figure 5-33: S_oP and T_o for the CHGE controller

The figures of S_o , S_oP (Figure 5-34) and the inverse of W_y are shown. S_o changes with the uncertainty (roll rate) so the performances of the system will change depending on the roll rate. All the S_o curves are under the weight so in principle all the plants will meet the required performances.

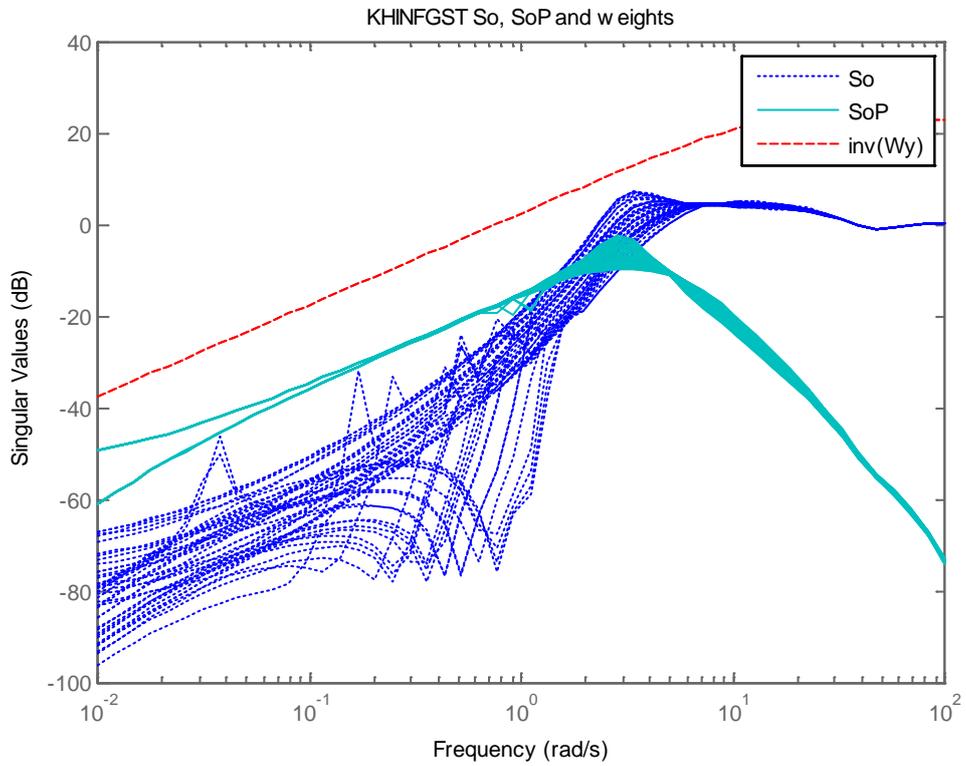


Figure 5-34: S_o and S_oP for the CHGE controller

The figures of T_o , T_i and W_u (Figure 5-35) are shown.

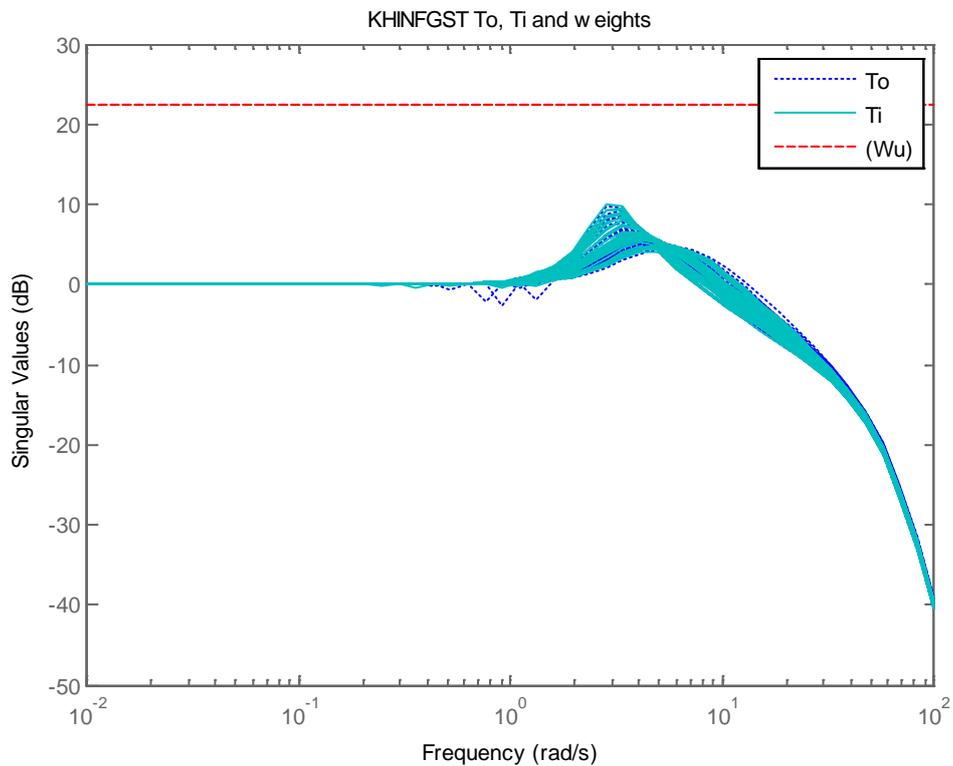


Figure 5-35: T_o , T_i and weights for the CHGE controller

5.5.2.2 The proposed H-infinity controller (Structured)

The Structured 'Apkarian' approach described in previous chapters has been followed for the design and synthesis of the controller.

We will follow for the control a similar strategy for the CDC plant. A static decoupler and 2 PIDs are chosen as control structure (note PIDs are used instead PI). Also a roll-off block is used.

Then, the following weighting scheme (Figure 5-36) is designed:

- A weight W_s , weighting S
- A weight W_d , weighting the disturbances at plant input
- A weight W_u , weighting the control signal has been introduced

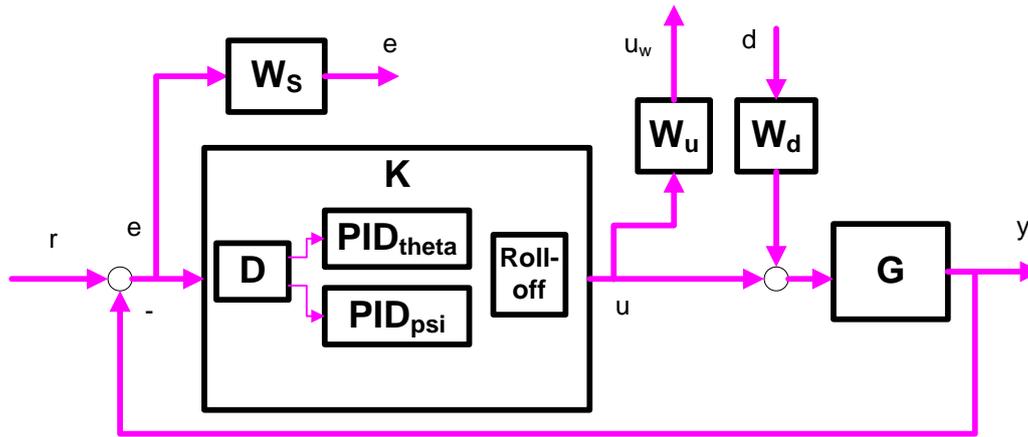


Figure 5-36: Setup for the K-infinity Structured controller

The augmented plant is created with the standard Matlab function 'connect'. The equations are created as follows.

```
DM = ltiblock.gain('Decoupler',eye(2));

% PIDs
PID_Theta = ltiblock.pid('PID_Theta','PID');
PID_Psi = ltiblock.pid('PID_Psi','PID');

% Label block I/Os: Inputs and outputs of the plant
GDesign.InputName = {'iTheta','iPsi'};
GDesign.OutputName = {'oTheta','oPsi'};

% Input to the decoupler is the error signal
DM.u = 'e'; DM.y = {'oDTheta','oDPsi'};

% Input to each PID are the decoupler outputs
PID_Theta.u = 'oDTheta'; PID_Theta.y = 'oPID_Theta';
PID_Psi.u = 'oDPsi'; PID_Psi.y = 'oPID_Psi';

% Error is reference minus output of the plant. Ref components rTheta, rPsi
Sum1 = sumblk('e = %r - %y', {'rTheta','rPsi'}, GDesign.y);

% Input to plant G is sum of disturbance and controller output (PID output)
Sum2 = sumblk('%up = %uc + d', GDesign.u, [PID_Theta.y ; PID_Psi.y]);

CL0 = connect(GDesign,DM,PID_Theta,PID_Psi,Sum1,Sum2,
{'rTheta','rPsi','d'},{'e','oTheta','oPsi','oPID_Theta','oPID_Psi'});
CL0 = blkdiag(Wp, eye(2), Wu) * CL0 * blkdiag(eye(2), Wd);

op = hinfstructOptions('RandomStart',3);
CL = hinfstruct(CL0, op);
```

Choice of Weight W_s

We follow the strategy used previously, an 'ideal 2nd order system' instead of the original weight proposed by Apkarian.

- Choose a 2nd order system that meets the specifications (chose ω_n and delta)
- $T_{o_id} = \omega_n^2 / (s^2 + 2 \text{ delta } \omega_n s + \omega_n^2)$
- Calculate $S_{o_id} = 1 - T_{o_id}$
- Define the weight as the inverse of S_{o_id}

We take $\omega_n = 20$ and delta = 1.0 for both channels. This corresponds to a response time of 400 msec that is 10 times greater than the control period (40 msec).

$$W_s = \text{inv} \begin{pmatrix} 1.00001 - \frac{1}{s^2 + 40s + 20^2} & 0 \\ 0 & 1.00001 - \frac{1}{s^2 + 40s + 20^2} \end{pmatrix}$$

Choice of Weight W_d

In this problem the main source of uncertainty at plant input is the roll rate.

From the open loop diagram of the uncertainty plant we can see that static gain at 45 deg/s is about 7 dB greater than the static gain at 0 deg/s. This means about 2.3 times greater. This suggests taking a uncertainty of about a 230%, so the weight is:

$$W_d = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Choice of Weight W_u

We want to penalize the actuators effort at high frequencies. We chose the standard form of a weight.

$$W_u = \frac{1}{M} \frac{s + \omega_b}{s + \omega_b A}$$

The thrust vector control for the VEGA nozzle has the following limits:

- Maximum deflection ± 7 degrees
- Maximum deflection rate 10 degrees/s

Following the rules described in chapter 2 for selection of the actuator weight we have:

- A frequency for the actuator $\omega_{act} = 2.24$ rad/s and we should take bandwidth about $\omega_B = 22.4$ rad/s
- Maximum deflection, $A = 7 \cdot \pi / 180$.

- We take $M = 0.9$.

The weight is identical for both channels:

$$W_d = \begin{pmatrix} \frac{1}{M} \frac{s + \omega_b}{s + \omega_b A} & 0 \\ 0 & \frac{1}{M} \frac{s + \omega_b}{s + \omega_b A} \end{pmatrix} = \begin{pmatrix} \frac{1}{0.9} \frac{s + 22.4}{s + 22.4 \frac{7\pi}{180}} & 0 \\ 0 & \frac{1}{0.9} \frac{s + 22.4}{s + 22.4 \frac{7\pi}{180}} \end{pmatrix}$$

We know that the PID controller will not roll-off at high frequencies. For the Launcher this is undesirable because it will mean coupling with the bending modes of the Launcher. Even if they are not taken into account we will add a roll-off term a posteriori.

```
% Obtain the blocks of the controller
DM2 = getBlockValue(CL, 'Decoupler');

PID_Theta2 = getBlockValue(CL, 'PID_Theta');
PID_Psi2 = getBlockValue(CL, 'PID_Psi');
K = blkdiag(PID_Theta2, PID_Psi2) * DM2;

% roll off
K = blkdiag(tf(1, [0.005 1]), tf(1, [0.01 1])) * K;
```

The following figures show the singular values diagrams of the controller and closed loop plant. We have introduced uncertainty on the plant that as was discussed in (5.3.5.1) depends mostly on the roll rate.

The synthesized controller K and KS_o are shown (Figure 5-37).

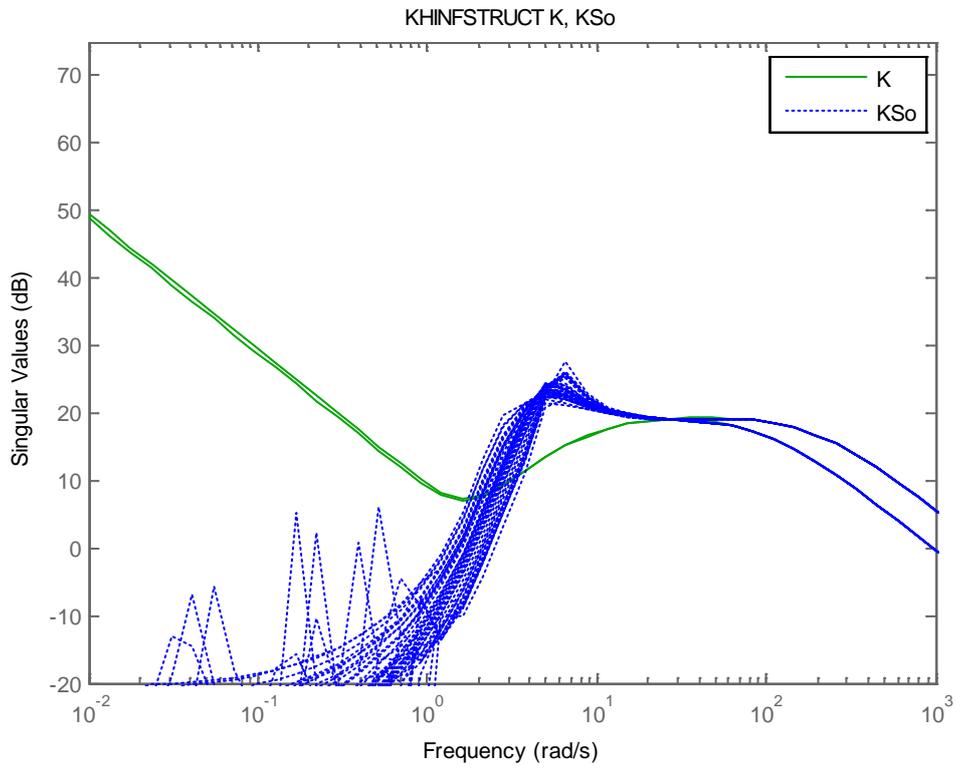


Figure 5-37: K and KS_o for the structured controller

The figures of S_oP and T_o (Figure 5-38) for a set of uncertainty plants shows that the magnitude of S_oP is always below T_o . This means that the system will have a good degree of robustness and will be in principle robust to input uncertainties (uncertainty at plant inputs: actuators and winds)

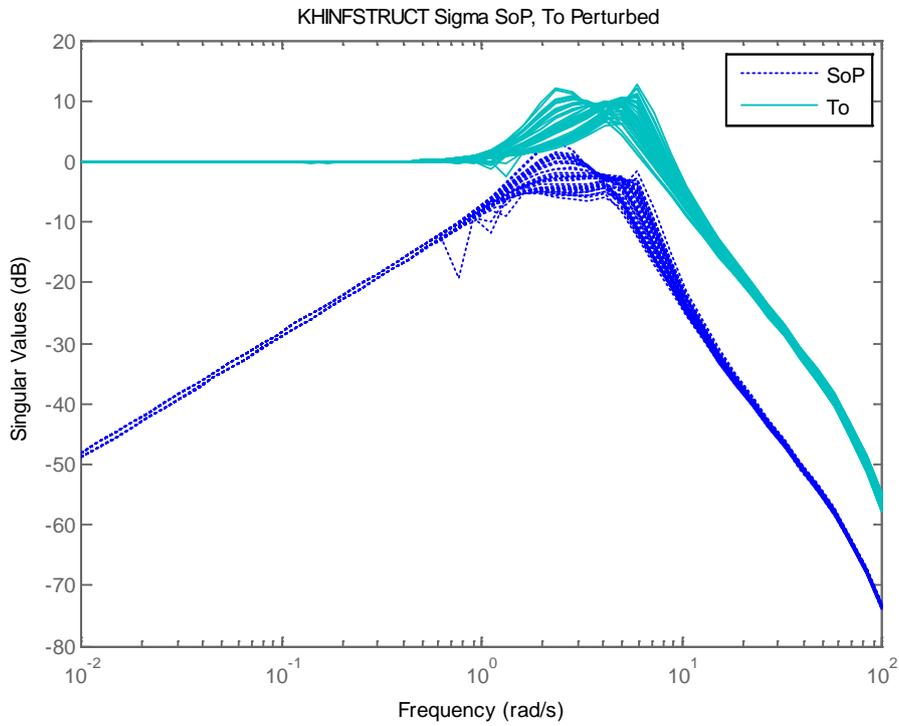


Figure 5-38: S_oP and T_o for the structured controller

The figures of T_o , T_i and W_u (Figure 5-39) are shown.

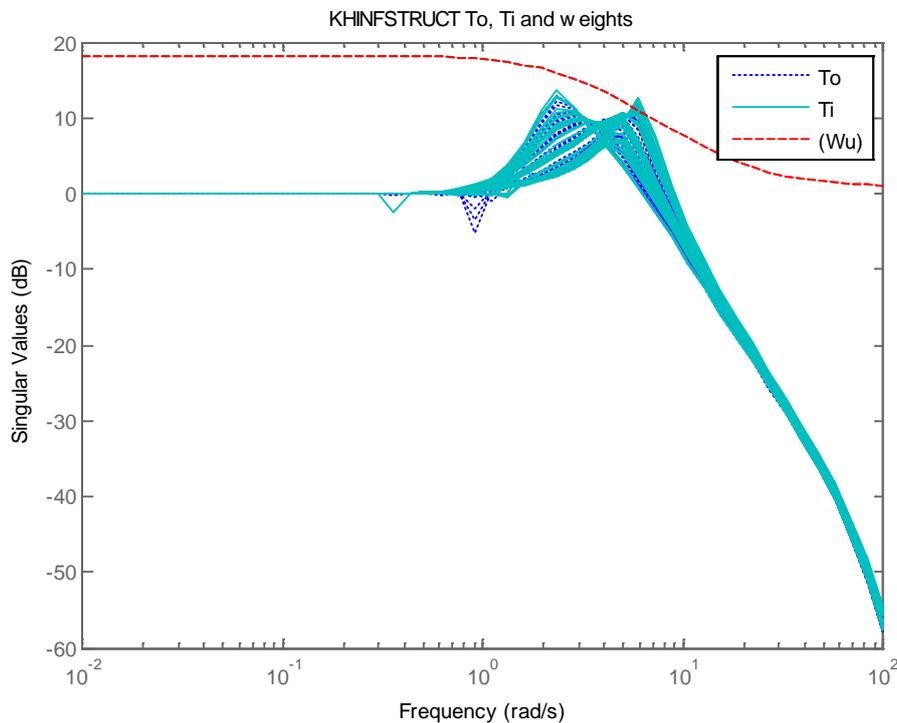


Figure 5-39: T_o , T_i and weights for the structured controller

5.5.3 Simulation scenarios

A number of scenarios have been defined to evaluate the performances of the controllers. Each scenario defines different LV inputs and different wind conditions. Three scenarios have been defined:

- Response to ramps commands in theta, psi in absence of winds
- Response to ramps commands in theta, psi in presence of light winds
- Response to medium to strong winds and wind gusts (regulation problem)

The ramp commands are defined as:

- For theta, a ramp that starts at $t = t_0$ and reaches 1 deg at $t = t_0 + 3$ s.
- For psi, a ramp that starts at $t = t_0$ and reaches 1.5 deg at $t = t_0 + 5$ s.

Note: the slope of these ramps is even greater than the commands typically sent by the guidance during the atmospheric flight. For a launcher the commands for changing the direction are very smooth.

Each scenario is run with combinations of:

- The controller under test
- A set of roll rates:

- no roll (0 deg/s),
- slight roll (10 deg/s),
- medium roll (30 deg/s),
- high roll (45 deg/s)
- negative high roll (-45 deg/s)

We use the following notation for the controllers:

- KHINFGST: the proposed H-infinity CHGE controller
- KHINFSTRUCT: the proposed H-infinity KHINFSTRUCT controller
- KELVIG: the ELV PID controller (with gyro compensation and outer loop disconnected)
- KELVGF: the ELV PID controller (with gyro compensation and outer loop connected)

Note: the scenarios are run at from $t = 50$ s to $t = 65$ s ($t = 0$ is lift-off). The justification is that at these times the maximum dynamic pressure along the flight is reached. This is considered the most difficult phase of the atmospheric flight because the stability margin is minimum.

5.5.4 Simulation analysis

5.5.4.1 Scenario 1: response to ramps in absence of winds

Ramp inputs are commanded in theta and psi with different roll rates. It shall be noted that for the ELV controllers the outer loop in position and velocity is disconnected in this scenario. The justification is as follows: if a constant pitch angle is required, we should also command a change on the $yDot$ velocity and y position. If $yDot$ and y are kept to zero, we are sending contradictory orders to the controller. In order to keep the guidance as simple as possible, the outer loop is disconnected in this scenario.

First let's analyze the simulation at zero roll rate. Results are very similar between the compared controllers (Figure 5-40). The KHINFSTRUCT controller is the more aggressive (more activity in the actuators). Note also that the results of the ELV PID and the ELV PID Gyro controller are identical as expected: the gyro compensation does not have any effect with $p = 0$ deg/s.

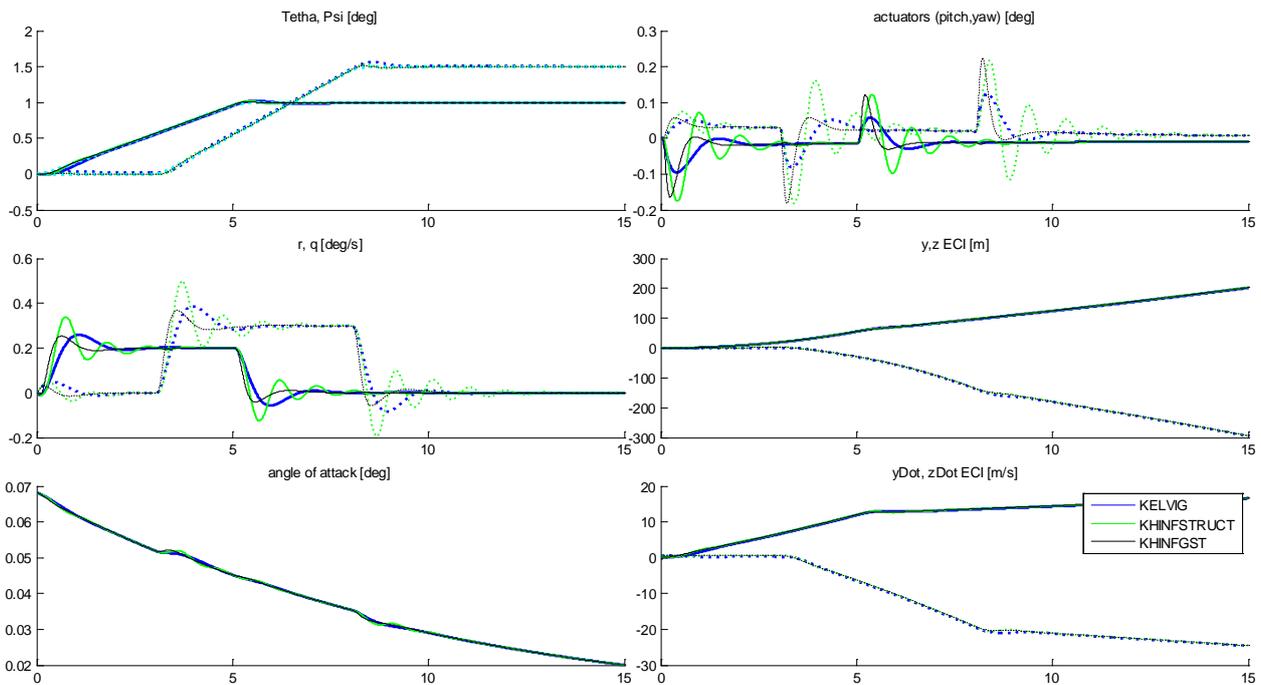


Figure 5-40: Scenario 1 roll rate $p = 0$ deg/s

Note: in all the figures the continuous line is the variable for the pitch plane and the dotted line the variable on the yaw plane. Each color represents a controller.

The results are almost similar for the simulation at 10 deg/s of roll rate.

In the simulation at 30 deg/s of roll rate is appreciated an interesting effect: the ELV PID controller has an error in the stationary on the tracking of θ and ψ . In both ELV PID

Gyro and KHINFGST controller this stationary tracking error is not present. However this effect is very small in absence of winds.

The effect is more visible in the simulation at 45 deg/s (or the one at -45 deg/s), (Figure 5-41). The stationary error for the yaw command is 0.15 deg for ELV PID and a peak of 0.25 deg is present. For the ELV PID GYRO the error is 0.05 deg and the peak is 0.12 deg. For the KHINFGST and KHINFSTRUCT controller the error is almost zero and the peak is 0.05 deg.

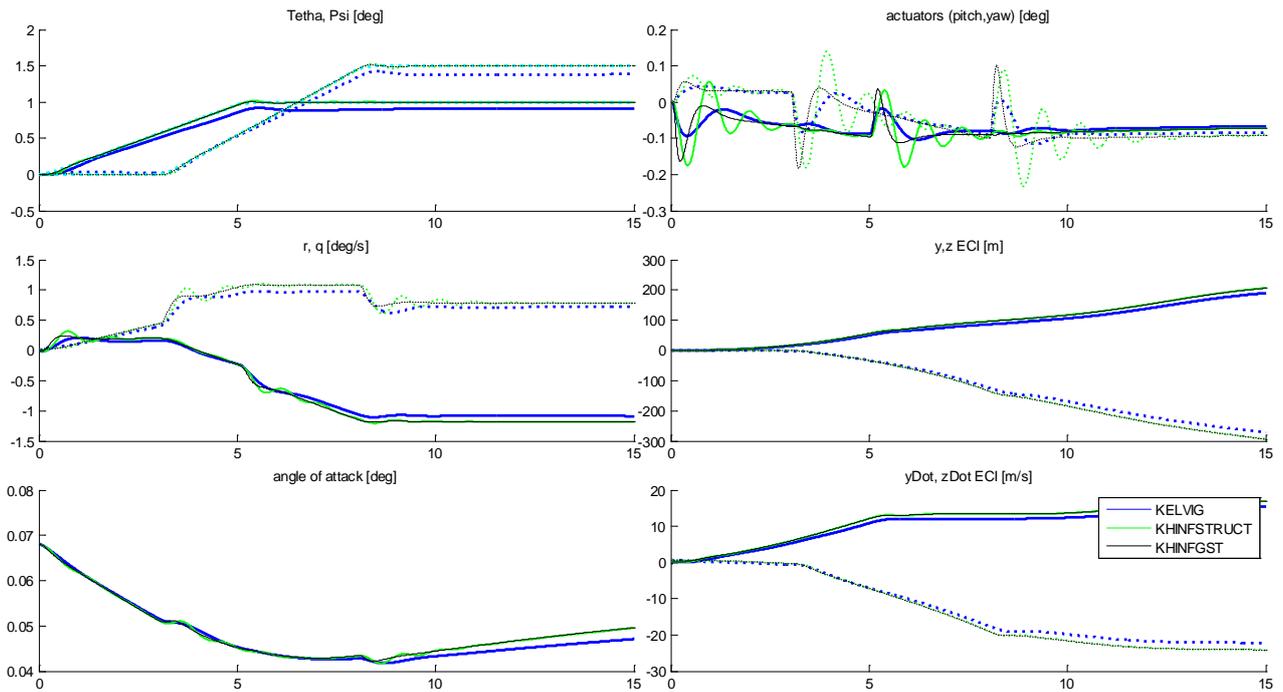


Figure 5-41: Scenario 1 roll rate p = 45 deg/s

5.5.4.2 Scenario 2: response to ramps in presence of winds

Ramp inputs are commanded in theta and psi with different roll rates and this time with lateral winds. As in the previous scenario the outer loop in position and velocity is disconnected for the ELV PID and ELV PID Gyro controller.

First let's analyze the simulation at zero roll rate (Figure 5-42). The presence of lateral winds introduces stationary errors in for KELVIG even when the roll rate is zero. The error is about 0.25 deg for yaw. The KHINFGST AND KHINFSTRUCT controller does not have stationary error. The results of the KELVI and the KELVIG controller are identical as the gyro compensation does not have any effect with $p = 0$ deg/s (only the KELVIG is shown for clarity).

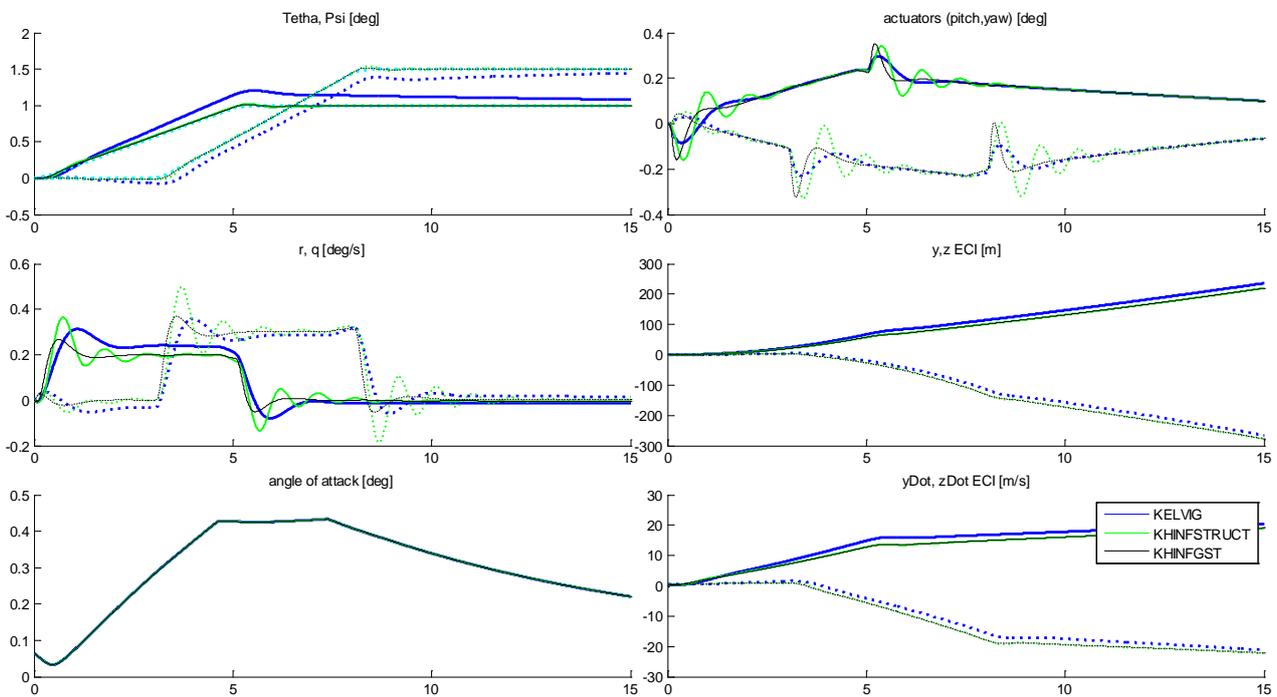


Figure 5-42: Scenario 2 roll rate $p = 0$ deg/s

At 10 deg/s the effect of the lateral winds on the tracking errors in *theta* and *psi* start to manifest (Figure 5-43). The difference between the KELVI and the KELVIG is very small because the gyro compensation is tuned for relative high roll rates.

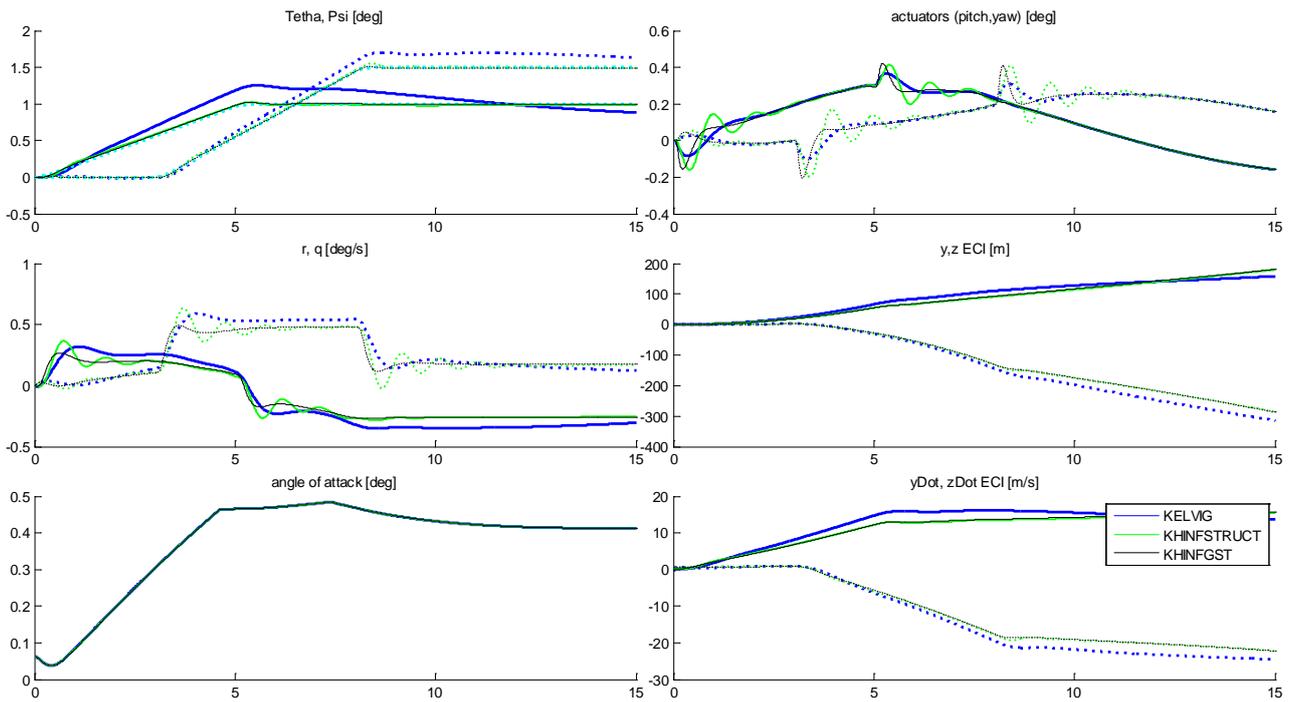


Figure 5-43: Scenario 2 roll rate $p = 10 \text{ deg/s}$

When the roll rate increases the situation changes. As the LV is rolling the wind is not received always on the same side of the lv. At $p = 10 \text{ deg/s}$, after 9 seconds the LV has rolled 90 deg. At this time the wind is received on the yaw plane. This is more evident when the roll rate increases. At 45 deg/s the LV completes a full rotation in 8 seconds. This creates oscillations in the figures of *theta*, *psi* and the actuators that try to counteract the changing wind. The situation is symmetric for a roll rate of minus 45 deg/s (Figure 5-44).

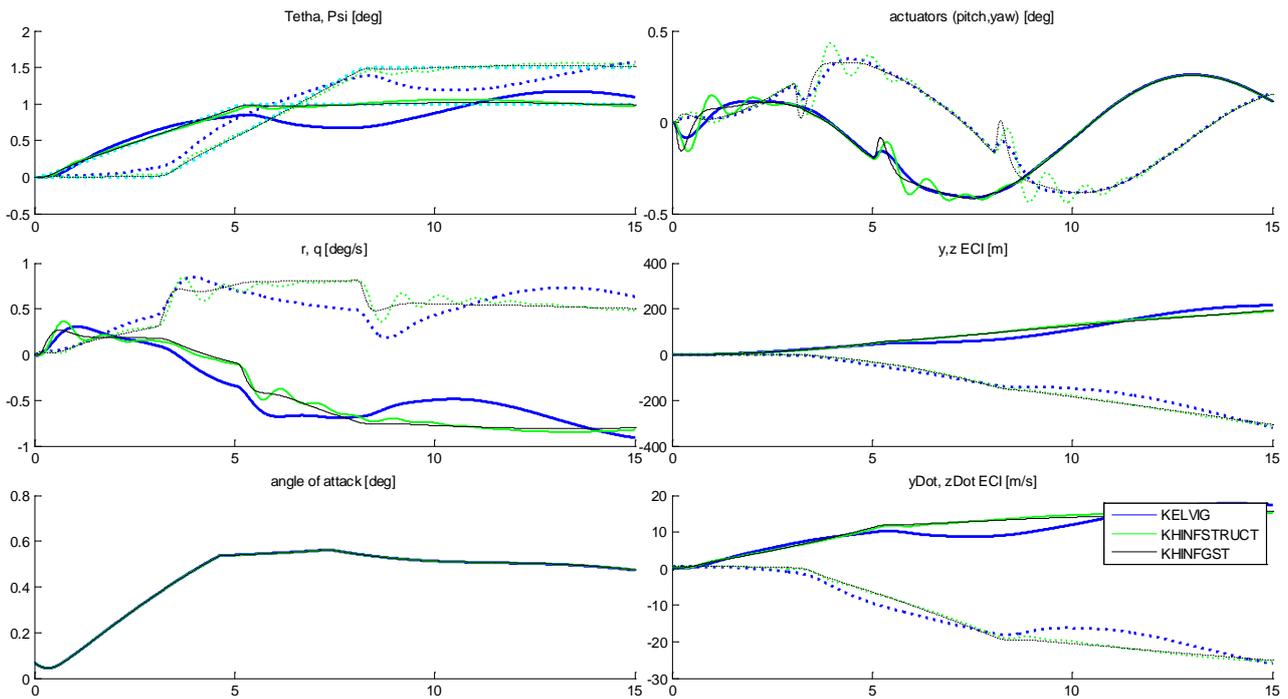


Figure 5-44: Scenario 2 roll rate $p = 45 \text{ deg/s}$

Only the KHINFGST and KHINFSTRUCT controllers are able to keep a low stationary error and to decrease the amplitude of the oscillations on theta and psi. The (Figure 5-45) shows a comparison between these 2 controllers. The KHINFGST controller is slightly better than the KHINFSTRUCT that present a less damped behavior.

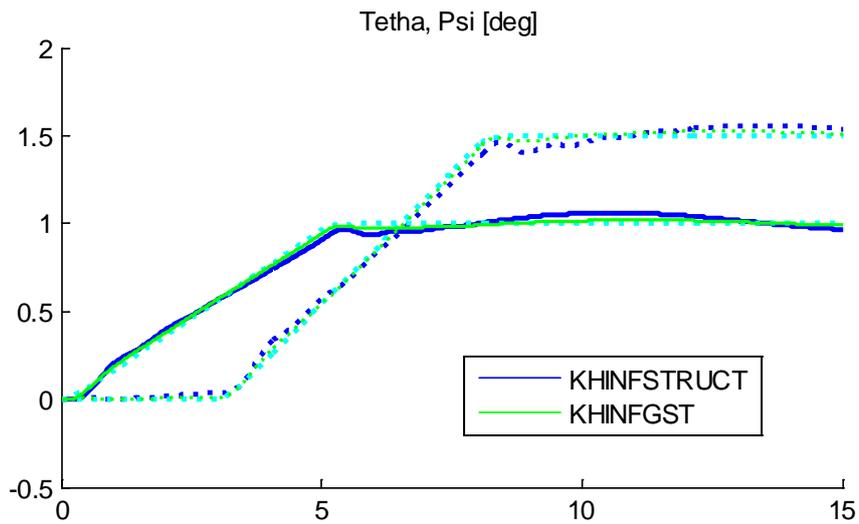


Figure 5-45: Scenario 2 roll rate $p = 45 \text{ deg/s}$ (zoom)

5.5.4.3 Scenario 3: disturbance rejections with wind gusts

This is a disturbance rejection problem in presence of winds. The references are commanded to zero. Medium to strong winds and wind gusts are injected. The controller has to reject the effect of the disturbance on the regulated variables. The scenario is evaluated with several roll rates.

It shall be noted that for the ELV controllers the outer loop in position and velocity is connected on this scenario. This is a regulation scenario where we want to keep the trajectory (θ , ψ and lateral velocities and positions near) to zero.

First let's analyze the simulation at zero roll rate (Figure 5-46). The presence of lateral winds (received in component Y) motivates an appreciable deviation on θ (pitch plane) for the KELVGF and KELVF controllers. (Note that the curves are identical for these 2 controllers). The deviation is of almost 1.5 degrees in θ and 1 deg/s in r . In turn, this provokes a lateral deviation of almost 150 m in y and of 25 m/s in $yDot$. The KHINFSTRUCT and KHINFGST controller instead is able to counteract the wind disturbances: maximum error on θ is about 0.15 deg and 0.5 deg/s in r . Maximum lateral deviation is 50 m in y and of 5 m/s in $yDot$.

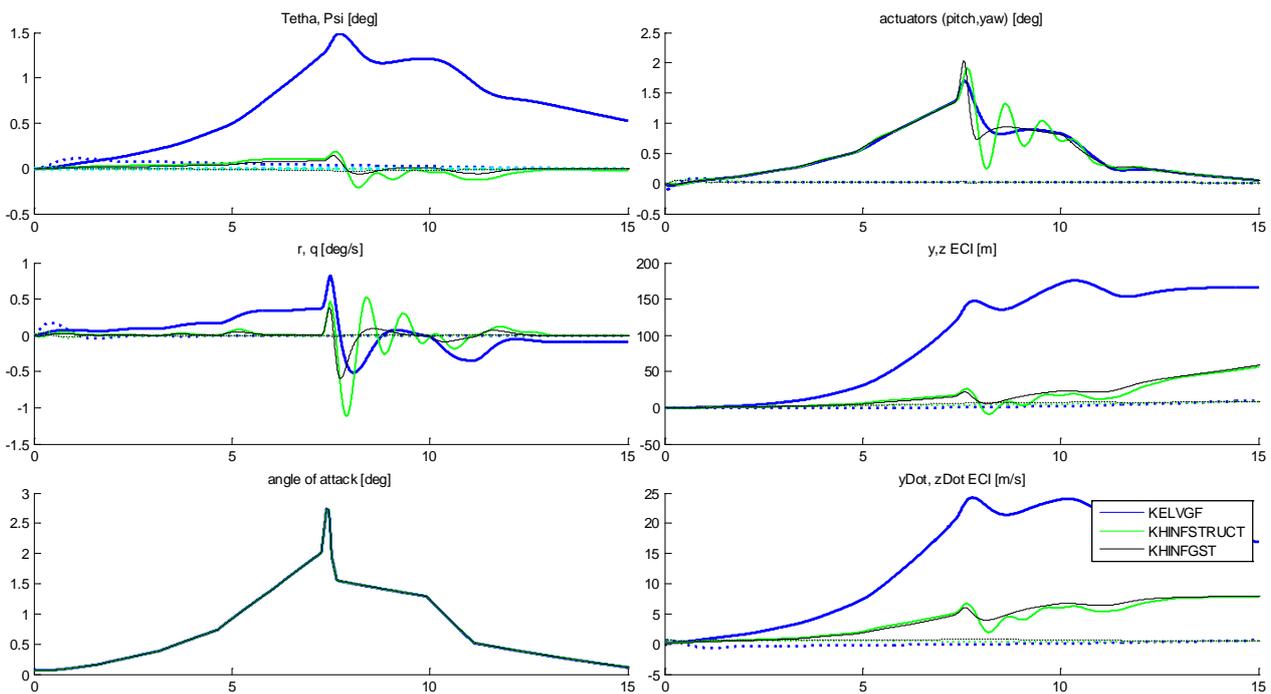


Figure 5-46: Scenario 3 roll rate $p = 0$ deg/s

At roll rate 10 deg/s the situation changes (Figure 5-47). The LV rotates 90 deg on 9 seconds, so the perturbation moves from the pitch plane to the yaw plane. For the KELVGF and KELVF controllers the error in theta is 1 deg/s at $t = 8$ s and then inverts. At the end of the simulation the lateral deviation is about 50 m in y and 150 m in z . The velocity error is 5 m/s in $yDot$ and 15 m/s in $zDot$.

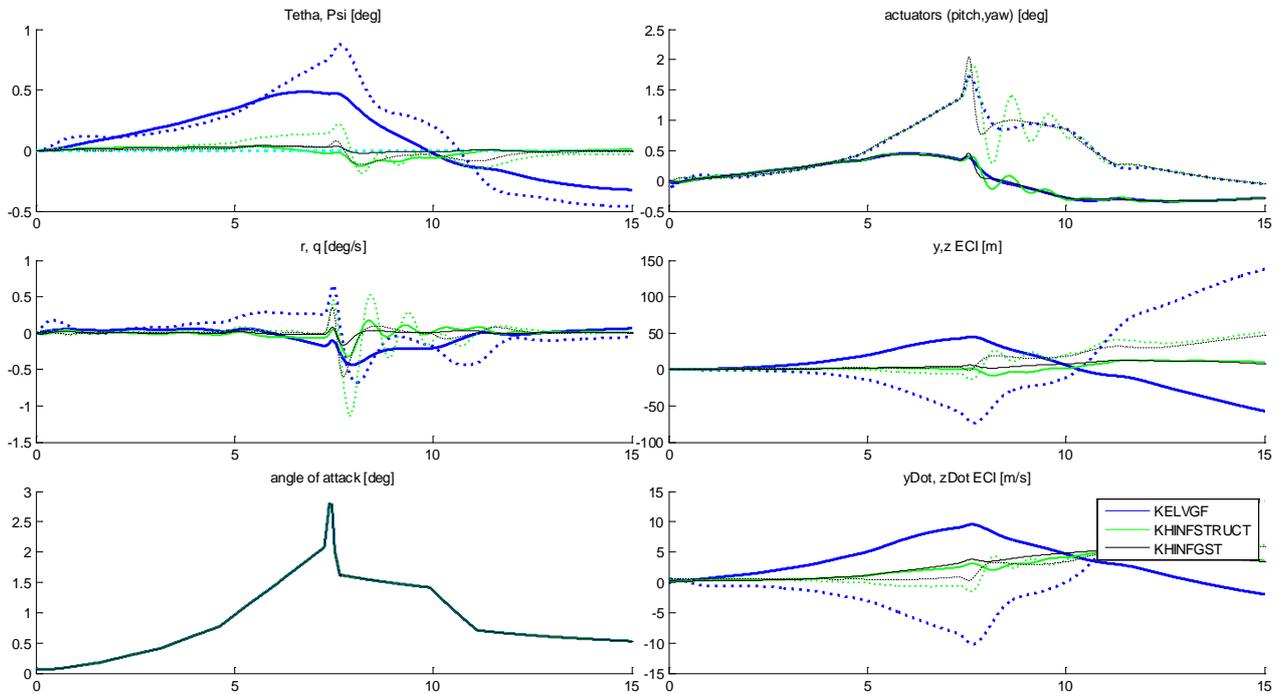


Figure 5-47: Scenario 3 roll rate $p = 10$ deg/s

At roll rate 45 deg/s the LV rotates 360 deg on 8 seconds. This creates oscillations on θ and ψ being for the KELVGF and KELVF controllers the maximum deviations of about 1 deg at the time of the wind gust. The maximum lateral deviation is about 100 m and the maximum lateral velocity deviation of about 14 m/s (Figure 5-48).

There is no significant difference on the KELVGF and KELVF controller because the gyro compensation can compensate the roll in the rotational dynamics but not on the lateral dynamics: i.e. the wind does not increase or modifies the roll rate.

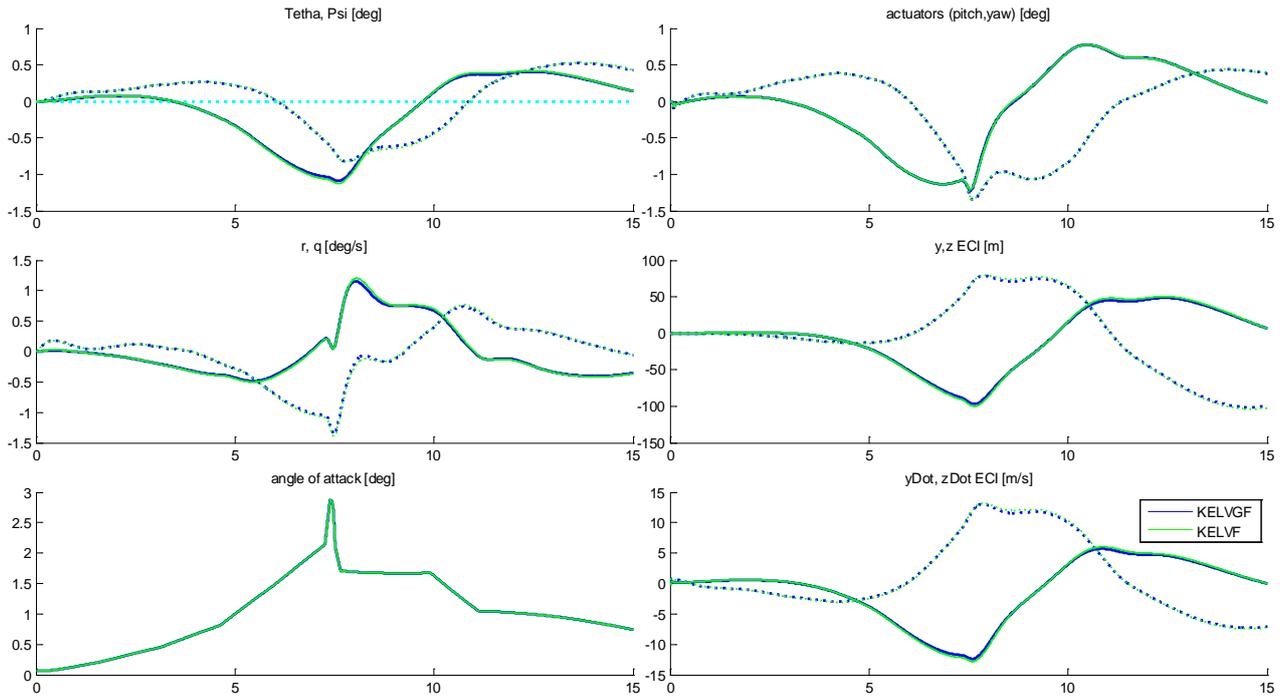


Figure 5-48: Scenario 3 roll rate at $p = 45$ deg/s (ELV controllers)

The KHINFSTRUCT and KHINFGST controllers have better behavior than the previous controllers being the KHINFGST controller the best (Figure 5-49).

The KHINFSTRUCT has a maximum deviation in θ or ψ of about 0.33 deg. The maximum lateral deviation is about 40 m and the maximum lateral velocity deviation of about 6 m/s at around the wind gust. The KHINFSTRUCT has a maximum deviation in θ or ψ of about 0.33 deg.

The KHINFGST has a maximum deviation in θ or ψ of about 0.1 deg. The maximum lateral deviation is about 30 m and the maximum lateral velocity deviation of about 3.5 m/s at around the wind gust.

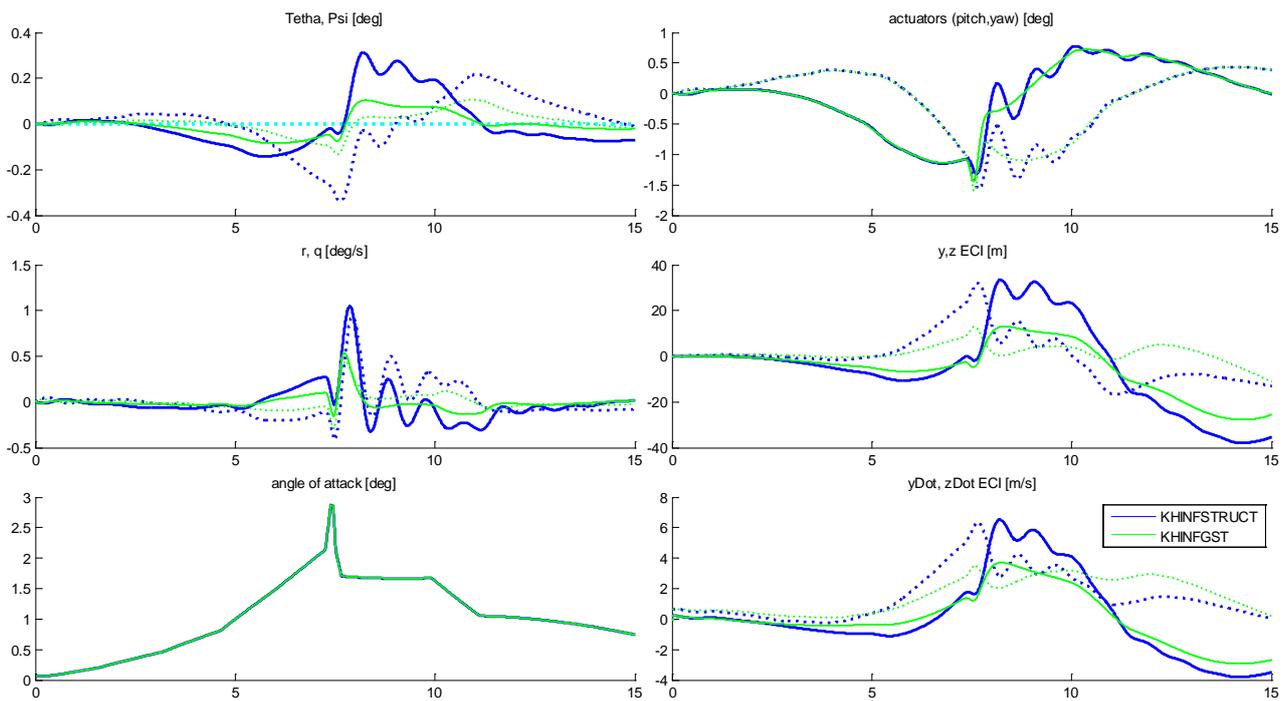


Figure 5-49: Scenario 3 roll rate at $p = 45$ deg/s (H-Infinity controllers)

The angle of attack is almost identical for all the controllers. The actuator range (deflection and deflection rate) are respected in all the cases.

5.5.5 Comparison of controllers

Two satisfactory H-Infinity controllers have been developed for the first stage of the VEGA LV. The first one “KHINFGST” uses the CHGE technique. The second one “KHINFSTRUCT” used the H-Infinity Structured technique.

Both controllers perform better than the real controllers in presence of roll rate and lateral winds without penalizing the angle of attack. The cause is that the ‘knowledge’ about how the roll rate affects the translational and rotational dynamics is present on the MIMO open loop plant. The original KELV controllers are instead basically SISO controllers with a gyroscopic compensation term.

The original KELV gyro compensation algorithm can only limit the roll rate effects on the rotational dynamics. It “does not see” the effects of the roll on the translational dynamics. Due to this cannot compensate deviations due to lateral wind. This is important in case of steady lateral winds.

The benefits of the H-infinity controllers are present without introducing penalties in the angle of attack criteria that is almost identical for all the controllers.

Of the H-infinity controllers each one has some advantages and disadvantages:

- The KHINFGST has performances better than the KINFSTRUCT.
- The KHINFSTRUCT has of less order (6 states vs. 18 states) that means less CPU time
- The KHINFSTRUCT has a well-known structure (decoupler plus PIDs plus roll-off filter). This means that could be much easier to tune by slightly changing the PID parameters if needed. This fact could help the new technology to be accepted by ‘classical GNC engineers’.
- A drawback of the KHINFSTRUCT algorithm is that being based in a random search of a local minimum, it could obtain different results in different synthesis even if nothing has been changed. This is a point that surprises when found by first time. In the practicum it can be solved by indicating the number of “restarts” of the algorithm (4 restarts was used)

The synthesized H-infinity controllers are not ‘qualified’ controllers in the aerospace sense. For achieving this goal the controllers shall be tested on the full flight domain (full scatterings, different payloads, etc.). The bending modes cannot be neglected. This is a work to be done. However results seem quite promising.

6 Contributions, Overall Conclusions and Future Work

6.1 Contributions

The main contributions of this thesis are:

- Demonstrates that H-Infinity is a valid method for controlling of ill-conditioned plants when the appropriate H-infinity “variants” are used. This is done with an academic example (the CDC benchmark) and one real life example (VEGA Launcher on first state). It is demonstrated that some remarks in literature claiming inadequacy of the H-Infinity method in general for controlling ill-conditioned plants are not exact: only the H-Infinity *Mixed Sensitivity* is affected.
- Provides an in depth comparison in terms of performances, robustness and design effort of the different H-Infinity improvements to cope with ill conditioned plants. In particular an exhaustive study of the CDC problem is performed. Some μ controllers are included also as reference.
- Applies the previous results to a real world aerospace problem, the VEGA Launcher using two of the H-Infinity optimizations (Christeen-Geering technique and Structured technique). In our knowledge it is the first time that the H-infinity Structured technique is applied to a Launcher. The synthesized controllers are cross compared in the most critical point of the first stage (maximum dynamic pressure). The real controller used in the VEGA Launcher (classical PIDs plus filters) is included in the comparison. The comparison is done with a representative nonlinear simulator of the LV named “miniVEGA” developed at hoc from scratch for this thesis. The results are satisfactory.
- Provides a good compendium (with value as tutorial) of the optimization approach to H-Infinity (LMI solution). This topic is relatively new. The existing papers are usually highly mathematical and complex. The chapter 3 can be used as base for a course or presentation in LMI techniques applied to H-Infinity theory.

6.2 Overall Conclusions

We offer here the overall conclusions of this thesis.

First conclusion is theoretical: the H-Infinity method has a solid theoretical background is very mature and the new LMI solution has been given “new life” to it because introduces a wider approach to new control problems (multi-objective minimization, LPV).

This has been described in chapter 3 where the theoretical background of the H-Infinity theory has been introduced, the modern LMI solution has been studied in detail.

Second conclusion applies to one of the objectives of the thesis: the possibility of using H-Infinity theory for ill conditioned control plants is confirmed, providing the appropriate variant is adopted.

Two interesting examples of ill conditioned plants have been selected: one academic problem (CDC distillation process) and one real world problem (Control of the first stage of the VEGA Launcher in presence of roll).

Third conclusion applies to the particular field of aerospace.

On chapter 5 two controllers based on the H-Infinity CHGE method and on the H-Infinity Structured approaches have been developed. Both controllers compare favorably with the original control used in VEGA (PID plus gyroscopic gain) in term of robustness to roll rate and automation of design.

The H-infinity Structured technique seems very suitable for aerospace applications (in particular space applications) because allows the GNC engineer to impose a predefined control structure (and this is important because the aerospace world is very conservative and reluctant to adopt revolutionary changes). The resulting controller can be further tuned manually by the engineer. This is important because allows the experienced GNC engineer to apply its know-how and provides the feeling of “mastering the design”. The order of the designed controller is chosen a priori. This allows designing low order controllers that for the space world is still an issue due to the low computational power of “radiation hardened” computers.

6.3 Future work

We mention two lines of investigations as future work: the control of the VEGA LV with inclusion of the flexible dynamics (bending modes) and the synthesis of a LPV controller.

Inclusion of the VEGA LV flexible dynamics

The classical way to take into account the flexible body dynamics is to manage it separately from the rigid body. The frequency of the bending modes is calculated by FEM methods (Finite Elements Methods). A band-pass filter is designed and added to the output of the rigid body controller. In principle this filter should be independent of the rigid dynamics and could be used with any controller designed for the rigid body dynamics.

The problem with this approach is that the rigid and flexible body dynamics are not totally independent. What happens in practice is that the GNC team designs first the rigid body controller and after the flexible body controller. When the controllers are put together there are some interactions that force to re-design and tune each one.

Adopting an H-Infinity approach for a LV 6DoF model containing both rigid and flexible dynamics would allow designing the controller in a single step, with the advantage that the optimization would be done globally for both dynamics and in a unique step.

Connections to Linear Parameter Varying (LPV)

A rocket is a paradigmatic example of a Linear Parameter Varying plant because the thrust, inertia and mass change during the trajectory as the propellant is exhausted. The control algorithms used in VEGA uses gain scheduling of classical PIDs. There are 12 stationary points along the first stage (that implies 12 manual tunings). The flight SW executes a smooth interpolation between each 2 design points.

The LPV technique is a good candidate for improving the design cycle: it could guarantee robustness by design, reduce cost and effort without penalizing the performances.

7 Bibliography

7.1 List of references

- (Apkarian, 2006) P. Apkarian and D. Noll.
Nonsmooth H-Infinity synthesis.
IEEE Trans. on Automatic Control, 51(1): 71-86, 2006.
- (Aranda, 2000) Diseño de un controlador H_∞ para un piloto automático de un avión comercial
Joaquín Aranda y David Sánchez
Actas XXI Jornadas de Automática. De. F.R. Rubio. Septiembre 2000. ISBN: 84-699-3163-6
- (Balas, 2012) Gary J. Balas, Andrew K. Packard, Peter J. Seiler
Robust Control Short Course for Space Systems
Course at ESTEC, European Space Agency, 2012
- (Byrnes, 1991) Christopher I. Byrnes
New Methods for non-Optimal Control
European Control Conference 1991, vol 3.
- (Cao, 1997) Liyu Cao, Yoichi Hori.
Mixed Sensitivity Optimization to avoid pole-zero cancellation.
Automatica, vol 33 num 7, 1997
- (Christen, 1997) Urs Christen, Hans. P. Geering.
Inverting and non Inverting H-infinity Controllers
System and Control Letters 30, 1997 31-38.
- (Churchill, 1986) R. V. Churchill, J. W. Brown
Variable Compleja y Aplicaciones
Mc Graw Hill (Text book)
- (Cruciani, 2008) Cruciani, I.
Stabilita di un Lanciatore Roll Free. Ph. D. Thesis. 2008.
Universita de la Sapienza. Rome
- (Datta, 2004) Numerical Methods for Linear Control Systems
Biswa Nath Datta (Text Book)
Elsevier 2004
- (Doyle, 1984) Doyle, J.C., "Lecture notes in advances in multivariable control,"
ONR/Honeywell Workshop, Minneapolis, 1984.

Bibliography

- (Doyle, 1989) Doyle, J.C., K. Glover, P. Khargonekar, and B. Francis,
State-space solutions to standard H₂ and H₁ control problems
IEEE Transactions on Automatic Control, vol. AC-34, no. 8, pp.
831-847, August 1989.
- (Dullerud, 2005) Dullerud, Geir E., Paganini, Fernando,
A Course in Robust Control Theory: a convex approach
Springer (Text Book)
- (Gahinet, 1994) P. Gahinet and P. Apkarian
A linear matrix inequality approach to H_∞ control, control
Int. Journal of Robust and Nonlinear Control, Vol. 4, pp 421-448,
1994.
- (Gahinet, 2011) Pascal Gahinet and Pierre Apkarian.
Decentralized and Fixed-Structure H_∞ Control in MATLAB
Decision and Control and European Control Conference (CDC-
ECC), 2011
- (Glover, 1984) Glover, K.
All optimal Hankel-norm approximations of linear multivariable
systems and their L₁ error bounds. International J. of Control, vol.
39, pp. 1115-1193, 1984.
- (Greensite, 1970) Greensite, A.L (1970).
Analysis and Design of Space Vehicle Flight Control Systems.
Spartan Books, New York.
- (Kwakernaak, 1993) Robust Control and H-Infinity optimization. Tutorial Paper.
Automatica, 29, 1993
- (Limebeer, 1991) Limebeer.
The Specification and Purpose of a Controller Design Case Study.
Proceedings of the 30th Conference on Decision and Control
Brighton, England December 1991
- (Lundstrom, 1999) Petter Lundstrom, Sigurd Skogestad and John C. Doyle.
Two-Degree-of-Freedom Controller Design for an Ill-Conditioned
Plant using Mu synthesis.
IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,
VOL. 7, NO. 1, JANUARY 1999
- (Maciejowski, 1989) Multivariable Feedback Design
(Text book)
- (Mohammadpour, 2012) Mohammadpour, Javad, Scherer, Carsten (Editors)
Control of Linear Parameter Varying Systems with Applications

- Springer, 2012
- (Roux, 2007) Roux, C. and Cruciani, I.
Roll Couplings Effects on the Stability Margins for VEGA Launcher.
AIAA Atmospheric Flight Mechanics Conference and Exhibit (2007).
- (Skogestad, 1996) Skogestad and Postlethwaite.
Multivariable Feedback Systems (Text book)
Wiley and Sons 1996
- (Scherer, 1997) Carsten Scherer, Pascal Gahinet, and Mahmoud Chilali
Multiobjective Output-Feedback Control via LMI Optimization
IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 42, NO. 7,
JULY 1997
- (Sanchez, 2013) David Sánchez de la Llana, Samir Bennani, Irene Cruciani,
Joaquín Aranda
H-Infinity control of the VEGA Launch Vehicle first stage in
presence of roll
ACA 2013. 19th IFAC Symposium on Automatic Control in
Aerospace
Wurzberg, Germany
- (Sanchez, 2015) David Sánchez de la Llana, Joaquín Aranda
Revisita al control de plantas mal condicionadas con H_{∞}
"Estructurado".
Revista Iberoamericana de Automática e Informática industrial
(RIAI). (Article submission accepted March 2015)
- (Skogestad, 1988) Skogestad, Morari, Doyle.
Robust control of ill-conditioned plants: high purity distillation.
IEEE Transactions on Automatic Control. VOL N12 December 1988
- (VanAntwerp, 2000) Jeremy G. VanAntwerp, Richard D. Braatz
A tutorial on linear and bilinear matrix inequalities
Journal of Process Control 10 (2000) 363-385
- (Zames, 1981) Zames, G.
Feedback and optimal sensitivity: model reference transformations,
multiplicative seminorms, and approximate inverses.
IEEE Transactions on Automatic Control, vol. AC-26, pp. 301-320,
1981.
- (Zhou, 1995) Robust and Optimal Control
(Text book)

7.2 Comments on bibliography

There are three classic books for the student of the theory of Robust Control.

The book “Multivariable Feedback Design” (Maciejowski, 1989) is a classical reference. Only inconvenient is that is becoming outdated. The demonstration of the state space solution shown in the book is the 1984 solution.

The book Multivariable Feedback Control (Skogestad, 1996) is also a classical reference. It is an excellent reference book because it contains in addition to the theory a lot of examples and practical experience and know-how.

The book “Robust and Optimal Control” (Zhou, 1995) is totally different from the previous two. It is a highly mathematical book very useful as reference but not valid as introductory text due to its complexity.

Internet has changed the way we work: at today there are thousands of links with very high quality electronics books, presentations and online video courses. We will try to enumerate some of the links used in this thesis.

The theory of H-Infinity of chapter 3 and supporting appendix has been elaborated from several sources between them:

- The “Robust Control” course of professor Masayuki Fuyita (<http://www.fl.ctrl.titech.ac.jp/course/ROC>)
- The “Multivariable Feedback Control” course of professor Masayuki Fuyita (<http://www.sites.mech.ubc.ca/~nagamune>)
- The course (Balas, 2012) presented by A. Packard and P. Seiler

The modern convex approach to robust control and in particular the LMI solution described in chapter 3 has been elaborated from several sources between them

- The very interesting presentations of professor Scherer (<http://www.dcsc.tudelft.nl/~cscherer/lmi.html>) (now in the university of Stuttgart)
- The “Robust Multivariable Control” presentations of professor Anders Helmersson (<http://users.isy.liu.se/en/rt/andersh/teaching/robkurs.html>)
- The book “A Course in Robust Control Theory: a convex approach” (Dullerud, 2005)
- The slides of a presentation in university of Calabria by professor Apkarian (<http://pierre.apkarian.free.fr/COURS/CALABRIAseminar.pdf>)

For chapter 5, the classical reference for deriving the equations of a rocket is (Greensite, 1970) . A quite complete and compressive reference specific to the VEGA Launcher is the PhD (Cruciani, 2008). The 6DoF equations of chapter 5 and supporting appendix are mainly elaborated from this reference but with ideas taken from several courses between them:

Bibliography

- The MIT course “Aerospace Dynamics” (<http://ocw.mit.edu/courses/aeronautics-and-astronautics/16-61-aerospace-dynamics-spring-2003>)
- The MIT course “Dynamics” (<http://ocw.mit.edu/courses/aeronautics-and-astronautics/16-07-dynamics-fall-2009>)
- For the “transport theorem” the course “Robot Kinematic and Dynamic” (<http://www.control.aau.dk/~jan/undervisning/MechanicsI/Robot-ST8/mechbook.pdf>)
- For the quaternion formalism, the main source has been the website (<http://www.euclideanspace.com/math/algebra/realNormedAlgebra/quaternions/index.htm>)

The referenced MIT courses are part of the fantastic Open CourseWare MIT website (<http://ocw.mit.edu/index.htm>).

8 Appendix: Demonstrations

8.1 Mathematical background

Some mathematical notes are recalled in this appendix.

8.1.1 Some notes on complex functions

8.1.1.1 Cauchy Riemann equations

A complex function $f(z)$ where z is a complex variable ($z = x + iy$) can be written as made of two parts:

$$f(z) = u(x, y) + i v(x, y)$$

The derivative is defined by analogy with real functions as:

$$\frac{d(f(z))}{dz} = \lim_{\Delta z \rightarrow 0} \left(\frac{f(z + \Delta z) - f(z)}{\Delta z} \right)$$

With real functions, the existence of the derivative means that the function is *smooth*. In complex functions, it implies also that the derivative is the same in all the approaching directions to the point.

$$\frac{\partial f(z)}{\partial x} = \frac{\partial f(z)}{\partial(iy)}$$

$$\frac{\partial f(x + iy)}{\partial x} = \frac{\partial f(x + iy)}{\partial(iy)}$$

Applying the usual rules of differentiation:

$$\frac{\partial u(x, y)}{\partial x} + i \frac{\partial v(x, y)}{\partial x} = \frac{\partial u(x, y)}{\partial(iy)} + i \frac{\partial v(x, y)}{\partial(iy)}$$

$$\frac{\partial u(x, y)}{\partial x} + i \frac{\partial v(x, y)}{\partial x} = \frac{\partial u(x, y)}{i \partial(y)} + i \frac{\partial v(x, y)}{i \partial(y)}$$

$$\frac{\partial u(x, y)}{\partial x} + i \frac{\partial v(x, y)}{\partial x} = -i \frac{\partial u(x, y)}{\partial(y)} + \frac{\partial v(x, y)}{\partial(y)}$$

Equating real and imaginary parts the Cauchy-Riemann equations are deduced.

$$\frac{\partial u(x, y)}{\partial x} = \frac{\partial v(x, y)}{\partial y} \quad \frac{\partial v(x, y)}{\partial x} = -\frac{\partial u(x, y)}{\partial y} \quad (\text{Eq 8-1})$$

A function is called analytic if the Cauchy-Riemann equations are fulfilled.

Using the notation:

$$\frac{\partial u(x, y)}{\partial x} = u_{xx}$$

Taking the derivative of the Cauchy-Riemann equations a second time on x:

$$u_{xx} = v_{yx} \quad v_{xx} = -u_{yx}$$

Taking the derivative a second time on y:

$$u_{xy} = v_{yy} \quad v_{xy} = -u_{yy}$$

Using $u_{xy} = u_{yx}$, $v_{xy} = v_{yx}$:

$$u_{xx} = v_{yx} = -u_{yy} \quad v_{xx} = -u_{yx} = -v_{yy}$$

A consequence of Cauchy-Riemann equations is that for complex analytic functions, there cannot be a point z_0 where the second derivative on all the directions has the same sign (except when derivative is zero, i.e. function is a constant). This means that for complex functions there are not maximums or minimums but saddle points. This fact will be used later for demonstrating the “Maximum Modulus Theorem”.

A representation of the complex function $f(z) = z^2$ follows (Figure 8-1). The magnitude of the function is the real part of $f(z)$. The imaginary part of $f(z)$ would be the phase of the small vector (made of the real part and imaginary part of $f(z)$). The function has a minimum along the real axis but a maximum along the imaginary axis, i.e. a saddle point in zero.

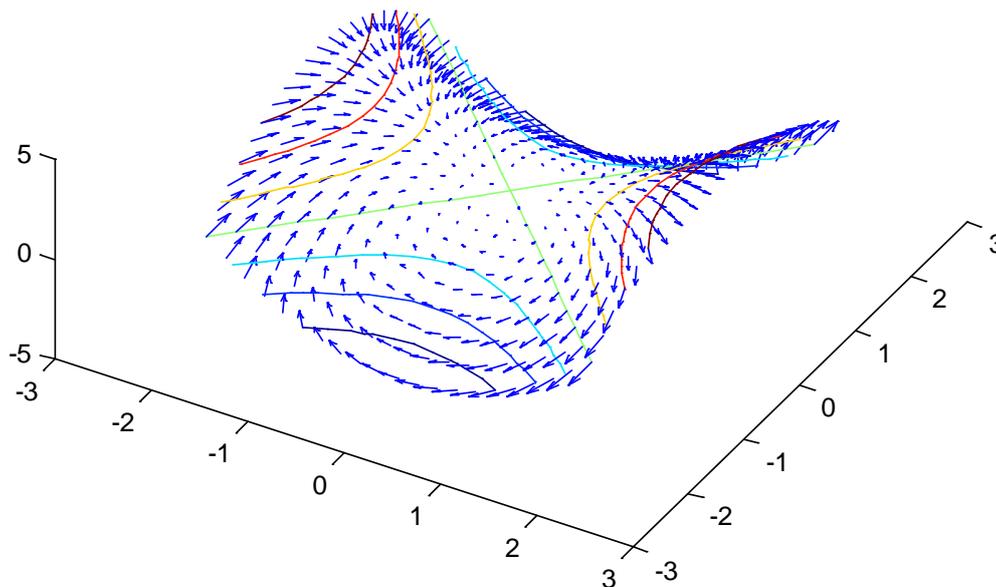


Figure 8-1: The complex function z^2

8.1.1.2 Cauchy Goursat theorem

Let be $f(z)$ an analytic function. Be C a closed contour. Then

$$\int_C f(z)dz = 0$$

We will not perform a rigorous demonstration but instead show an intuition of the demonstration. A formal demonstration can be found in (Churchill, 1986).

This intuitive demonstration requires a number of steps.

Analogies with real functions

Note that the concept of integration through a closed contour can be generalized from the integration through a closed path in the real domain. First, the integral of the real function $f(x)$ between A and B is:

$$\int_A^B f(x)dx = F(B) - F(A)$$

And so:

$$\int_B^A f(x)dx = F(A) - F(B) = -\int_A^B f(x)dx$$

So in the real domain the integral over a path that starts in A , goes to B then back to A is equal to zero. Note also that the integral along an interval that goes from A to B through an arbitrary list of intermediate steps (not necessarily located between A and B) is exactly equal to the integral along the direct path A to B (all the back and forth steps cancels).

Definition of complex integral along an arbitrary path

This can be generalized to the complex domain. Given a variable z and a complex function of the variable z , both the variable and the function can be expressed as a real part and an imaginary part:

$$z = x + iy$$

$$f(z) = u(x, y) + i v(x, y)$$

Where $x, y \in \mathbb{R}$ and $u(x, y)$ and $v(x, y)$ are real valued functions. In other words the function $f(z)$ can be considered made of two real functions of two variables. This fact made easy to introduce the concept of derivative, integral, etc. for imaginary functions. Just the desired operation is applied to each part separately.

When the variable z is evaluated about a path C (being the path C parametrized by a parameter t , i.e. $C = C(t)$) we have:

$$z(t) = x(t) + iy(t)$$

$$f(z(t)) = u(x(t), y(t)) + i v(x(t), y(t))$$

Then the integral through a path C from A to B in the complex plane is defined as:

- Partition the path [A, B] in n sub-intervals $[t_i, t_{i+1}]$, being $\Delta t = t_{i+1} - t_i$
- Define the distance between 2 points in the path: $\Delta s_i = z(t_{i+1}) - z(t_i) = z(t_{i+1} + \Delta t) - z(t_i)$
- Evaluate the Riemann sum in the usual way

$$S = \lim_{\Delta s_i \rightarrow 0} \sum_{i=1}^n f(z(t_i)) \Delta s_i$$

But as:

$$\Delta s_i = z(t_{i+1} + \Delta t) - z(t_i) \approx z'(t) \Delta t$$

With this we have:

$$S = \lim_{\Delta t \rightarrow 0} \sum_{i=1}^n f(z(t_i)) z'(t) \Delta t$$

And passing to the limit

$$S = \int_A^B f(z(t)) z'(t) dt$$

Let's study some simple paths in the complex plane.

Integral of a complex function along a path on the real axis

First case is a path from A to B where both A and B are real numbers. This case is exactly the real integral along the real axis. So:

$$S = \int_A^B f(z) dz = \int_A^B f(z(t)) z'(t) dt$$

As along the real axis: $z(t) = x(t) + i0 = x(t)$

$$S = \int_A^B f(z(t)) x'(t) dt = \int_A^B (u(x(t), 0) + iv(x(t), 0)) x'(t) dt = \int_A^B u(x(t)) x'(t) + i \int_A^B v(x(t)) x'(t)$$

Where the integral is splits on two real integrals of a real valued function.

If the path is from B to A along the real axis we know that:

$$S = \int_A^B f(z) dz = - \int_B^A f(z) dz$$

Integral of a complex function along a path on the imaginary axis

Second case is a path from A to B where both A and B are pure imaginary numbers.

As along the imaginary axis: $z(t) = 0 + iy(t) = y(t)$

$$S = \int_A^B f(z(t))x'(t)dt = \int_A^B (u(0, y(t)) + iv(0, y(t)))y'(t)dt = i \int_A^B u(y(t))y'(t) - \int_A^B v(y(t))y'(t)$$

Where the integral is splits on two real integrals of a real valued function.

If the path is from B to A:

$$S = \int_B^A f(z(t))x'(t)dt = \int_B^A (u(0, y(t)) + iv(0, y(t)))y'(t)dt = i \int_B^A u(y(t))y'(t) - \int_B^A v(y(t))y'(t)$$

Again we know that for real integrals if the limit are exchanged the integral changes of sign:

$$S = -i \int_A^B u(y(t))y'(t) + \int_A^B v(y(t))y'(t)$$

So also for complex functions, for an integration path in the imaginary axis:

$$S = \int_A^B f(z)dz = - \int_B^A f(z)dz$$

From this result, we can see that the integral along a closed horizontal or vertical path A to B and back to A is zero because the integral in a direction and the opposite cancels each other (see the analogy with real functions in the real axis).

Then, any closed path compound of straight segments can be approximated by a closed path that duplicates each segment in opposite direction (Figure 8-2). Each segment cancels with the opposite as previously. We can pass from the left path to the right path in a continuous differentiable process without cutting / pasting.

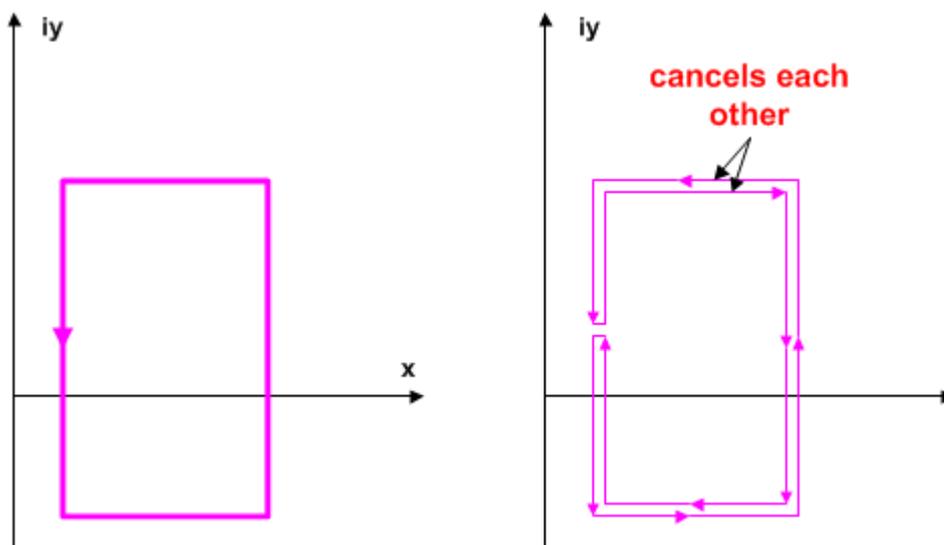


Figure 8-2: Complex integral along a rectangular path

In turn any closed arbitrary path (not straight) could be approximated by a double path and in turn could be approximated by a connected set of small 'straight paths' (Figure 8-3).

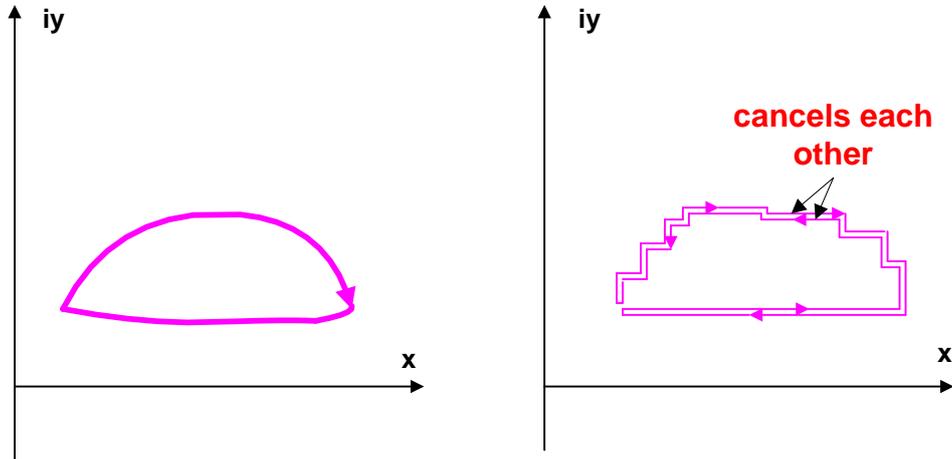


Figure 8-3: Complex integral along an arbitrary closed path

So we have demonstrated (informally) that the integral of a complex analytic function along a closed path C is zero because the path can be approximated by a number of small double paths that cancels each other.

Path independence of the integral between 2 points

Finally, we show (also intuitively) that the integral along 2 points A and B on the complex plane is independent of the path chosen.

In (Figure 8-4), two paths are used. Each deviation of path P_2 from path P_1 can be deformed continuously to be a closed subpath, that we know is zero.

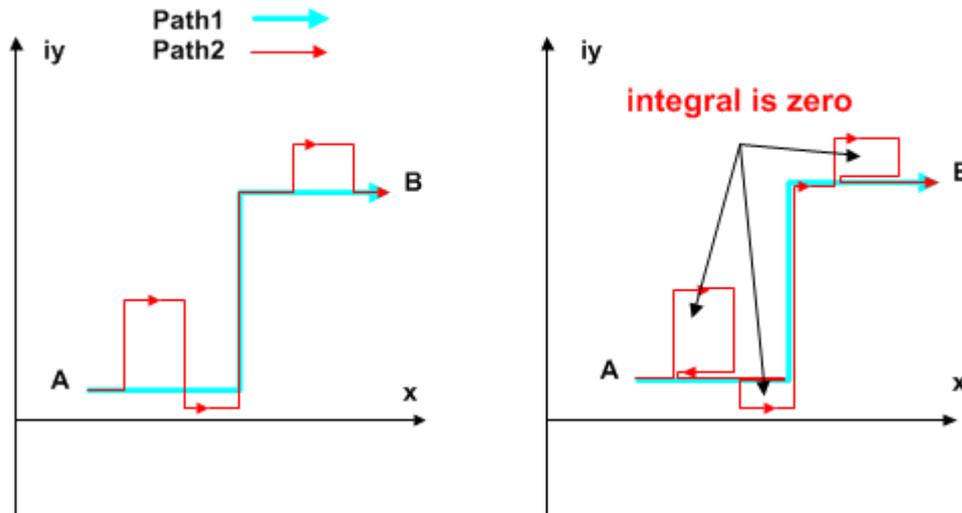


Figure 8-4: Path independence of complex integrals

8.1.1.3 Laplace transform

The Laplace transform is defined as:

$$F(s) = \int_0^{\infty} f(t)e^{-st} dt$$

The Laplace transform is a lineal operator: we take the original function and multiply by the exponential function on each point.

Example: Let's be $f(t) = e^{-at}u(t)$, where $u(t)$ is the step function (0 if $t < 0$, 1 if $t \geq 0$) and 'a' is a positive constant. Then

$$F(s) = \int_0^{\infty} f(t)e^{-st} dt = \int_0^{\infty} e^{-at}u(t)e^{-st} dt = \int_0^{\infty} e^{-(a+s)t}u(t)dt$$

$$F(s) = -\frac{1}{(s+a)} \left(e^{-(a+s)t} \Big|_{t=\infty} - e^{-(a+s)t} \Big|_{t=0} \right) = -\frac{1}{(s+a)}(0-1) = \frac{1}{(s+a)}$$

That is stable iff $s > -a$.

Note that functions as $f(t) = 1$, $f(t)=z^n$, that have not defined real integral in $[0, \infty]$ has a perfectly defined Laplace transform. Note that the Laplace transform is defined for any functions that growth slower than e^{-at} .

The Laplace transform changes derivative into multiplication by s and integral into division by s .

$$L(f'(t)) \rightarrow sF(s) - f(0)$$

$$L\left(\int f(t)\right) \rightarrow \frac{1}{s}F(s)$$

Example: the Laplace transform of the derivative is:

$$F(s) = \int_0^{\infty} \frac{df}{dt}(t) e^{-st} dt$$

As the derivative of a product is:

$$(fv)' = f'v + fv'$$

$$f'v = (fv)' - fv'$$

$$\int f'v = \int ((fv)' - fv')$$

$$\int f'v = fv - \int fv'$$

Identifying

$$f'v \equiv \frac{df}{dt}(t) e^{-st}$$

We have:

$$F(s) = \int_0^{\infty} \frac{df}{dt}(t) e^{-st} dt = f(t) e^{-st} \Big|_0^{\infty} - \int_0^{\infty} f(t) (-s) e^{-st} dt = -f(0) + s \underbrace{\int_0^{\infty} f(t) e^{-st} dt}_{F(s)}$$

8.1.2 Introduction to norms

A norm is a mathematical operation that assigns a scalar to each element of a space. In order to be a norm, the following properties shall be observed:

- Non negative: $norm(v) \geq 0$
- Positivity: $norm(v) \text{ iff } v = 0$
- Homogeneity: $norm(a v) = a norm(v)$ for any scalar a
- Triangle inequality: $norm(e1 + e2) \leq norm(e1) + norm(e2)$

It is precise to introduce the following concepts: norm of a vector, norm of a matrix, norm of a temporal signal, norm of a system.

Norm of a vector

There are several types of vector norms.

- 1-norm for a vector is the sum of the absolute value of the elements
- 2-norm for a vector is the square root of the sum of the squares of the (absolute value of the elements). This is the usual 'Euclidean norm'.
- Infinity-norm for a vector is the maximum of the absolute value of the components

Induced norm of a matrix

The norm of a matrix could have been defined in a similar way that for a vector (i.e. the norm only depends on the elements of the matrix). It is however much more useful to introduce the concept of “*induced matrix norm by a vector*”, because it has a connection with the classical concept of ‘gain’.

Given:

$$z = A w$$

The induced norm of A by the vector norm-p is:

$$\|A\|_p \equiv \max \frac{\|Aw\|_p}{\|w\|_p}$$

i.e. the induced norm is max ratio between the norm of the vector as transformed by the matrix divided by the norm of the vector.

The induced vector norm of a matrix satisfies the triangle inequality:

- Norm (A B) <= norm(A) norm(B)

Norm of a signal that varies along time or with frequency

Given a signal $e(t)$ or $e(s)$, we can define the signal norm in different ways:

- Take something proportional to the area of the signal (for example square root of the signal square would be a 2-norm)
- Take the maximum value over time (this is an infinity norm)

For multidimensional signals, we shall evaluate first the norm at each instant or frequency (using a vector norm) and then evaluate over time or frequency using H_2 or H_∞ norm.

The choice of the vector norm made the criteria more or less conservative. In (Figure 8-5), it can be appreciated the location of points with induced p-norm = 1 for p in (-1, -2 or $-\infty$):

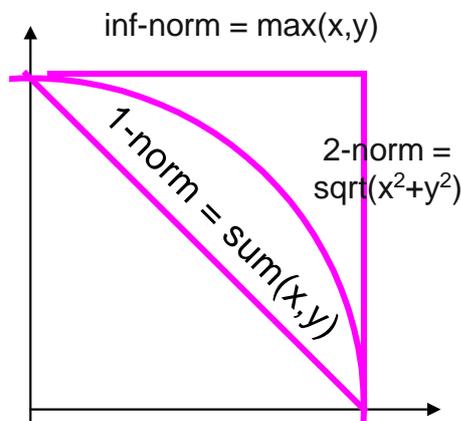


Figure 8-5: Visualization of norms

Example, the vector $u = (1, 0)$ has 1-norm = 1, 2-norm = 1, inf-norm = 1.
 The vector $u = (0.5, 0.5)$ has 1-norm = 1, 2-norm = 0.7071, inf-norm = 0.5.

Let's suppose that the criteria is to have p-norm greater than 1 (to stay far from the point (-1,0)). The point A will not satisfy the inf-norm. Point B will satisfy the 1-norm but not the 2-norm. Point C will satisfy all the norms (Figure 8-6). This means that a criteria based in 1-norm is the less conservative. A criterion based in inf-norm instead discarded valid points as "too close to the origin". Usually the 2-norm is a good compromise.

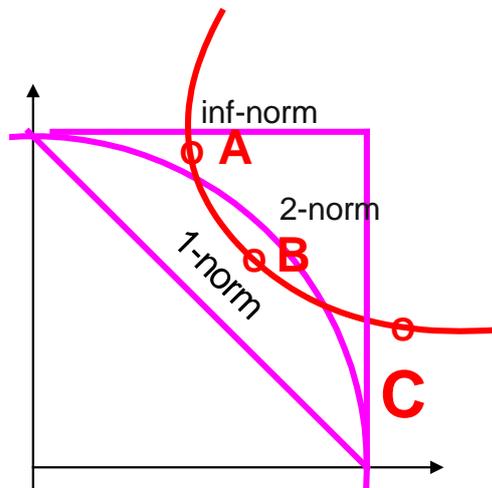


Figure 8-6: Conservatism of p-norms

Norm of a system

A SISO system is represented by a transfer function. The transfer function has a state space representation associated as matrices $[A,B,C,D]$. These matrices $[A,B,C,D]$ form a composed matrix for which we can associate a matrix norm.

A MIMO system is represented by a matrix of transfer functions. The matrix transfer function has a state space representation associated (in state space the MIMO system simply has B,C,D matrices that are not 1x1). The matrices $[A,B,C,D]$ form a matrix for which we can associate a matrix induced norm.

We have now all the elements for defining the H-Infinity norm of a system. We take the 2-induced norm (singular values) as the matrix norm evaluated at each instant of frequency and then we take the infinity-norm along the time or frequency.

H_∞ norm of a system is defined as:

$$\| \cdot \|_\infty \equiv \sup_{w \in \mathbb{R}} \overline{\sigma}(G(iw)) \equiv \sup_{\text{Re}(s) > 0} \overline{\sigma}(G(s))$$

8.1.3 Mathematical introduction to LFT

A Linear Fractional Transformation is a function of complex variable z defined by four parameters (a, b, c, d):

$$F(z) = \frac{a + bz}{c + dz}$$

A LFT can be written also as

$$F(z) = a + bz(1 - dz)^{-1}c$$

The LFT is called also the *Moebius* transformation. Doyle was the first to apply the concept to control system, replacing the variable z by a state space matrix K .

If we have a complex matrix

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

The lower linear functional transformation $F_L(M, K)$ is defined as

$$F_L(M, K) = M_{11} + M_{12}K(1 - M_{22}K)^{-1}M_{21} \quad (\text{Eq 8-2})$$

And represents the system (Figure 8-7)

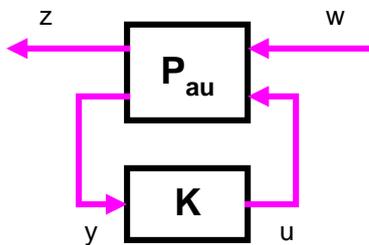


Figure 8-7: Lower LFT

The upper linear functional transformation $F_U(M, K)$ is defined as:

$$F_U(M, K) = M_{22} + M_{21}K(1 - M_{11}K)^{-1}M_{12} \quad (\text{Eq 8-3})$$

And represents the system (Figure 8-8)

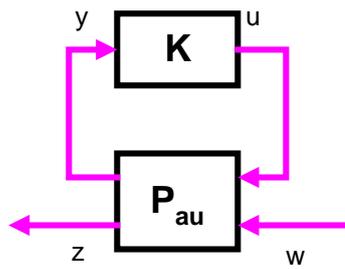


Figure 8-8: Upper LFT

A nice property of the LFTs is that composition of LFT is again a LFT.

A lot of structures used in control can be represented as LFTs.

First one is used widely in this work: to express the composition of the augmented plant and the controller as an LFT.

An uncertain parameter can be expressed as LFT.

For example, given a parameter $p \in [2.0, 2.8]$, it can be written as $p = 2.4 + 0.4d_c$ where $d_c \in [-1, 1]$. And this can be put as a LFT:

$$F_L(p, d_c) = F_L \left(\begin{bmatrix} 2.4 & 0.4 \\ 1 & 0 \end{bmatrix} \right), \quad F_U(p, d_c) = F_U \left(\begin{bmatrix} 0 & 0.8 \\ 0.5 & 2.4 \end{bmatrix} \right),$$

Applying the LFT formula:

$$F_L(M, K) = M_{11} + M_{12}K(1 - M_{22}K)^{-1}M_{21} = 2.4 + 0.4d_c(1 - 0d_c)^{-1}1 = 2.4 + 0.4d_c$$

More generally, additive or multiplicative uncertainty can be expressed as LFTs. The uncertainty can be dynamic:

Additive uncertainty

$$G(s) = G_0(s) + W(s)\Delta(s)$$

$$G(s) = F_U \left(\begin{bmatrix} 0 & 1 \\ W(s) & G_0(s) \end{bmatrix}, \Delta(s) \right)$$

Multiplicative uncertainty

$$G(s) = G_0(s)(1 + W(s)\Delta(s))$$

$$G(s) = F_U \left(\begin{bmatrix} 0 & W(s) \\ G_0(s) & G_0(s) \end{bmatrix}, \Delta(s) \right)$$

8.1.4 Kernel and null space

Given a linear map A, its kernel or null space is defined as:

$$\ker(A) = \{v \in V : A(v) = 0\}$$

Its image space is defined as:

$$\text{Im}(A) = \{w \in W : A(v) = w\}$$

The image and the kernel subspaces ‘fill’ the original space V. The image and the kernel subspaces are mutually orthogonal:

$$\dim(V) = \dim(\ker(A)) + \dim(\text{Im}(A))$$

$$\text{Im}(A)^\perp = \ker(A)$$

8.1.5 LMIs

Only a brief introduction is provided in this thesis. Refer to (VanAntwerp, 2000) for a good LMI tutorial (including a lot of references) and its role in control.

A symmetric matrix P is called positive definite “ $P \succ 0$ ” (where the special font of the “greater sign” denotes “positive definite”) if:

$$x^T P x > 0, \forall x \in R^n, x \neq 0$$

Example:

$$P = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \rightarrow x^T P x = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 2x_1^2 + x_2^2 > 0, x \neq 0$$

Facts:

- $P \succ 0 \Leftrightarrow \lambda_i(P) > 0$ (all the eigenvalues are positive)
- $P \succ 0 \Leftrightarrow \det(P(1:i, 1:i)) > 0$ (all the upper left sub matrix are positive)
- $P \succ 0 \Leftrightarrow p_{ii} > 0$ (all the diagonal elements are positive)

A Linear Matrix Inequality is an expression of the form:

$$F(x) = P_0 + x_1 P_1 + \dots + x_n P_n \succ 0 \quad \text{where } P_i \text{ are real symmetric matrices } (P = P^T)$$

A Linear Matrix Inequality is a particular case of the more general Semi Definite Programming, where it is required that the constraint $F(x)$ is affine (i.e. depends linearly in the P_i).

Examples of LMI expressions are:

$$1 + x \succ 0$$

$$1 + x_1 + 2 x_2 \succ 0$$

$$F(x) = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} + x_1 \begin{bmatrix} 2 & -5 \\ -5 & 1 \end{bmatrix} + x_2 \begin{bmatrix} 0 & 8 \\ 8 & 1 \end{bmatrix} \succ 0$$

A LMI has a number of nice properties:

A LMI defines a convex set: if x_1 and x_2 are solution of a convex problem, then any convex combination $x = (1 - \lambda) x_1 + \lambda x_2$ (with λ in $[0..1]$) is also a solution.

Note:

$$x = (1 - \lambda) x_1 + \lambda x_2 \text{ (with } \lambda \text{ in } [0..1]) \text{ is the segment between } x_1 \text{ and } x_2$$

There is an equivalent with classical real functions. To be convex we require that

$$F((1 - \lambda)x_1 + \lambda x_2) \leq (1 - \lambda)F(x_1) + \lambda F(x_2)$$

For all x_1, x_2 and λ in $[0..1]$. But that is simply to require that the value of the function is always greater than the value of the first order approximation (think on a parabola and its first order approximation).

The combination of LMIs is a LMI. Given:

$$F_1(x) > 0, F_2(x) > 0, \dots, F_n(x) > 0$$

The combined LMI is:

$$F(x) = \text{diag}\{F_1(x), F_2(x), \dots, F_n(x) > 0\} > 0$$

This allows expressing multiple control requirements as only one LMI.

A LMI has a number of applications in control. The stability of a system can be expressed as a Lyapunov equation that admits a LMI formulation. The crucial Bounded Real Lemma (studied in chapter 3) relates the size of the H-Infinity norm of a system with a LMI. A Riccati equation can be expressed as a LMI (see demonstration in 8.2.5).

8.1.6 Schur complement

Given the matrix M

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

The Schur complement of D in M is given by

$$A - BD^{-1}C \quad (\text{Eq 8-4})$$

The Schur complement is derived when the matrix M is interpreted as the set of equations:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}$$

And we express c and d in function of x and y

$$Cx + Dy = d;$$

$$y = D^{-1}(d - Cx)$$

$$Ax + By = c;$$

$$Ax + BD^{-1}(d - Cx) = c;$$

$$Ax + BD^{-1}d - BD^{-1}Cx = c;$$

$$(A - BD^{-1}C)x + BD^{-1}d = c;$$

So we arrive to:

$$x = (A - BD^{-1}C)^{-1}(c - BD^{-1}d)$$

And

$$y = D^{-1}(d - C(A - BD^{-1}C)^{-1}(c - BD^{-1}d))$$

And we have c and d in function of x and y :

$$x = (A - BD^{-1}C)^{-1}c - (A - BD^{-1}C)^{-1}BD^{-1}d$$

$$y = -D^{-1}C(A - BD^{-1}C)^{-1}c + (D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1})d$$

But this gives the formula for the inverse of M :

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} \begin{bmatrix} c \\ d \end{bmatrix}$$

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} (A - BD^{-1}C)^{-1} & -(A - BD^{-1}C)^{-1}BD^{-1} \\ -D^{-1}C(A - BD^{-1}C)^{-1} & D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1} \end{bmatrix}$$

That in turn, can be factorized as:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{pmatrix} I & 0 \\ -D^{-1}C & I \end{pmatrix} \begin{pmatrix} (A-BD^{-1}C)^{-1} & 0 \\ 0 & -D^{-1}C \end{pmatrix} \begin{pmatrix} I & -BD^{-1} \\ 0 & I \end{pmatrix}$$

And for M itself as:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{pmatrix} I & BD^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} A-BD^{-1}C & 0 \\ 0 & D \end{pmatrix} \begin{pmatrix} I & 0 \\ -D^{-1}C & I \end{pmatrix}$$

It could be argued that we have found a more complicated expression, but is not the case: we have factorized a complex compound matrix in function of diagonal or semi-diagonal matrices that are always easier to manage.

Now, imposing that M is symmetric, so A and D are symmetric and $C = B^T$:

$$M = \begin{bmatrix} A & B \\ B^T & D \end{bmatrix} = \begin{pmatrix} I & BD^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} A-BD^{-1}B^T & 0 \\ 0 & D \end{pmatrix} \begin{pmatrix} I & 0 \\ -D^{-1}B^T & I \end{pmatrix}$$

Connection of the Schur complement with a LMI:

Let's be M a symmetric matrix:

$$M = \begin{bmatrix} A & B \\ B^T & D \end{bmatrix}$$

$M \succ 0$ (definite positive) if and only iff $D \succ 0$ and the Schur complement of D in M is definite positive (i.e. $A-BD^{-1}B^T \succ 0$).

Demonstration:

Any matrix is definite positive if their determinants are all positive. We have seen that M can be decomposed as:

$$M = \begin{bmatrix} A & B \\ B^T & D \end{bmatrix} = \begin{pmatrix} I & BD^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} A-BD^{-1}B^T & 0 \\ 0 & D \end{pmatrix} \begin{pmatrix} I & 0 \\ -D^{-1}B^T & I \end{pmatrix}$$

The first factor $\begin{pmatrix} I & BD^{-1} \\ 0 & I \end{pmatrix} \succ 0$ because I is positive and $(I * I - 0 * BD^{-1} = I)$ is positive.

The same argument applies to the third factor.

So the fact of being $M \succ 0$ only depends on the second factor, with the assumed conditions $D \succ 0$ and $A-BD^{-1}B^T \succ 0$.

8.1.7 The Congruence Transformation

A congruence transformation is simply a change of base by a non-singular matrix.

It follows that if U is a non-singular matrix, it preserves the ‘definiteness’ of a LMI. The following are equivalent:

$$F \succ 0$$

$$U^T F U \succ 0$$

The congruence transformation is used in the LMI solution to the H-Infinity problem.

8.2 Demonstrations for chapter 3

8.2.1 Maximum modulus theorem

If $f(z)$ is an analytic function in a closed region C , then $f(z)$ does not achieve its maximum (or minimum) in the interior of C , except if $f(z)$ is constant.

Demonstration

If $f(z)$ is analytic $f(z)$ shall fulfill the Cauchy-Riemann conditions. But these conditions prevent the existence of maximum or minimums for $f(z)$ (because all the points where the derivative is zero are saddle points). This means that $f(z)$ achieves its maximum and minimum over the boundary of C and not in its interior.

An application to control theory is that for each $F(s)$ stable,

$$\|F(s)\|_{\infty} = \sup_{\omega \in R} |F(j\omega)| = \sup_{\text{Re}(s) \geq 0} |F(s)|$$

We can choose as closed region a rectangle (as big as wanted) with left side along the imaginary axis. We know that $F(s)$ cannot have maximum or minimums inside the rectangle and we know that $F(s) \rightarrow 0$ when $s \rightarrow \infty$. So the maximum is attained in the imaginary axis.

8.2.2 Youla Parametrization of stabilizing controllers

Given a standard feedback system, we can choose a structure for the controller that incorporates the plant P (Figure 8-9).

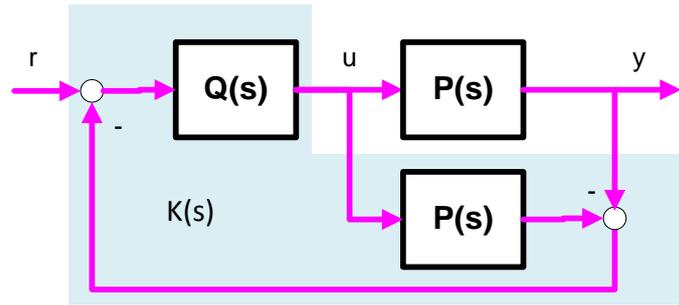


Figure 8-9: Internal model control

This is called the Youla parametrization. $Q(s)$ is a proper stable transfer function. The expression of $K(s)$ gives all the stabilizing controllers:

$$K(s) = (1 - Q(s)P(s))^{-1} Q(s)$$

8.2.3 Stability of a system with Lyapunov method

The stability of a system can be defined as follows: the LTI unforced system $x' = Ax$ is stable (i.e. the state tends to zero as time goes to infinity) if exist a Lyapunov function $V(x) = x^T Px$ with $P \succ 0$ such that the derivative is negative for all the states x (intuitively, if the 'norm' of the state decreases along time whatever the trajectory followed the state will approach to zero):

$$\begin{aligned} \frac{d(V(x))}{dt} &= \dot{x}^T Px + x^T P\dot{x} = (Ax)^T Px + x^T P(Ax) = x^T A^T Px + x^T P(Ax) = x^T (A^T P + PA)x \\ \frac{d(V(x))}{dt} &= x^T (A^T P + PA)x < 0 \\ (A^T P + PA) &< 0 \end{aligned}$$

The Lyapunov equation is a LMI:

- $P \succ 0$
- $A^T P + PA < 0 \rightarrow -A^T P - PA \succ 0$

That is usually expressed as:

$$F(x) = \begin{bmatrix} P & 0 \\ 0 & -A^T P - PA \end{bmatrix} \succ 0$$

Also can be written in the standard form as:

$$F(x) = 0 + x \begin{bmatrix} P & 0 \\ 0 & -A^T P - PA \end{bmatrix} \succ 0$$

8.2.4 Hamiltonian and LMI equivalence for a linear system

The general Hamiltonian equation

$$\frac{\partial V}{\partial x} f(x, w) \leq s(w, g(x, w))$$

Has been particularized for a linear system:

$$x' = f(x, w) = Ax + Bw$$

$$z = g(x, w) = Cx + Dw$$

As:

$$2x^T Kx' \leq \begin{bmatrix} w & z \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix} \quad (\text{Eq 8-5})$$

(Note: K is not the controller but a symmetric matrix).

We want to demonstrate that the previous expression is a LMI.

Demonstration:

Note that:

$$2x^T Kx' = \begin{bmatrix} x' & x \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} x \\ x' \end{bmatrix}$$

$$\text{Because } \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} x \\ x' \end{bmatrix} = \begin{bmatrix} Kx \\ Kx' \end{bmatrix} \text{ and } \begin{bmatrix} x' & x \end{bmatrix} \begin{bmatrix} Kx \\ Kx' \end{bmatrix} = x'Kx + xKx' = 2xKx'$$

Note that

$$\begin{bmatrix} w \\ z \end{bmatrix} = \begin{bmatrix} w \\ Cx + Dw \end{bmatrix} = \begin{bmatrix} 0 & I \\ C & D \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

Note that $x' = Ax + Bw$, and we can write

$$\begin{bmatrix} x \\ x' \end{bmatrix} = \begin{bmatrix} x \\ Ax + Bw \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ A & B \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

And also

$$\begin{bmatrix} x' & x \end{bmatrix} = \begin{bmatrix} Ax + Bw & x \end{bmatrix} = \begin{bmatrix} x & w \end{bmatrix} \begin{bmatrix} A & 1 \\ B & 0 \end{bmatrix}$$

So the first term is:

$$2x^T Kx' = \begin{bmatrix} x' & x \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} x \\ x' \end{bmatrix} = \begin{bmatrix} x & w \end{bmatrix} \begin{bmatrix} A & 1 \\ B & 0 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A & B \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

Finally, let be:

$$J = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad J \text{ is its own inverse: } \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We can introduce $I = J^*J$

$$2x^T Kx' = [x \quad w] \begin{bmatrix} A & 1 \\ B & 0 \end{bmatrix} \left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A & B \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

And then

$$2x^T Kx' = [x \quad w] \left(\begin{bmatrix} A & 1 \\ B & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \right) \begin{bmatrix} 1 & 0 \\ A & B \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

Arriving to:

$$2x^T Kx' = [x \quad w] \begin{bmatrix} 1 & A \\ 0 & B \end{bmatrix} \begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A & B \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} \quad (\text{Eq 8-6})$$

Then we can replace both terms on the general formula (Eq 8-5):

$$2x^T Kx' \leq [w \quad z] \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix}$$

$$[x \quad w] \begin{bmatrix} I & A \\ 0 & B \end{bmatrix} \begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ A & B \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} < [x \quad w] \begin{bmatrix} 0 & C \\ I & D \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} 0 & I \\ C & D \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}$$

That is a LMI in Kl

$$[x \quad w] \left(\begin{bmatrix} I & 0 \\ A & B \end{bmatrix} \begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ A & B \end{bmatrix} - \begin{bmatrix} 0 & I \\ C & D \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} 0 & I \\ C & D \end{bmatrix} \right) \begin{bmatrix} x \\ w \end{bmatrix} < 0,$$

8.2.5 Hamiltonian matrix, Riccati Equation and H_∞ norm

8.2.5.1 Hamiltonian matrices

Let define the matrix $J \in R^{2n \times 2n}$ as:

$$J = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \quad (\text{Note that } J^T = -J)$$

A second matrix A is called Hamiltonian if the product $J * A$ is asymmetric, i.e.

$$JA = (JA)^T$$

but note that:

$$(JA)^T = A^T J^T = A^T (-J)$$

So we have

$$(JA)^T - JA = 0$$

$$A^T (-J) - JA = 0$$

$$A^T J + JA = 0$$

We can define the space of Hamiltonian matrices as:

$$H^n = \{A \in R^{2n \times 2n} : A^T J + JA = 0\}$$

It can be demonstrated, see (Datta, 2004) that:

- If $A \in H^n$ and $B \in H^n$, then $A+B \in H^n$ (sum of Hamiltonian is Hamiltonian)
- If $A \in H^n$ and $c \in R^n$, then $cA \in H^n$ (scaling of a Hamiltonian is Hamiltonian)
- If $A \in H^n$ and $B \in H^n$, then “product $A*B$ ” defined as $A*B - BA$ is Hamiltonian
- What’s more, if λ is an eigenvalue of H, then $(-\lambda)$ is also an eigenvalue

8.2.5.2 Mathematical equivalence of Hamiltonian matrix and Riccati equation

An algebraic Riccati equation has the form

$$0 = F + A^T X + XA - XGX$$

Where F and G are symmetric matrices ($F = F^T$, $G = G^T$) and X is the variable.

It is possible to associate a Hamiltonian matrix to a Riccati equation.

Define a $2n \times 2n$ matrix:

$$H = \begin{bmatrix} A & G \\ F & -A^T \end{bmatrix}$$

Let U, V vectors defining a subspace of H.

$$\begin{bmatrix} A & G \\ F & -A^T \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} U \\ V \end{bmatrix} Z$$

From first row we obtain:

$$AU + GV = UZ \rightarrow U^{-1}AU + U^{-1}GV = Z$$

From second row we obtain:

$$FU - A^T V = VZ$$

And replacing Z

$$FU - A^T V = VU^{-1}AU + VU^{-1}GV$$

That can be put in form of a Riccati equation,

$$0 = FU - A^T V - VU^{-1}AU - VU^{-1}GV$$

Multiplying by the right by U^{-1} :

$$0 = F - A^T VU^{-1} - VU^{-1}AUU^{-1} - VU^{-1}GVU^{-1}$$

if we define $X = -VU^{-1}$

$$0 = F - A^T X - XA - XGX$$

$H = \begin{bmatrix} A & G \\ F & -A^T \end{bmatrix}$ is equivalent to Riccati: $F - A^T X - XA - XGX = 0$

8.2.5.3 Hamiltonian matrix and H_∞ norm of a system

In previous paragraph has been demonstrated that from a mathematical point of view, a Riccati equation has associated a Hamiltonian representation. In this paragraph we demonstrate that the Hamiltonian representation is in fact the 'A' state space matrix of the following closed loop system (Figure 8-10):

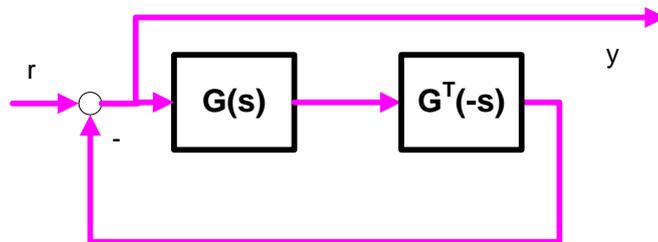


Figure 8-10: Setup for Hamiltonian calculus

Where:

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right], \quad G^T(-s) = \left[\begin{array}{c|c} -A^T & -C^T \\ \hline B^T & 0 \end{array} \right],$$

And we had taken $\gamma=1$. (This can be done with appropriate scaling $G \rightarrow \gamma^{-1}G$ and $B \rightarrow \gamma^{-1}B$).

The state space representation of the closed loop is:

$$\begin{bmatrix} x' \\ u \end{bmatrix} = \left[\begin{array}{cc|c} A & BB^T & B \\ -CC^T & -A^T & 0 \\ \hline 0 & B^T & I \end{array} \right] \begin{bmatrix} x \\ r \end{bmatrix}$$

H is the A matrix of the system $S = 1/(I - G_T(-s)G(s))$. The poles of S are described by the eigenvalues of H . As we have assumed that $\bar{\sigma}(G) < 1$, $(I - G_T(-s)G(s))$ is always > 0 . This implies that the denominator in S is never zero, so S has no poles on the $j\omega$ -axis.

Then we identify the 'A' matrix of the system with the Hamiltonian matrix that by previous paragraph is equivalent to a Riccati equation:

$$H = \begin{bmatrix} A & BB^T \\ -CC^T & -A^T \end{bmatrix} \text{ is equivalent to Riccati } -CC^T - A^T X - XA - XBB^T X = 0$$

The following statements have been demonstrated to be equivalent:

- $\|G\|_\infty < \gamma$
- the Hamiltonian matrix H of $S = 1/(I - G_T(-s)G(s))$ has no eigenvalues in the $j\omega$ -axis
- The Riccati equation has solution

8.2.5.4 Computing the H_∞ norm of a system

It is not easy to calculate the H_∞ norm of a system from its state space equation. Instead, it is easy to check that the norm is minor than a given value.

The bisection search algorithm is commonly used:

1. Select γ_l, γ_u such that $\gamma_l < \|G\|_\infty < \gamma_u$
2. Test if $(\gamma_u - \gamma_l) / \gamma_l < \text{tolerance}$ (i.e. the gammas are separated enough)
 - If yes, STOP : $\|G\|_\infty = 0.5 * (\gamma_l + \gamma_u)$
 - If not, GOTO step 3
3. Do $\gamma = 0.5 * (\gamma_l + \gamma_u)$ and test if $\|G\|_\infty < \gamma$ using the Hamiltonian matrix
4. If $\lambda_i(H)$ in imaginary axis, then $\gamma_l = \gamma$ (try with upper semi-interval),
5. If $\lambda_i(H)$ not in imaginary axis, then $\gamma_u = \gamma$ (try with lower semi-interval).
6. GO TO step 2

8.2.6 LMI of the closed loop in plant and controller variables

Given the general expression for closed loop of the plant P and the controller K :

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{cc|c} A + B_2 D_K C_2 & B_2 C_K & B_1 + B_2 D_K D_{21} \\ B_K C_2 & A_K & B_K D_{21} \\ \hline C_1 + D_{12} + D_K C_2 & D_{12} C_K & D_{11} + D_{12} D_K D_{21} \end{array} \right]$$

The general expression for the LMI for the closed loop is:

$$F(X) = \begin{bmatrix} A^T X + XA & XB & C^T \\ B^T X & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0 \text{ (where } A, B, C, D \text{ refers to closed loop variables!)}$$

Let's do a change of notation in the closed loop plant:

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{cc|c} \bar{A} + \bar{B}_2 K \bar{C}_2 & \bar{B}_1 + \bar{B}_2 K \bar{D}_{21} \\ \hline \bar{C}_1 + \bar{D}_{12} K \bar{C}_2 & \bar{D}_{11} + D_{12} K \bar{D}_{21} \end{array} \right]$$

With notation:

$$\begin{aligned} \bar{A} &= \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}, & \bar{B}_1 &= \begin{bmatrix} B_1 \\ 0 \end{bmatrix}, & \bar{B}_2 &= \begin{bmatrix} 0 & B_2 \\ I & 0 \end{bmatrix} \\ \bar{C}_1 &= [C_1 \quad 0], & \bar{C}_2 &= \begin{bmatrix} 0 & I \\ C_2 & 0 \end{bmatrix} \\ \bar{D}_{11} &= D_{11}, & \bar{D}_{21} &= \begin{bmatrix} 0 \\ D_{21} \end{bmatrix}, & \bar{D}_{12} &= [0 \quad D_{12}] \end{aligned}$$

With a change of variables is defined as follows:

$$\begin{aligned} P_X &= \begin{bmatrix} \bar{B}_2^T X & 0 & \bar{D}_{12}^T \end{bmatrix} \\ Q &= \begin{bmatrix} \bar{C}_2 & \bar{D}_{21} & 0 \end{bmatrix} \end{aligned}$$

And writing the following LMI in function of $\bar{A}, \bar{B}_1 \dots$ variables

$$H(X) = \begin{bmatrix} \bar{A}^T X + X \bar{A} & X \bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X & -\gamma I & \bar{D}_{11}^T \\ \bar{C}_1 & \bar{D}_{11} & -\gamma I \end{bmatrix}$$

We want to demonstrate that the LMI $F(X)$ becomes:

$$F(X) = H(X) + Q^T K^T P_X + P_X^T K Q < 0 \quad (\text{Eq 8-7})$$

i.e. the LMI in closed loop $F(X)$ can be written in function of the LMI in open loop variables (plant variables) $H(X)$ plus the terms on Q, K, P_X .

Demonstration

Let's calculate the terms of the right part of (Eq 8-7) :

Term $H(X)$

$$H(X) = \begin{bmatrix} \overline{A}^T X + X\overline{A} & X\overline{B}_1 & \overline{C}_1^T \\ \overline{B}_1^T X & -\mathcal{H} & \overline{D}_{11}^T \\ \overline{C}_1 & \overline{D}_{11} & -\mathcal{H} \end{bmatrix} = \left[\begin{array}{cc|cc} A^T X + XA & 0 & XB_1 & C_1^T \\ 0 & 0 & 0 & 0 \\ \hline B_1^T X & 0 & I & D_{11}^T \\ \hline C_1 & 0 & D_{11} & I \end{array} \right]$$

Term $(P_X^T KQ)$:

$$\begin{aligned} P_X^T KQ &= [\overline{B}_2^T X \quad 0 \quad \overline{D}_{12}^T]^T K [\overline{C}_2 \quad \overline{D}_{21} \quad 0] \\ &= [\overline{B}_2^T X \quad 0 \quad \overline{D}_{12}^T]^T \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix} \begin{bmatrix} (0 \quad I) \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ (C_2 \quad 0) \begin{pmatrix} 0 \\ D_{21} \end{pmatrix} \\ (0) \end{bmatrix} = \\ &= [\overline{B}_2^T X \quad 0 \quad \overline{D}_{12}^T]^T \begin{bmatrix} B_K C_2 & A_K & B_K D_{21} & 0 \\ D_K C_2 & C_K & D_K D_{21} & 0 \end{bmatrix} \end{aligned}$$

$$\begin{bmatrix} X \begin{pmatrix} 0 & B_2 \\ I & 0 \\ 0 & 0 \\ 0 & D_{12} \end{pmatrix} \end{bmatrix} \begin{bmatrix} B_K C_2 & A_K & B_K D_{21} & 0 \\ D_K C_2 & C_K & D_K D_{21} & 0 \end{bmatrix} =$$

$$\begin{bmatrix} \begin{pmatrix} 0 & XB_2 \\ X & 0 \\ 0 & 0 \\ 0 & D_{12} \end{pmatrix} \end{bmatrix} \begin{bmatrix} B_K C_2 & A_K & B_K D_{21} & 0 \\ D_K C_2 & C_K & D_K D_{21} & 0 \end{bmatrix} =$$

$$\begin{bmatrix} XB_2 D_K C_2 & XB_2 C_K & XB_2 D_K D_{21} & 0 \\ XB_K C_2 & XA_K & XB_K D_{21} & 0 \\ 0 & 0 & 0 & 0 \\ D_{12} D_K C_2 & D_{12} C_K & D_{12} D_K D_{21} & 0 \end{bmatrix}$$

Term $(Q^T K^T P_X)$:

It is the transposed of previous

$$\begin{bmatrix} XB_2D_KC_2 & XB_2C_K & XB_2D_KD_{21} & 0 \\ XB_KC_2 & XA_K & XB_KD_{21} & 0 \\ 0 & 0 & 0 & 0 \\ D_{12}D_KC_2 & D_{12}C_K & D_{12}D_KD_{21} & 0 \end{bmatrix}^T$$

Now let's add the 3 terms, element by element:

$$F(X) = H(X) + Q^T K^T P_X + P_X^T K Q$$

First element is

$$\begin{aligned} & (A^T X + XA) + (B_2D_KC_2)^T X + X(B_2D_KC_2) = \\ & (A + B_2D_KC_2)^T X + X(A + B_2D_KC_2) \end{aligned}$$

Other elements are calculated directly, given the matrix:

$$\begin{bmatrix} (A + B_2D_KC_2)^T X + X(A + B_2D_KC_2) & (B_KC_2)^T X + XB_2C_K & XB_1 + XB_2D_KD_{21} & C_1^T + (D_{12}D_KC_2)^T \\ (B_2C_K)^T X + XB_KC_2 & A_K^T X + XA_K & XB_KD_{21} & (D_{12}C_K)^T \\ B_1^T X + (B_2D_KD_{21})^T X & (B_KD_{21})^T X & I & D_{11}^T + (D_{12}D_KD_{21})^T \\ C_1 + D_{12}D_KC_2 & D_{12}C_K & D_{11} + D_{12}D_KD_{21} & 0 \end{bmatrix}$$

(Eq 8-8)

Let's calculate the left part (closed loop LMI) from the general formula:

$$F(X) = \begin{bmatrix} A^T X + XA & XB & C^T \\ B^T X & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} \prec 0$$

Where the (A, B, C, D) of the closed loop systems are given by:

$$T_{zw} = \begin{bmatrix} A_{CL} & B_{CL} \\ C_{CL} & D_{CL} \end{bmatrix} = \left[\begin{array}{cc|c} A + B_2D_KC_2 & B_2C_K & B_1 + B_2D_KD_{21} \\ B_KC_2 & A_K & B_KD_{21} \\ \hline C_1 + D_{12}D_KC_2 & D_{12}C_K & D_{11} + D_{12}D_KD_{21} \end{array} \right]$$

Example: terms (1, 1) to (2, 2) are:

$$\begin{aligned} A^T X + XA &= \begin{bmatrix} A + B_2D_KC_2 & B_2C_K \\ B_KC_2 & A_K \end{bmatrix}^T X + X \begin{bmatrix} A + B_2D_KC_2 & B_2C_K \\ B_KC_2 & A_K \end{bmatrix} = \\ & \begin{bmatrix} (A + B_2D_KC_2)^T X + X(A + B_2D_KC_2) & (B_KC_2)^T X + X(B_2C_K) \\ (B_2C_K)^T X + X(B_KC_2) & A_K^T X + XA_K \end{bmatrix} \end{aligned}$$

Term XB is:

$$X(B_1 + B_2 D_K D_{21}) \\ B_K D_{21}$$

Other terms are straightforward, giving the matrix:

$$\begin{bmatrix} (A + B_2 D_K C_2)^T X + X(A + B_2 D_K C_2) & (B_K C_2)^T X + X B_2 C_K & X B_1 + X B_2 D_K D_{21} & C_1^T + (D_{12} D_K C_2)^T \\ (B_2 C_K)^T X + X B_K C_2 & A_K^T X + X A_K & X B_K D_{21} & (D_{12} C_K)^T \\ B_1^T X + (B_2 D_K D_{21})^T X & (B_K D_{21})^T X & I & D_{11}^T + (D_{12} D_K D_{21})^T \\ C_1 + D_{12} D_K C_2 & D_{12} C_K & D_{11} + D_{12} D_K D_{21} & 0 \end{bmatrix}$$

That is identical to the matrix in (Eq 8-8) .

8.2.7 The Elimination Lemma

We want to decompose the following LMI in two LMIs, but removing the dependency on the controller K .

$$F(X) = H(X) + Q^T K^T P_X + P_X^T K Q < 0$$

Let's define BP_X as a basis of the kernel of P_X . Let's define W_{P_X} as the orthogonal operator of BP_X : $W_{P_X} \equiv BP_X^\perp$

Multiplying by $(BP_X^\perp)^T$ to the left and (BP_X^\perp) to the right we have:

$$F(X) = (BP_X^\perp)^T H(X) (BP_X^\perp) + (BP_X^\perp)^T Q^T K^T P_X (BP_X^\perp) + (BP_X^\perp)^T P_X^T K Q (BP_X^\perp) < 0$$

as

$$P_X BP_X^\perp = 0, \quad BP_X^\perp P_X = 0,$$

We have:

$$F(X) = (BP_X^\perp)^T H(X) BP_X^\perp < 0$$

Or

$$F(X) = W_{P_X}^T H(X) W_{P_X} < 0$$

Similarly, let's define BQ as a basis of the kernel of Q . Let's define W_Q as the orthogonal operator of BQ : $W_Q \equiv BQ^\perp$

Multiplying by $(BQ^\perp)^T$ to the left and BQ^\perp to the right we arrive to:

$$F(X) = (BQ^\perp)^T H(X) BQ^\perp < 0$$

Or

$$F(X) = (W_Q)^T H(X) B W_Q < 0$$

So the LMI:

$$F(X) = H(X) + Q^T K^T P_x + P_x^T K Q \prec 0$$

Is equivalent to the two LMIs:

$$F(X) = (P_x^\perp)^T H(X) P_x^\perp \prec 0$$

$$F(X) = (Q^\perp)^T H(X) Q^\perp \prec 0$$

Or

$$F(X) = W_{P_x}^T H(X) W_{P_x} \prec 0$$

$$F(X) = (W_Q)^T H(X) B W_Q \prec 0$$

8.2.8 Final step of LMI demonstration

We want to demonstrate that the formulation of the problem in terms of LMI closed loop variables:

$$W_P^T T(Y) W_P = W_P^T \begin{bmatrix} \bar{A}^T Y + Y \bar{A} & \bar{B}_1 & Y^T \bar{C}_1^T \\ \bar{B}_1^T & -\mathcal{I} & \bar{D}_{11}^T \\ \bar{C}_1 Y & \bar{D}_{11} & -\mathcal{I} \end{bmatrix} W_P \prec 0$$

$$W_Q^T H(X) W_Q = W_Q^T \begin{bmatrix} \bar{A}^T X + X \bar{A} & X \bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X & -\mathcal{I} & \bar{D}_{11}^T \\ \bar{C}_1 & \bar{D}_{11} & -\mathcal{I} \end{bmatrix} W_Q \prec 0$$

With X and Y portioned as:

$$X = \begin{bmatrix} X_P & X_2 \\ X_2^T & I \end{bmatrix}, \quad Y = X^{-1} = \begin{bmatrix} Y_P & Y_2 \\ Y_2^T & I \end{bmatrix}$$

Is equivalent to the conditions expressed by the following 3 LMIs:

$$\begin{bmatrix} N_X & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T X_P + X_P A & X_P B_1 & C_1^T \\ B_1^T X_P & -\mathcal{I} & D_{11}^T \\ C_1 & D_{11} & -\mathcal{I} \end{bmatrix} \begin{bmatrix} N_X & 0 \\ 0 & I \end{bmatrix} \prec 0,$$

$$\begin{bmatrix} N_Y & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T Y_P + Y_P A & Y_P C_1^T & B_1 \\ C_1 Y_P & -\mathcal{I} & D_{11} \\ B_1^T & D_{11}^T & -\mathcal{I} \end{bmatrix} \begin{bmatrix} N_Y & 0 \\ 0 & I \end{bmatrix} \prec 0,$$

$$\begin{bmatrix} X_P & 0 \\ 0 & Y_P \end{bmatrix} \geq 0$$

Where:

$$\begin{aligned} \text{Im } N_X &= \ker \begin{bmatrix} C_2 & D_{21} \end{bmatrix} \\ \text{Im } N_Y &= \ker \begin{bmatrix} B_2^T & D_{12}^T \end{bmatrix} \end{aligned}$$

And all is expressed in open loop plant variables.

Demonstration:

Step1: First let's work with the LMI on X (Eq 3-12). The LMI is:

$$W_Q^T H(X) W_Q = W_Q^T \begin{bmatrix} \bar{A}^T X + X \bar{A} & X \bar{B}_1 & \bar{C}_1^T \\ \bar{B}_1^T X & -\mathcal{I} & \bar{D}_{11}^T \\ \bar{C}_1 & \bar{D}_{11} & -\mathcal{I} \end{bmatrix} W_Q \prec 0$$

Where:

$$X = \begin{bmatrix} X_p & X_2 \\ X_2^T & I \end{bmatrix}$$

$$Q = \begin{bmatrix} \bar{C}_2 & \bar{D}_{21} & 0 \end{bmatrix}$$

$$W_Q^T H(X) W_Q = W_Q^T \begin{bmatrix} A^T X_p + X_p A & A^T X_2 & X_p B_1 & C_1^T \\ X_2^T A & 0 & X_2^T B_1 & 0 \\ \hline B_1^T X_p & B_1^T X_2 & -\mathcal{I} & D_{11}^T \\ \hline C_1 & 0 & D_{11} & -\mathcal{I} \end{bmatrix} W_Q \prec 0$$

Move the second row as last row. Move the second column as last column:

$$W_Q^T H(X) W_Q = W_Q^T \begin{bmatrix} A^T X_p + X_p A & X_p B_1 & C_1^T & A^T X_2 \\ B_1^T X_p & -\mathcal{I} & D_{11}^T & B_1^T X_2 \\ C_1 & D_{11} & -\mathcal{I} & 0 \\ X_2^T A & X_2^T B_1 & 0 & 0 \end{bmatrix} W_Q \prec 0$$

The orthogonal vectors Q^\perp can be built on a smart way such that cancels the X_2 variables. First introduce the change of rows (2 row moved last row) also in the matrix Q .

$$Q = \begin{bmatrix} \bar{C}_2 & \bar{D}_{21} & 0 \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 \\ C_2 & 0 & D_{21} & 0 \end{bmatrix}$$

$$Q = \begin{bmatrix} 0 & 0 & 0 & I \\ C_2 & D_{21} & 0 & 0 \end{bmatrix}$$

Then choose the orthogonal vector

$$Q^\perp = \begin{bmatrix} N_{x1} & 0 \\ N_{x2} & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix}$$

Such that:

$$\begin{bmatrix} 0 & 0 & 0 & I \\ C_2 & D_{21} & 0 & 0 \end{bmatrix} \begin{bmatrix} N_{x1} & 0 \\ N_{x2} & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix} = 0$$

With these we have

$$W_Q^T H(X) W_Q = \begin{bmatrix} N_{x1} & N_{x2} & 0 & 0 \\ 0 & 0 & I & 0 \end{bmatrix} \begin{bmatrix} A^T X_p + X_p A & X_p B_1 & C_1^T & A^T X_2 \\ B_1^T X_p & -\mathcal{I} & D_{11}^T & B_1^T X_2 \\ C_1 & D_{11} & -\mathcal{I} & 0 \\ X_2^T A & X_2^T B_1 & 0 & 0 \end{bmatrix} \begin{bmatrix} N_{x1} & 0 \\ N_{x2} & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix} \prec 0 \quad (\text{Eq 8-9})$$

As the last row of Q^\perp is zero, the last column of the LMI does not constraint the LMI. As the last column of Q^\perp is zero, the last row of the LMI does not constraint the LMI. So these can be removed. (It is easy to see just by performing the product and checking that the terms on X_2 and X_2^T disappear). With this we arrive to (Eq 3-13) .

$$W_Q^T H(X) W_Q = \begin{bmatrix} N_{x1} & 0 \\ N_{x2} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T X + XA & XB_1 & C_1^T \\ B_1^T X & -\mathcal{I} & D_{11}^T \\ C_1 & D_{11} & -\mathcal{I} \end{bmatrix} \begin{bmatrix} N_{x1} & 0 \\ N_{x2} & 0 \\ 0 & I \end{bmatrix} \prec 0$$

Step 2: A similar demonstration can be done for the LMI on Y .

Step 3: The third condition

$$\begin{bmatrix} X_p & 0 \\ 0 & Y_p \end{bmatrix} \succ 0$$

Is deduced as follows: as Y is the inverse of X

$$XY = XX^{-1} = I$$

$$XY = \begin{bmatrix} X_p & X_2 \\ X_2^T & I \end{bmatrix} \begin{bmatrix} Y_p & Y_2 \\ Y_2^T & I \end{bmatrix} = \begin{bmatrix} X_p Y_p + X_2 Y_2^T & X_p Y_2 + X_2 \\ X_2^T Y_p + Y_2^T & X_2^T Y_2 + I \end{bmatrix}$$

Identifying the elements (1,1) and (2,1) with the identity matrix:

$$X_p Y_p + X_2 Y_2^T = 1$$

$$X_2^T Y_p + Y_2^T = 0, \quad Y_2^T = -X_2^T Y_p$$

Replacing on the first equation

$$X_p Y_p + X_2 (-X_2^T Y_p) = 1$$

$$X_p - X_2 X_2^T = Y_p^{-1}$$

$$X_p - Y_p^{-1} = X_2 X_2^T$$

But $X_2 X_2^T$ is quadratic, so is definite positive

$$X_p - Y_p^{-1} = X_2 X_2^T \succ 0$$

$$X_p - Y_p^{-1} \succ 0$$

Applying the Schur complement the third condition is found

$$\begin{bmatrix} X_p & I \\ I & Y_p \end{bmatrix} \succ 0$$

8.3 Demonstrations for chapter 4

8.3.1 State space formulas for augmented plants

Latest Matlab version include commands for create state space representations for augmented plants. This was not the case some years ago: these representations had to be constructed manually (also to guarantee that the representations were minimal, i.e. with the minimum possible number of states).

8.3.1.1 General formulas

The following state space representations for typical matrix transfer functions are given in the literature:

Composition of systems:

The following representation is provided in (Skogestad, 1996).

$$G = \begin{bmatrix} G_1 & G_2 \\ G_3 & G_4 \end{bmatrix} = \left[\begin{array}{cccc|cc} A_1 & 0 & 0 & 0 & B_1 & 0 \\ 0 & A_2 & 0 & 0 & 0 & B_2 \\ 0 & 0 & A_3 & 0 & B_3 & 0 \\ 0 & 0 & 0 & A_4 & 0 & B_4 \\ \hline C_1 & C_2 & 0 & 0 & D_1 & D_2 \\ 0 & 0 & C_3 & C_4 & D_3 & D_4 \end{array} \right] \quad (ss1)$$

Representations for $[G1 \ G2]$, $[G1 \ G3]^T$ can be obtained from previous removing row and columns identical to zero.

$$G = [G_1 \ G_2] = \left[\begin{array}{cccc|cc} A_1 & 0 & 0 & 0 & B_1 & 0 \\ 0 & A_2 & 0 & 0 & 0 & B_2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline C_1 & C_2 & 0 & 0 & D_1 & D_2 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right] = \left[\begin{array}{cc|cc} A_1 & 0 & B_1 & 0 \\ 0 & A_2 & 0 & B_2 \\ \hline C_1 & C_2 & D_1 & D_2 \end{array} \right]$$

$$G = \begin{bmatrix} G_1 \\ G_3 \end{bmatrix} = \left[\begin{array}{cccc|cc} A_1 & 0 & 0 & 0 & B_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_3 & 0 & B_3 & 0 \\ \hline C_1 & 0 & 0 & 0 & D_1 & 0 \\ 0 & 0 & C_3 & 0 & D_3 & 0 \end{array} \right] = \left[\begin{array}{cc|c} A_1 & 0 & B_1 \\ 0 & A_3 & B_3 \\ \hline C_1 & 0 & D_1 \\ 0 & C_3 & D_3 \end{array} \right]$$

The product of transfer functions has the following representation, see (Skogestad, 1996).

$$G = G_1 G_2 = \left[\begin{array}{cc|c} A_2 & 0 & B_2 \\ \hline B_1 C_2 & A_1 & B_1 D_2 \\ \hline D_1 C_2 & C_1 & D_1 D_2 \end{array} \right] \quad (ss2)$$

8.3.1.2 Demonstration of Kwakernaak state space representation

The augmented plant is:

$$\begin{bmatrix} z_1 \\ z_2 \\ u \end{bmatrix} = \begin{bmatrix} -W_1 V & -W_1 P \\ 0 & W_2 \\ -V & -P \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix}$$

That can be factorized as:

$$\begin{bmatrix} z_1 \\ z_2 \\ u \end{bmatrix} = \begin{bmatrix} -W_1 & 0 & 0 \\ 0 & W_2 & 0 \\ -I & 0 & 0 \end{bmatrix} \begin{bmatrix} V & P \\ 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix}$$

The left matrix is formed (by ss1) as:

$$\begin{bmatrix} -W_1 & 0 \\ 0 & W_2 \end{bmatrix} = \left[\begin{array}{cc|cc} A_1 & 0 & -B_1 & 0 \\ 0 & A_2 & 0 & B_2 \\ \hline C_1 & 0 & -D_1 & 0 \\ 0 & C_2 & 0 & D_2 \end{array} \right]$$

Then

$$\begin{bmatrix} -I & 0 \end{bmatrix} = \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \hline 0 & -I & 0 \end{array} \right]$$

And composing

$$\begin{bmatrix} -W_1 & 0 & 0 \\ 0 & W_2 & 0 \\ -I & 0 & 0 \end{bmatrix} = \left[\begin{array}{cc|ccc} A_1 & 0 & -B_1 & 0 & 0 \\ 0 & A_2 & 0 & B_2 & 0 \\ \hline C_1 & 0 & -D_1 & 0 & 0 \\ 0 & C_2 & 0 & D_2 & 0 \\ 0 & 0 & -I & 0 & 0 \end{array} \right]$$

The second factor is formed as (by ss1):

$$\begin{bmatrix} V & P \\ 0 & I \end{bmatrix} = \left[\begin{array}{cccc|cc} A_V & 0 & 0 & 0 & B_V & 0 \\ 0 & A_P & 0 & 0 & 0 & B_P \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline C_V & C_P & 0 & 0 & D_V & D_P \\ 0 & 0 & 0 & 0 & 0 & I \end{array} \right] = \left[\begin{array}{cc|cc} A_V & 0 & B_V & 0 \\ 0 & A_P & 0 & B_P \\ \hline C_V & C_P & D_V & D_P \\ 0 & 0 & 0 & I \end{array} \right]$$

Composing...

$$\begin{bmatrix} V & P \\ 0 & I \\ 0 & 0 \end{bmatrix} = \left[\begin{array}{cc|cc} A_V & 0 & B_V & 0 \\ 0 & A_P & 0 & B_P \\ \hline C_V & C_P & D_V & D_P \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{array} \right]$$

The composition of both factors is (by ss2):

$$P_{AU} = \left[\begin{array}{cc|cc|cc} A_V & 0 & 0 & 0 & B_V & 0 \\ 0 & A_P & 0 & 0 & 0 & B_P \\ \hline -B_1 C_V & -B_1 C_P & A_1 & 0 & -B_1 D_V & -B_1 D_P \\ 0 & 0 & 0 & A_2 & 0 & B_2 \\ \hline -D_1 C_V & -D_1 C_P & C_1 & 0 & -D_1 D_V & -D_1 D_P \\ 0 & 0 & 0 & C_2 & 0 & D_2 \\ -C_V & -C_P & 0 & 0 & -D_V & -D_P \end{array} \right]$$

8.3.1.3 Demonstration of Christen–Geering state space representation

The augmented plant is:

$$\begin{bmatrix} z_1 \\ z_2 \\ d_y \end{bmatrix} = \begin{bmatrix} 0 & 0 & W_u \\ W_y P & 0 & W_y P \\ -P & W_d & -P \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ d_u \end{bmatrix}$$

That can be factorized as:

$$\begin{bmatrix} z_1 \\ z_2 \\ d_y \end{bmatrix} = \begin{bmatrix} W_u & 0 & 0 \\ 0 & W_y & 0 \\ 0 & -1 & W_d \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ P & 0 & P \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ d_u \end{bmatrix}$$

The left matrix is formed (applying ss1) as:

$$\begin{bmatrix} W_y & 0 \\ -1 & W_d \end{bmatrix} = \left[\begin{array}{cc|cc} A_y & 0 & B_y & 0 \\ 0 & 0 & 0 & 0 \\ \hline C_y & 0 & D_y & 0 \\ 0 & 0 & -1 & w_d \end{array} \right]$$

$$\left[\begin{array}{c|ccc} W_u & 0 & 0 & \\ \hline 0 & W_y & 0 & \\ 0 & -1 & W_d & \end{array} \right] = \left[\begin{array}{cc|ccc} A_u & 0 & B_u & 0 & 0 \\ 0 & A_y & 0 & B_y & 0 \\ \hline C_u & 0 & D_u & 0 & 0 \\ 0 & C_y & 0 & D_y & 0 \\ 0 & 0 & 0 & -1 & w_d \end{array} \right]$$

The second matrix is formed (ss1) as:

$$\begin{bmatrix} P & 0 \\ 0 & 1 \end{bmatrix} = \left[\begin{array}{cc|cc} A_p & 0 & B_p & 0 \\ 0 & 0 & 0 & 0 \\ \hline C_p & 0 & D_p & 0 \\ 0 & 0 & 0 & 1 \end{array} \right]$$

$$\begin{bmatrix} P & 0 & P \\ 0 & 1 & 0 \end{bmatrix} = \left[\begin{array}{ccc|ccc} A_p & 0 & 0 & B_p & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_p & 0 & 0 & B_p \\ \hline C_p & 0 & C_p & D_p & 0 & D_p \\ 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right] = \left[\begin{array}{c|cccc} A_p & B_p & 0 & B_p \\ \hline C_p & D_p & 0 & D_p \\ 0 & 0 & 1 & 0 \end{array} \right]$$

$$\begin{bmatrix} 0 & 0 & 1 \\ P & 0 & P \\ 0 & 1 & 0 \end{bmatrix} = \left[\begin{array}{cc|ccc} 0 & 0 & 0 & 0 & 0 \\ 0 & A_p & B_p & 0 & B_p \\ \hline 0 & 0 & 0 & 0 & 1 \\ 0 & C_p & D_p & 0 & D_p \\ 0 & 0 & 0 & 1 & 0 \end{array} \right]$$

Finally, applying the composition formula (ss2) to both factors:

$$\left[\begin{array}{cccc|ccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & A_p & 0 & 0 & B_p & 0 & B_p \\ 0 & 0 & A_u & 0 & 0 & 0 & B_u \\ 0 & B_y C_p & 0 & A_y & B_y D_p & 0 & B_y D_p \\ \hline 0 & 0 & C_u & 0 & 0 & 0 & D_u \\ 0 & D_y C_p & 0 & C_y & D_y D_p & 0 & D_y D_p \\ 0 & -C_p & 0 & 0 & -D_p & w_d & -D_p \end{array} \right]$$

That is identical to the proposed in this work if we take $D_p = 0$ (so $B_y D_p = 0$, $D_y D_p = 0$) and remove the rows and column identically zero.

8.4 Demonstrations for chapter 5

8.4.1 No effect of gravity in LV acceleration (gravity turn)

We demonstrate here that the effect of the gravity during a gravity turn in the LV acceleration is small.

Let's take a 2-D problem with axis X and h (altitude). We have the generic Newton equations:

$$\begin{aligned} M\dot{v}_H &= F_H \\ M\dot{v}_X &= F_X \end{aligned}$$

Let's express v in polar coordinates, in function of the modulus of the velocity and the gamma angle (named the flight path angle, the angle between the velocity vector and the x-axis).

$$\begin{aligned} v_H &= v \sin(\gamma) \\ v_X &= v \cos(\gamma) \end{aligned}$$

$$\begin{aligned} M \frac{d}{dt}(v \sin(\gamma)) &= F_H \\ M \frac{d}{dt}(v \cos(\gamma)) &= F_X \end{aligned}$$

Applying the derivate chain rule (considering M constant):

$$\begin{aligned} M(\dot{v} \sin(\gamma) + v \cos(\gamma) \dot{\gamma}) &= F_H \\ M(\dot{v} \cos(\gamma) - v \sin(\gamma) \dot{\gamma}) &= F_X \end{aligned}$$

Multiplying the equation on F_H by $\sin(\gamma)$ and the equation on F_X by $\cos(\gamma)$ and adding them we have:

$$\begin{aligned} &M(\dot{v} \sin(\gamma) \sin(\gamma) + v \cos(\gamma) \sin(\gamma) \dot{\gamma}) \\ &+ M(\dot{v} \cos(\gamma) \cos(\gamma) - v \sin(\gamma) \cos(\gamma) \dot{\gamma}) = F_H \sin(\gamma) + F_X \cos(\gamma) \end{aligned}$$

First term can be simplified as:

$$\begin{aligned} &M(\dot{v} \sin(\gamma) \sin(\gamma) + \dot{v} \cos(\gamma) \cos(\gamma) + v \cos(\gamma) \sin(\gamma) \dot{\gamma} - v \sin(\gamma) \cos(\gamma) \dot{\gamma}) = \\ &M(\dot{v} \sin^2(\gamma) + \dot{v} \cos^2(\gamma)) = M\dot{v} \end{aligned}$$

Having

$$M\dot{v} = F_H \sin(\gamma) + F_X \cos(\gamma) \quad (\text{Eq 8-10})$$

Multiplying the equation on F_H by $\cos(\gamma)$ and the equation on F_X by $\sin(\gamma)$ and subtracting them we have:

$$\begin{aligned} & M(\dot{v} \sin(\gamma) \cos(\gamma) + v \cos(\gamma) \cos(\gamma) \dot{\gamma}) \\ & - M(\dot{v} \cos(\gamma) \sin(\gamma) - v \sin(\gamma) \sin(\gamma) \dot{\gamma}) = F_H \cos(\gamma) - F_X \sin(\gamma) \end{aligned}$$

First term can be simplified as:

$$M(\dot{v} \sin(\gamma) \cos(\gamma) - \dot{v} \cos(\gamma) \sin(\gamma) + v \cos(\gamma) \cos(\gamma) \dot{\gamma} + v \sin(\gamma) \sin(\gamma) \dot{\gamma}) = Mv\dot{\gamma}$$

Having

$$Mv\dot{\gamma} = F_H \cos(\gamma) - F_X \sin(\gamma) \quad (\text{Eq 8-11})$$

Now the forces acting on the launch vehicle are:

$$F_H = (T - D) \sin(\gamma) - Mg$$

$$F_X = (T - D) \cos(\gamma)$$

Substituting F_X and F_H in (Eq 8-10) we have:

$$M\dot{v} = ((T - D) \sin(\gamma) - Mg) \sin(\gamma) + ((T - D) \cos(\gamma)) \cos(\gamma)$$

Substituting F_X and F_H in (Eq 8-11) we have

$$Mv\dot{\gamma} = ((T - D) \sin(\gamma) - Mg) \cos(\gamma) - ((T - D) \cos(\gamma)) \sin(\gamma)$$

So we arrive to:

$$\dot{v} = (T - D)g \sin(\gamma)$$

$$v\dot{\gamma} = -g \cos(\gamma)$$

Substituting g in first equation:

$$g = \frac{-v\dot{\gamma}}{\cos(\gamma)}$$

We have:

$$\dot{v} = (T - D) \left(\frac{-v\dot{\gamma}}{\cos(\gamma)} \right) \sin(\gamma) \quad (\text{Eq 8-12})$$

So the expression of the acceleration of the rocket does not depends on g , so does not depends on the gravity in a first approximation.

8.4.2 Deduction of the Transport Theorem

Let's consider 2 reference frames: an inertial frame, denoted by the suffix I and a rotating frame with regard the previous denoted by the suffix R. The frame R rotates with an angular velocity ω with regard the frame I. See (Figure 8-11) and (Figure 8-12).

Let be a base (u_{1R}, u_{2R}, u_{3R}) for the rotating frame.

Let be a base (u_{1I}, u_{2I}, u_{3I}) for the inertial frame.

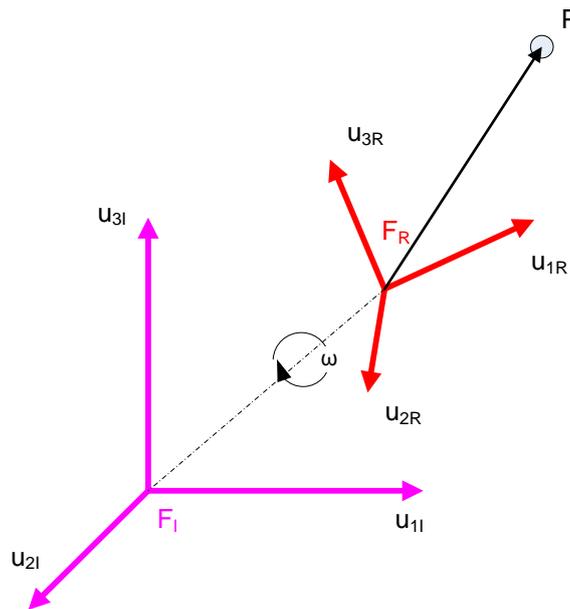


Figure 8-11: Inertial and rotating frame

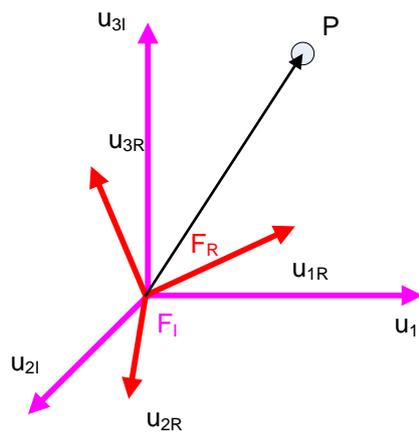


Figure 8-12: Inertial and rotating frame with same origin

Let's assume both frames have the same origin in a given instant (this simplifies the demonstration). There is a particle at point P that is moving. The vector r denotes the vector from the common origin of both frames to the point P . At each instant of time, the

vector r is identical on both frames (but the decomposition of r in components is of course different for each frame).

$$\vec{r} = x_R u_{1R} + y_R u_{2R} + z_R u_{3R} \quad \text{in the rotating frame}$$

$$\vec{r} = x_I u_{1I} + y_I u_{2I} + z_I u_{3I} \quad \text{in the inertial frame}$$

Let's take the derivative of the vector r in the inertial frame, but expressed in components of the rotating frame (we can do this because physically, we know that the vector is identical in both frames, even if the decomposition in components may be different)

$$\frac{d^I \vec{r}}{dt} \underset{(inI)}{=} \frac{d^I \vec{r}}{dt} \underset{(inB)}{=} \frac{d^I (x_R u_{1R} + y_R u_{2R} + z_R u_{3R})}{dt}$$

We apply now the usual derivative of a sum of products, having into account that both, the components of the vector in the rotating frame (x_R, y_R, z_R) and the unit vector of the rotating frame (u_{1R}, u_{2R}, u_{3R}) change along time as seen from the inertial frame.

$$\begin{aligned} \frac{d^I \vec{r}}{dt} &= \left(\frac{dx_R}{dt} u_{1R} + x_R \frac{du_{1R}}{dt} \right) + \left(\frac{dy_R}{dt} u_{2R} + y_R \frac{du_{2R}}{dt} \right) + \left(\frac{dz_R}{dt} u_{3R} + z_R \frac{du_{3R}}{dt} \right) \\ \frac{d^I \vec{r}}{dt} &= \left(\frac{dx_R}{dt} u_{1R} + \frac{dy_R}{dt} u_{2R} + \frac{dz_R}{dt} u_{3R} \right) + \left(x_R \frac{du_{1R}}{dt} + y_R \frac{du_{2R}}{dt} + z_R \frac{du_{3R}}{dt} \right) \end{aligned}$$

The first term of the previous equation is the usual derivative on an inertial frame. For the second term, we note that in a given instant of time it describes the rotation of the point P around the inertial frame. This means that the point P has a linear velocity

$$V_{P(inI)} = \omega \otimes r_R$$

And we can write:

$$\frac{d^I \vec{r}}{dt} = \left(\frac{dx_R}{dt} u_{1R} + \frac{dy_R}{dt} u_{2R} + \frac{dz_R}{dt} u_{3R} \right) + (x_R (\omega \otimes u_{1R}) + y_R (\omega \otimes u_{2R}) + z_R (\omega \otimes u_{3R}))$$

$$\frac{d^I \vec{r}}{dt} = \frac{d^R \vec{r}}{dt} + \omega \otimes \vec{r}_{(inR)}$$

That is the expression of the Transport Theorem.

Two easy scenarios are investigated hereafter:

Scenario 1: point P is not moving on the inertial frame. The inertial observer does not see any change along time. The observer in the rotational frame sees the point rotate in the opposite direction

$$\begin{aligned}\frac{d^I \vec{r}}{dt} &= \frac{d^R \vec{r}}{dt} + \omega \otimes \vec{r}_{(inR)} \\ 0 &= \frac{d^R \vec{r}}{dt} + \omega \otimes \vec{r}_{(inR)} \\ \frac{d^R \vec{r}}{dt} &= -\omega \otimes \vec{r}_{(inR)}\end{aligned}$$

Scenario 2: point P is rotating in a circle around the inertial frame with angular velocity ω . Suppose the rotational frame is attached to the point P (so rotates around the inertial frame at the same angular velocity)

$$\frac{d^I \vec{r}}{dt} = \frac{d^R \vec{r}}{dt} + \omega \otimes \vec{r}_{(inR)} = 0 + \omega \otimes \vec{r}$$

That is simply a circular motion.

8.4.3 Expression of angular velocities in function of inertial variables

We will use the general expression of the transport theorem to transform an arbitrary angular position from the body frame to an inertial frame. In particular, we are interested in the particular transformation from the body frame to the Guidance frame. It shall be noted that the Guidance frame rotates (with angular speed p along the X axis) but the Y and Z axis of the Guidance frame remains inertial.

$$\begin{aligned}\frac{d_I(\text{angPos}_{BinG})}{dt} &= \frac{d_B(\text{angPos}_{BinG})}{dt} + \omega \otimes (\text{angPos}_{BinG}) \\ \begin{bmatrix} \dot{\phi} \\ \dot{\psi} \\ \dot{\mathcal{G}} \end{bmatrix}_G &= \begin{bmatrix} \dot{\phi} \\ \dot{\psi} \\ \dot{\mathcal{G}} \end{bmatrix}_B + \omega \otimes \begin{bmatrix} \phi \\ \psi \\ \mathcal{G} \end{bmatrix}_B = \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \otimes \begin{bmatrix} \phi \\ \psi \\ \mathcal{G} \end{bmatrix}_B \\ \begin{bmatrix} \dot{\phi} \\ \dot{\psi} \\ \dot{\mathcal{G}} \end{bmatrix}_G &= \begin{bmatrix} p \\ q \\ r \end{bmatrix}_B + \begin{bmatrix} q\mathcal{G}_B - r\psi_B \\ -(p\mathcal{G}_B - r\phi_B) \\ p\psi_B - q\phi_B \end{bmatrix}\end{aligned}$$

We have required that Guidance frame have the same roll orientation that the LV Body frame, so that $\phi_B = -\phi_G$ and equal to zero. Also we know that the angle (Body as seen in Guidance) is the opposite of the angle (Guidance as seen in Body), so

$$\begin{aligned}\psi_B &= -\psi_G \\ \vartheta_B &= -\vartheta_G\end{aligned}$$

We are interested only in Y and Z axis so we remove the expression for phi.

$$\begin{bmatrix} \dot{\psi} \\ \dot{\vartheta} \end{bmatrix}_G = \begin{bmatrix} q \\ r \end{bmatrix}_B + \begin{bmatrix} -p\vartheta_B \\ p\psi_B \end{bmatrix} = \begin{bmatrix} q \\ r \end{bmatrix} + \begin{bmatrix} p\vartheta_G \\ -p\psi_G \end{bmatrix}$$

So we can express the angular velocities q, r of the body frame as the derivative of the angles in the Guidance frame and the angular position and roll rate of the body frame.

$$\begin{aligned}q &= \dot{\psi}_G - p\vartheta_G \\ r &= \dot{\vartheta}_G + p\psi_G\end{aligned}$$

8.4.4 Expression of angle of attack in inertial frame

We want to express the angle of attack in function of inertial variables.

$$\begin{aligned}v_y &= -\vartheta_G V + \dot{Y} \\ v_z &= \psi_G V + \dot{Z}\end{aligned} \quad \text{can be expressed as} \quad \begin{aligned}\alpha_\vartheta &= -\vartheta_G + \frac{\dot{Y}}{V} \\ \alpha_\psi &= \psi_G + \frac{\dot{Z}}{V}\end{aligned}$$

The demonstration is purely geometric.

Demonstration for pitch plane

The vector V is the relative velocity (i.e. velocity of the LV taking into account wind effects). The vector V has component \dot{Y}_{dot} in the inertial frame. The vector V has component v_y in the body frame.

The angle of attack in pitch plane is the angle between the velocity vector V and X_b . Visually this angle is the difference between the angle of V in inertial frame minus the angle of the body frame in the inertial frame.

See (Figure 8-13).

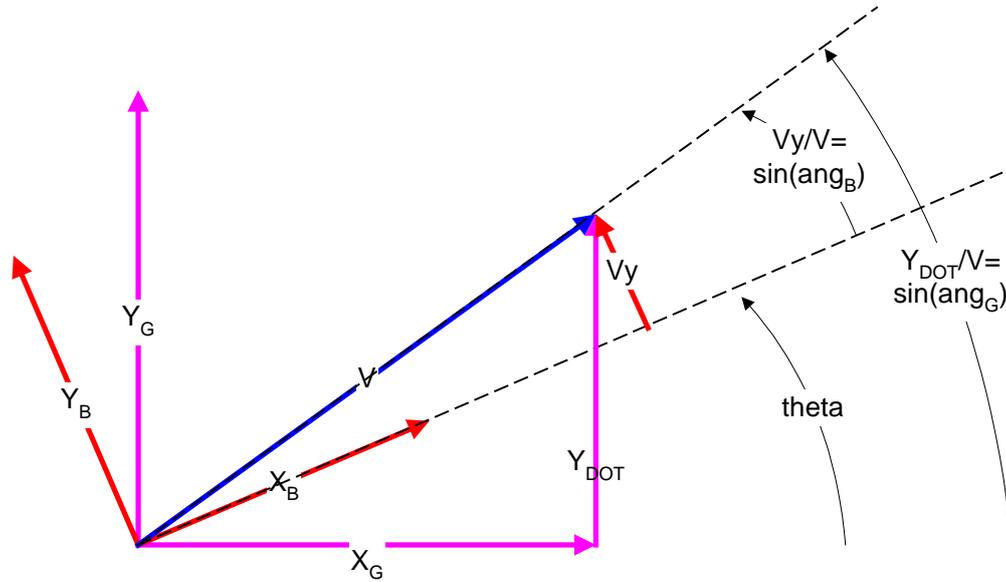


Figure 8-13: Angle of attack (theta)

$$ang_G = \vartheta_G + ang_B$$

$$\arcsin\left(\frac{\dot{Y}}{V}\right) = \vartheta_G + \arcsin\left(\frac{v_y}{V}\right)$$

For small v_y with regard V and small theta:

$$\frac{\dot{Y}}{V} = \vartheta_G + \frac{v_y}{V}$$

$$\dot{Y} = \vartheta_G V + v_y$$

$$v_y = -\vartheta_G V + \dot{Y}$$

Defining,

$$\alpha_g = \arcsin\left(\frac{v_y}{V}\right)$$

We arrive immediately to the expression for the angle of attack:

$$\arcsin\left(\frac{\dot{Y}}{V}\right) = \vartheta_G + \arcsin\left(\frac{v_y}{V}\right)$$

$$\arcsin\left(\frac{\dot{Y}}{V}\right) = \vartheta_G + \alpha_g$$

$$\alpha_g = -\vartheta_G + \frac{\dot{Y}}{V}$$

Demonstration for yaw plane.

The vector V is the relative velocity (i.e. velocity of the LV taking into account wind effects). The vector V has component Z_{dot} in the inertial frame. The vector V has component v_z in the body frame.

The angle of attack in yaw plane is the angle between the velocity vector V and X_b . Visually this angle is the difference between the angle of V in inertial frame minus the angle of the body frame in the inertial frame.

See (Figure 8-14).

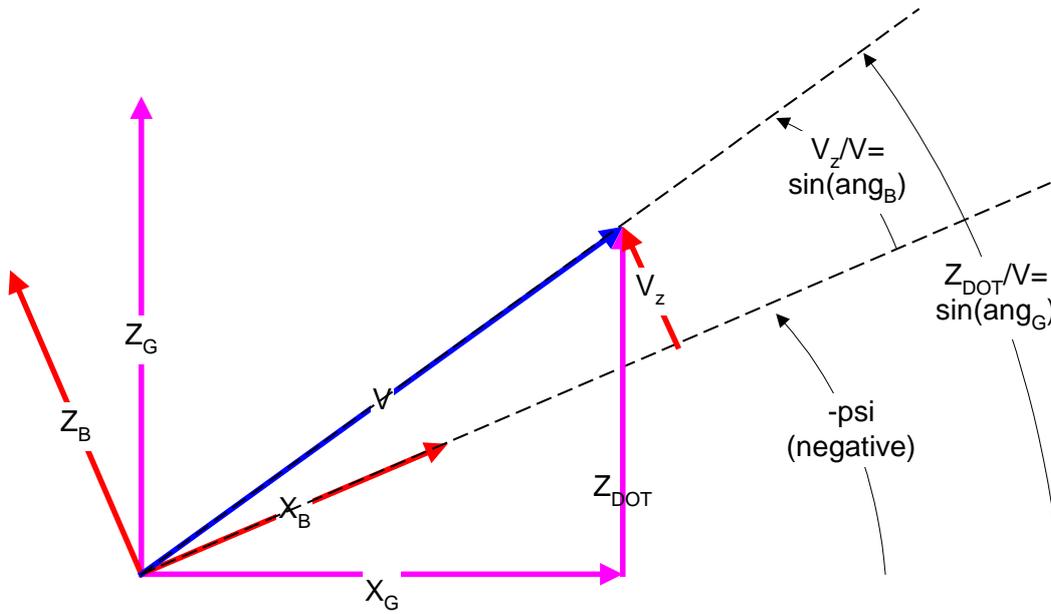


Figure 8-14: Angle of attack (psi)

Note however that due to right hand convention, Z_{dot} positive implies angle psi (inertial to body) negative.

$$ang_G = -\psi_G + ang_B$$

$$\arcsin\left(\frac{\dot{Z}}{V}\right) = -\psi_G + \arcsin\left(\frac{v_z}{V}\right)$$

For small v_z with regard V and small deviation between axis, angles are small

$$\frac{\dot{Z}}{V} = -\psi_G + \frac{v_z}{V}$$

$$\dot{Z} = -\psi_G V + v_z$$

$$v_z = \psi_G V + \dot{Z}$$

Defining,

$$\alpha_\psi = \arcsin\left(\frac{v_z}{V}\right)$$

We arrive immediately to the expression for the angle of attack:

$$\arcsin\left(\frac{\dot{Z}}{V}\right) = -\psi_G + \arcsin\left(\frac{v_z}{V}\right)$$

$$\arcsin\left(\frac{\dot{Z}}{V}\right) = \psi_G + \alpha_\psi$$

$$\alpha_\psi = \psi_G + \frac{\dot{Z}}{V}$$

8.4.5 Check of rotational and translational formulas under rotation of 90 degrees

By symmetry of the Launcher, we know that under a positive turn of 90 deg on the roll axis, the dynamics is identical (i.e. the selection of the Y and Z axis is just a convention). This symmetry shall be observed on the equations under such turn (Figure 8-15).

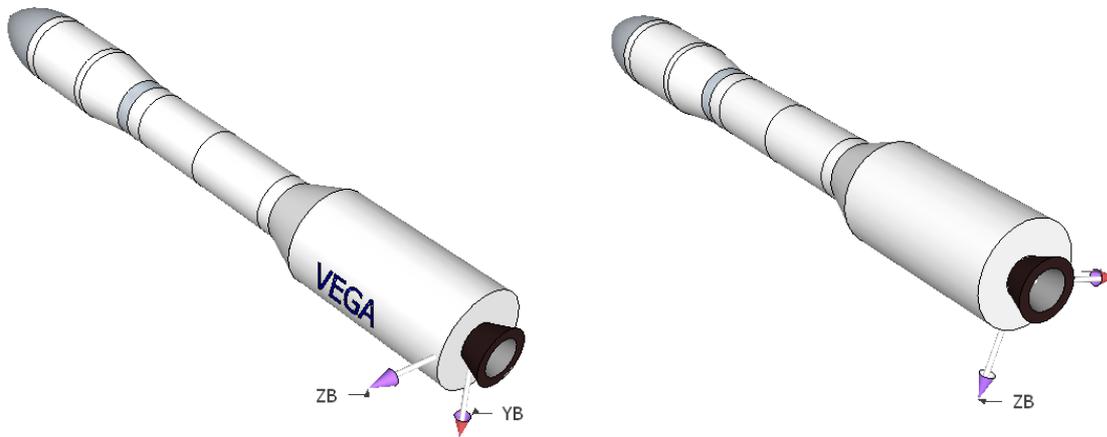


Figure 8-15: Rotation of LV around x axis (-90 deg)

Under this rotation, we can the following transformations:

$z \rightarrow y$	$\theta \rightarrow \psi$
$y \rightarrow -z$	$\psi \rightarrow -\theta$
$\beta_\theta \rightarrow \beta_\psi$	$r \rightarrow q$
$\beta_\psi \rightarrow -\beta_\theta$	$q \rightarrow -r$

Note that the Thrust (T), the transversal inertia (I_T), dynamic pressure (p_{dyn}), coefficients, X axis parameters, etc., are unchanged under this rotation. Accordingly the following expressions are unchanged under this rotation.

$$A_6 = \frac{1}{I_T}(x_{CP} - x_{COG})(p_{dyn}S_R C_N)$$

$$K_1 = -\frac{T}{I_T}(x_{COG} - x_{PP})$$

$$\lambda = \frac{I_T - I_{XX}}{I_T} = 1 - \frac{I_{XX}}{I_T}$$

Rotational formula

If we substitute in the original equation,

$$\dot{q} = (K_1 \beta_\psi) + A_6 \psi + \left(\frac{A_6}{V}\right) \dot{Z} + \lambda p_0 r$$

$$\dot{r} = (K_1 \beta_g) + A_6 \vartheta - \left(\frac{A_6}{V}\right) \dot{Y} - \lambda p_0 q$$

We have:

$$(-\dot{r}) = (K_1 (-\beta_g)) + A_6 (-\vartheta) + \left(\frac{A_6}{V}\right) (\dot{Y}) + \lambda p_0 (q)$$

$$(\dot{q}) = (K_1 (\beta_\psi)) + A_6 (\psi) - \left(\frac{A_6}{V}\right) (-\dot{Z}) - \lambda p_0 (-r)$$

And then the original equation is obtained:

$$(\dot{r}) = (K_1 (\beta_g)) + A_6 (\vartheta) - \left(\frac{A_6}{V}\right) (\dot{Y}) - \lambda p_0 (q)$$

$$(\dot{q}) = (K_1 (\beta_\psi)) + A_6 (\psi) + \left(\frac{A_6}{V}\right) \dot{Z} + \lambda p_0 (r)$$

Translational formula

If we substitute in the original equation,

$$\ddot{Y} = \frac{T}{m} \beta_g + \frac{T - p_{dyn} S_R (C_N - C_X)}{m} \vartheta - \frac{p_{dyn} S_R C_N}{mV} \dot{Y} + 2p \dot{Z} - Vr + Vp \psi$$

$$\ddot{Z} = -\frac{T}{m} \beta_\psi - \frac{T - p_{dyn} S_R (C_N - C_X)}{m} \psi - \frac{p_{dyn} S_R C_N}{mV} \dot{Z} - 2p \dot{Y} + Vq + Vp \vartheta$$

We have:

$$(-\ddot{Z}) = \frac{T}{m} (\beta_\psi) + \frac{T - p_{dyn} S_R (C_N - C_X)}{m} (\psi) - \frac{p_{dyn} S_R C_N}{mV} (-\dot{Z}) + 2p (\dot{Y}) - V(q) + Vp (-\vartheta)$$

$$(\ddot{Y}) = -\frac{T}{m} (-\beta_g \vartheta) - \frac{T - p_{dyn} S_R (C_N - C_X)}{m} (-\vartheta) \psi - \frac{p_{dyn} S_R C_N}{mV} (\dot{Y}) - 2p (-\dot{Z}) + V(-r) + Vp (\psi)$$

And then the original equation is obtained:

$$\begin{aligned}(\ddot{Z}) &= -\frac{T}{m}(\beta_\psi) - \frac{T - p_{dyn}S_R(C_N - C_X)}{m}(\psi) - \frac{p_{dyn}S_R C_N}{mV}(\dot{Z}) - 2p(\dot{Y}) + V(q) + Vp(\mathcal{G}) \\(\ddot{Y}) &= \frac{T}{m}(\beta_\vartheta) + \frac{T - p_{dyn}S_R(C_N - C_X)}{m}(\vartheta) - \frac{p_{dyn}S_R C_N}{mV}(\dot{Y}) + 2p(\dot{Z}) - V(r) + Vp(\psi)\end{aligned}$$

This demonstrates the coherency of the linear model with the physic of the problem.

8.4.6 Quaternions

Quaternions where introduced in 1847 by the English mathematician W. R. Hamilton as a way of generalize the algebra of complex numbers to higher dimensions. The quaternion is defined as:

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$$

With the following rules

$$\begin{aligned}i^2 = j^2 = k^2 &= -1 \\ij = k = -ji, \quad jk = i = -kj, \quad ki = j = -ik\end{aligned}$$

The sum of quaternions is the usual sum of vector on R^4 .

$$(p_1, p_2, p_3, p_4) + (q_1, q_2, q_3, q_4) = (p_1+q_1, p_2+q_2, p_3+q_3, p_4+q_4)$$

And the multiplication of quaternions is given by:

$$(a, \mathbf{x})(b, \mathbf{y}) = (ab - \mathbf{x} \cdot \mathbf{y}, a\mathbf{y} + b\mathbf{x} + (\mathbf{x} \times \mathbf{y}))$$

Than can be expressed as:

$$q_1 q_2 = \begin{bmatrix} q_1(1) & -q_1(2) & -q_1(3) & -q_1(4) \\ q_1(2) & q_1(1) & -q_1(4) & q_1(3) \\ q_1(3) & q_1(4) & q_1(1) & -q_1(2) \\ q_1(4) & -q_1(3) & q_1(2) & q_1(1) \end{bmatrix} q_2$$

A quaternion represents a rotation of an angle θ around an axis “a”:

$$q = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ a_1 \sin\left(\frac{\theta}{2}\right) \\ a_2 \sin\left(\frac{\theta}{2}\right) \\ a_3 \sin\left(\frac{\theta}{2}\right) \end{bmatrix}$$

In general is not easy to interpret the quaternions in term of Euler angles but some cases allows a direct interpretation.

$$q = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ a_1 \sin\left(\frac{\theta}{2}\right) \\ a_2 \sin\left(\frac{\theta}{2}\right) \\ a_3 \sin\left(\frac{\theta}{2}\right) \end{bmatrix} = \begin{bmatrix} \sqrt{1 - \sin^2\left(\frac{\theta}{2}\right)} \\ a_1 \sin\left(\frac{\theta}{2}\right) \\ a_2 \sin\left(\frac{\theta}{2}\right) \\ a_3 \sin\left(\frac{\theta}{2}\right) \end{bmatrix}$$

For small θ , the quaternion becomes:

$$q \approx \begin{bmatrix} 1 \\ \frac{\theta}{2} \\ a_1 \frac{\theta}{2} \\ a_2 \frac{\theta}{2} \\ a_3 \frac{\theta}{2} \end{bmatrix}$$

This means that a (normalized) quaternion with first component similar to 1 is a small rotation (of “ $\theta/2$ ” around the vector “ a ”).

If only one of the components 2 to 4 is not zero, the quaternion represents a rotation around that axis. Example, a rotation of 30 deg around the axis x becomes:

$$q = \begin{bmatrix} \cos(30/2) \\ 1 * \sin(30/2) \\ 0 * \sin(30/2) \\ 0 * \sin(30/2) \end{bmatrix} = \begin{bmatrix} 0.9659 \\ 0.2588 \\ 0 \\ 0 \end{bmatrix}$$

Rotation of a vector by means of a quaternion

Given a vector (V_A) in a frame A we can express it as a vector (V_B) in a frame B if we know the quaternion q_{BA} (quaternion to pass from A to B) just by applying the matrix transformation:

$$V_B = q_{BA} V_A q_{BA}^{-1}$$

Quaternion derivative

The derivative of the quaternion is defined as:

$$\dot{q} = \frac{1}{2} q_\omega q$$

Where q_ω is a 'fake' quaternion built with ω as:

$$q_\omega = [0 \ \omega(1) \ \omega(2) \ \omega(3)]';$$

Demonstration:

The relation is easy to demonstrate starting from the expression of the quaternion:

$$q(\vartheta) = \begin{bmatrix} \cos\left(\frac{\vartheta}{2}\right) \\ a \sin\left(\frac{\vartheta}{2}\right) \end{bmatrix} \quad \text{given } \vartheta(t) = \omega t$$

Taking the derivative:

$$\dot{q} = \begin{bmatrix} -\frac{\omega}{2} \sin\left(\frac{\omega}{2}\right) \\ a \frac{\omega}{2} \cos\left(\frac{\omega}{2}\right) \end{bmatrix}$$

And then:

$$\dot{q} = \frac{\omega}{2} \begin{bmatrix} -\sin\left(\frac{\vartheta}{2}\right) \\ a \cos\left(\frac{\vartheta}{2}\right) \end{bmatrix} = \frac{\omega}{2} \begin{bmatrix} -\cos\left(\frac{\vartheta}{2} + \frac{\pi}{2}\right) \\ a \sin\left(\frac{\vartheta}{2} - \frac{\pi}{2}\right) \end{bmatrix} = \frac{\omega}{2} \begin{bmatrix} \cos\left(\frac{\vartheta}{2} - \frac{\pi}{2}\right) \\ a \sin\left(\frac{\vartheta}{2} - \frac{\pi}{2}\right) \end{bmatrix} = \frac{\omega}{2} \begin{bmatrix} \cos\left(\frac{\vartheta - \pi}{2}\right) \\ a \sin\left(\frac{\vartheta - \pi}{2}\right) \end{bmatrix}$$

Identifying with the general expression of a quaternion:

$$\dot{q} = \frac{\omega}{2} q(\vartheta - \pi) = \frac{\omega}{2} q(\vartheta) q(-\pi) = \frac{1}{2} \omega q$$

Note: $q(\pi) = q(-\pi) = \text{unit quaternion } ([1 \ 0 \ 0 \ 0])$.

9 Appendix: Developed Software

9.1 Software for the CDC benchmark

Note: the scripts have been tested only with Matlab 2011b.

9.1.1 Software User Manual Notes

9.1.1.1 Execution

Open a Matlab session and change to the directory containing the code.

Execute the following command to design the controllers:

```
>> doBench
```

(Edit the file doBench for comment or uncomment the desired controllers to design)

The execution generates a html file with the controller performances (in folder 'rpt', name is '<CONTROLLER_NAME>_spec.html')

Execute the following command for printing the figures and generate a table with the controller performances:

```
>> printFig
```

9.1.1.2 List of important files

File	Purpose
doBench	Entry file that generates the controllers and executes the CDC benchmark (Edit the file and uncomment the wanted controllers)
la.m	Generates the open loop plan (nominal and disturbed)
graf.m	Executes the CDC for the controller passed, generating also the figures
printFig.m	Auxiliary files for generating HTML and png files
Aug_xxx.m	Auxiliary files for generating the augmented plants
khinf_mixed	Generates the H-Infinity controller Mixed Sensitivity
khinf_kwa	Generates the H-Infinity controller KWA
khinf_gst1 khinf_gst2	Generates the H-Infinity controller CHGE Variants 1 and 2

khinf_struct_3	Generates the H-Infinity Structured controller
kmuSKO96	Generates the mu controller Skogestad 96
kmuLUND99	Generates the mu controller Lundstrom 99
kmu2012	Generates the mu controller Balas 2012

9.2 Software for the VEGA Launcher

Note: the scripts and “miniVEGA” have been tested only with Matlab 2011b.

9.2.1 Software User Manual Notes

9.2.1.1 Execution

Open a Matlab session and change to the directory containing the code.

Edit the file “design.m” commenting out the desired controllers.

Execute it for designing the selected controllers:

```
>> desi gnK
```

Edit the file “runBatchSim.m” and select the desired scenario. Execute the scenarios with:

```
>> runBat chSi m
```

Plot the desired results with

```
>> pl otSi m(l i stSi m)
```

9.2.1.2 List of important files

File	Purpose
designK	Entry file that executes a simulation (See detailed description)
loadParamsFromVMAT	Returns a set of flight parameters linearized between times passed. Returns both the mean value and a vector with all the values
createOLPlant	Creates a LV plant (open loop) Returns: The TVC model The 6DoF model The full model sys = TVC * 6DoF

create6DoF	Creates a the 6DoF model Note the states are % states = [yDot theta r zDot psi q (y z)] % actuators = [betaTetha betaPsi]
createKPIDELV	Creates several ELV controllers (classical PID control plus advanced filters) Control on theta, psi
createTFs(K,G)	Creates the closed loop functions for the controller and plant passed
getELVTuning	Retrieves the tuning coefficients of gains and filters for the PID ELV controller at the time passed. The filters are defined in the digital domain. They are passed to continuous domain with the function d2c() and the 'Tustin' approach.
createHINFGST	Creates an H-Infinity CHGE controller
createHINFStruct	Creates an H-Infinity Structured controller
runBatchSim	Executes a set of simulations in Simulink (by calling runSim) (See detailed description)
runSim	Executes one simulation in Simulink (See detailed description)
plotSim	Do simulation plots (See detailed description)

9.2.2 Function details

The more important functions are described with some detail.

9.2.2.1 Function designK()

It is the main function for designing the controllers.

The script declares 3 global structs:

- Struct "DESIGN" that will store each designed controller
- Struct "SIM" that will store input and output data for the Simulink model
- Struct "SIMDATA" that will store the results of each simulation.

First the timeBegin and timeEnd for designing the controllers are chosen. These times are used:

- At design time for linearizing the 6DoF model
- At run time for providing the interpolation data for the Simulink block "Interpolated time variables"

A typical choice is timeBegin = 50, timeEnd = 65 to capture the zone of maximum dynamic pressure.

Then the data of the LV is loaded from a predefined simulation by calling the dedicated functions “loadParamFromVMAT()”. This function returns 2 structs. The struct “par” contains the average value of parameters between begin and end times (e.g. average mass, average inertia). The struct “parAll” contains instead all the samples between these times. The struct “parAll” is used at run time.

The data of the environment (atmosphere, winds) is loaded by calling the functions “loadAeroCoeff()” and “loadAtmoModel()”. These data are loaded in the struct “parAll”. These data is not used for design of the controllers but at run time during the 6DoF simulations.

A given roll rate is selected as “design roll rate”. The linear model will be instantiated with this roll rate.

Then the function “createOLPlant()” is called. It creates the linear open loop plant (composed of the 6DoF linearized model and the TVC actuators. The struct “par” with the linearization point is passed as argument.

Finally each controller is designed. The file is edited manually for commenting out the desired controllers.

Each controller is stored as a field of the global struct “DESIGN”.

9.2.2.2 Function runSim()

This script executes one simulation.

Receives as parameters:

- The structs “par” and “parAll” with the linearization points
- A label identifying the simulation to be generated
- The identifier of the controller to be used
- The name of the controller to use and the simulation options
- The simulation scenario

The simulation scenario is blended with default options as:

- Default duration
- Default roll rate
- Default delays

(i.e. the caller can overwrite the value for a default option. If not done, the default value is used).

Then several fields are added to the global struct SIM that is visible inside the Simulink model, for example:

- SIM.par that contains the parameters of the model
- SIM.TVC that contains the state space equations of the TVC

- SIM.K*** that contains the state space equations of the designed controllers

It is important to mention that the initial LV state (at the start of the simulation) is set in some SIM.par.xxx variables:

- Angular velocity in Body (initial value of the angular velocity integrator)
- Quaternion at the start of the simulation. It is the unit quaternion, i.e. at the instant of the simulation we consider that the LV body frame is aligned with the LV reference frame
- Linear velocity at the start of the simulation

The Simulink model (by default “miniVEGA.mdl”) is opened. Several parameters are adjusted as “SaveFormat”, “solver” etc.

Then, a placeholder block called “miniVEGA/GNC” is replaced with the designed controller “miniVEGA/GNC/<controllerID>”. Not that the model is edited dynamically for replacing the generic placeholder controller with the desired one.

Then the simulation is executed by called the function “sim()” passing the desired simulation time and simulation options (as for example the desired solver).

When the simulation finish, the simulations outputs are copied to the based workspace, to the struct SIM.outputs. Some derived outputs are computed (for example the error on attack angle).

Finally, the full SIM struct is saved as field of a global struct called SIMDATA. (E.g. SIMDATA.KELVI). In addition, each simulation is saved to a particular file.

9.2.2.3 Function runBatchSim

This script allows the automation of the execution of a set of simulations.

A set of scenarios are defined. Each simulation scenario defines:

- The name of the scenario
- The model to use (6DoF nonlinear or linear)
- The input references to use
- The wind profile

The cellarray “listSce” contains the scenarios to execute. The cellarray “listControllers” contains the controllers to simulate. Finally the array “listRoll” contains the roll rates.

A loop invokes the function runSim():

- For each scenario in “listSce”
- For each controller in “listControllers”
- For each roll rate in “listRoll”

The names of the simulations executed are stored in the global array `listSim[]`.

9.2.2.4 Function `plotSim()`

This function plots several figures for the simulations contained in the list passed.