Advanced Control Systems for Fast Orbit Feedback of Synchrotron Electron Beams

Idris Kempf
Lady Margaret Hall
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy
Hilary Term, 2023

To my father, in loving memory.

Acknowledgements

I extend my deepest gratitude to my DPhil supervisors, Professor Paul Goulart and Professor Stephen Duncan, for their invaluable guidance and unwavering support throughout my four and a half years of doctoral research.

Equally, I am grateful to my supervisors at Diamond Light Source, Dr. Lorraine Bobb, Dr. Michael Abbott, and Dr. Günther Rehm, for their tremendous generosity with their time and knowledge.

My sincere thanks also go to Dr. Glenn Christian, with whom I spent many night-shifts and weekends testing my algorithms at Diamond Light Source. His contributions were essential to the successful demonstration of their application.

I would like to express my appreciation to current and past members of the Control Group, especially Professor Ross Drummond, Dr. Nikitas Rontsis, and Yana Lishkova, for their kindness and inspiring conversations over the years.

I am grateful for the financial support provided by Diamond Light Source and the Engineering and Physical Sciences Research Council.

Lastly, I would like to thank my mother, Nadia Kempf, and my sister, Anissa Kempf, for their love, patience and support, which allowed me to focus on my DPhil studies.

Abstract

Diamond Light Source is the UK's national synchrotron facility that produces synchrotron radiation for research. At source points of synchrotron radiation, the electron beam stability relative to the beam size is critical for the optimal performance of synchrotrons. The current requirement at Diamond is that variations in the beam position should not exceed 10 % of the beam size for frequencies up to 140 Hz. This is guaranteed by the fast orbit feedback that actuates hundreds of corrector magnets at a sampling rate of 10 kHz to reduce beam vibrations down to sub-micron levels. For the next-generation upgrade, Diamond-II, the beam stability requirements will be raised to 3 % up to 1 kHz. Consequently, the sampling rate will be increased to 100 kHz and an additional array of fast correctors will be introduced, which precludes the use of the existing controller. This thesis develops two different control approaches to accommodate the additional array of fast correctors at Diamond-II: internal model control based on the generalised singular value decomposition (GSVD) and model predictive control (MPC). In contrast to existing controllers, the proposed approaches treat the control problem as a whole and consider both arrays simultaneously. To achieve the sampling rate of 100 kHz, this thesis proposes to reduce the computational complexity of the controllers in several ways, such as by exploiting symmetries of the magnetic lattice. To validate the controllers for Diamond-II, a real-time control system is implemented on high-performance hardware and integrated in the existing synchrotron. As a first-of-its-kind application to electron beam stabilisation in synchrotrons, this thesis presents real-world results from both MPC and GSVD-based controllers, demonstrating that the proposed approaches meet theoretical expectations with respect to performance and robustness in practice. The results from this thesis, and in particular the novel GSVD-based method, were successfully adopted for the Diamond-II upgrade. This may enable the use of more advanced control systems in similar large-scale and high-speed applications in the future.

Contents

List of Figures

Notation

Sets

$\mathbb{R}$ ($\mathbb{C}$)	the set of real (complex) numbers
$\mathbb{R}_+$ ($\mathbb{R}_{++}$)	the nonnegative (positive) real numbers
$\mathbb{R}^{m\times n}$ ($\mathbb{C}^{m\times n}$)	the set of real (complex) $m \times n$ real matrices
$\mathbb{Z}$	the set of integers $\{\dots, -2, -1, 0, 1, \dots\}$
$\mathbb{S}^n_{++}$ ($\mathbb{S}^n_+$)	the set of real symmetric positive (semi)definite matrices

Relations

$A \coloneqq B$	A is defined by B
$A \succ (\succeq) B$	$A - B$ is positive (semi)definite

Linear Algebra

$\|A\|_p$	(Induced) p–norm of a vector or matrix A
$\|A\|_F$	Frobenius norm of a matrix X
$\|x\|_Q^2$	2–norm of a vector x weighted by Q: $\|x\|_Q^2 \coloneqq \|x^{\mathrm{T}} Q x\|_2$
$\mathbf{1}_{n\times m}$	$n \times m$ matrix of ones
I	identity matrix of appropriate dimensions
A^{T} (A^*)	transpose (Hermitian transpose) of a vector or matrix A
$\mathrm{diag}(\{x_i\}_{i\in\mathcal{E}})$	diagonal concatenation of vectors (or matrices) x_i
$\ker(A)$	kernel (nullspace) of a matrix A
$\mathrm{range}(A)$	range of a matrix A
$\mathrm{eig}_i(A)$	i–th eigenvalue of A
$\rho(A)$	spectral radius of A: $\max_i \lvert\mathrm{eig}_i(A)\rvert$
$\sigma_i(A)$	i–th largest singular value of A
$\otimes$ ($\oplus$)	Kronecker product (sum)
$\odot$	Hadamard (elementwise) product

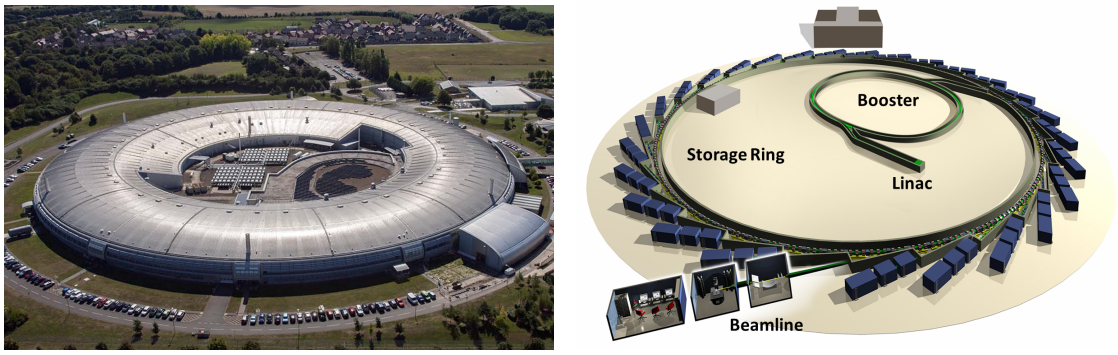
Acronyms

ADMM	Alternating directions method of multipliers
ASD	Amplitude spectral density
BPM	Beam position monitor
CD	Cross-directional
CQP	Constrained quadratic program
DFT	Discrete Fourier transformation
FGM	Fast gradient method
FOFB	Fast orbit feedback
FPGA	Field programmable gate array
GFLOPS	Giga floating-point operations per second
HO-GSVD	Higher-order generalised singular value decomposition
IIR	Infinite impulse response
IMC	Internal model control
LMI	Linear matrix inequality
LQG	Linear quadratic Gaussian (control)
MIMO	Multiple-input, multiple-output
MPC	Model predictive control
ORM	Orbit response matrix
PSD	Power spectral density
RMS	Root-mean-square
SDP	Semidefinite program
SISO	Single-input, single-output
TISO	Two-inputs, single-output

1 Introduction

A synchrotron light source is a particle accelerator in which electrons travel at relativistic speeds around a circular path called the *storage ring*. In most synchrotrons, the electrons' trajectories are bent using magnetic fields and synchrotron radiation is emitted when the electrons are accelerated in a direction perpendicular to their trajectory [165, Ch. 8.1.1]. The synchrotron radiation spans the electromagnetic spectrum from infrared to X-rays and is transported in *beamlines* to experimental stations, where it is used for various scientific techniques, such as microscopy, scattering, diffraction and spectroscopy. These techniques find application in various scientific and industrial fields, such as material science, chemistry, medicine, biology, molecular and condensed-matter physics [165, Ch. 13.5].

An important figure of merit for synchrotron light sources is the *spectral brightness* (or brilliance) [164, Ch. 24.8]. The spectral brightness is proportional to the electron beam energy and inversely proportional to the (horizontal and vertical) beam emittance, which is the product of the transverse beam size and divergence, representing a measure of "spread" of the beam cross-section [165]. To produce bright photon beams that can be easily manipulated in the beamlines, synchrotron light sources typically aim at having a low emittance combined with a high-energy electron beam.



(a) Aerial view. **(b)** Layout.

Figure 1.1: Aerial view of Diamond Light Source (a) and layout (b) showing the linear accelerator (Linac), booster, storage ring and beamlines. Figures used with permission from Diamond Light Source.

1.1 Diamond Light Source

Diamond Light Source is the UK's national synchrotron facility located in Didcot, Oxfordshire. Diamond currently operates 32 beamlines that are distributed along a 561.6 m circumference storage ring. An aerial view and a simplified layout of the facility are shown in Fig. 1.1. The electrons are produced by an electron gun and accelerated in the *linear accelerator* to an energy of 100 MeV by passing through a series of accelerating structures [165, Ch. 1.4]. The electrons are then injected into the *booster*, which is a 158 m circumference synchrotron that accelerates the electrons to their final energy of 3 GeV. After extraction from the booster, the electrons traverse a 68 m long transfer line to the storage ring injector, which uses a septum magnet and four kicker magnets for injecting bunches of electrons into the storage ring [25]. To maintain an operating beam current of 300 mA, the electron beam requires top-up every 10 min.

The left-hand side of Fig. 1.2 shows an overview of the machine components in the storage ring and the right-hand side the trajectory-based coordinate system. The electron beam follows a curved path in the horizontal plane and is confined in an elliptically-shaped vacuum vessel with nominal axes of 52 mm × 24 mm and an operating pressure of less than 10^{-9} mbar [47, Ch. 2.4.5]. The storage ring is built up from 24 cells that consist of straight sections angled together using arc

sections. Dipole magnets deflect the trajectory of the beam in the longitudinal direction and quadrupole and sextupole magnets focus the beam in the transverse direction. The magnets are mounted onto girders and the pattern of bending and focusing magnets, called lattice, is repeated around the storage ring. The lattice determines the *betatron function*, which in turn determines the oscillatory trajectory of the electrons, the number of oscillations per turn, called the *betatron tune*, and the elliptical cross-section of the beam [165, Ch. 3]. At Diamond, the lattice is a double bend achromat lattice [24, p. 2.2.4], with the exception of one cell that has been changed to a double-double bend achromat for experimental purposes [10]. In addition to the lattice magnets, each cell contains a regular number of smaller dipole magnets, referred to as *corrector magnets*, which are used to correct the beam trajectory in the horizontal and vertical directions.

The synchrotron radiation is extracted at different points around the storage ring. For some beamlines, the source point are bending magnets, whereas for other beamlines, the source points are *insertion devices* (IDs) that trigger the emission of synchrotron radiation, such as *undulators* and *wigglers* [24, Ch. 3.1.5]. IDs are magnet structures that cause transverse oscillations and stimulate the emission of synchrotron radiation, whose intensity can be varied by changing the so-called ID gap. To compensate for the energy emitted in the form of synchrotron radiation, the electron beam is re-accelerated in a radio-frequency (RF) cavity at every turn.

1.2 Fast Orbit Feedback

Like most synchrotrons, Diamond uses various feedback systems to control different beam characteristics and actively compensate for disturbances and instabilities [121]. An overview of the feedback systems in the storage ring is given in Table 1.1. The transverse and longitudinal multi-bunch feedback systems (TMBF and LMBF) run at 500 MHz and suppress instabilities caused by the interaction of electron bunches [24, Ch. 7.5.2]. The position of individual bunches is measured using 4 capacitive button pick-ups (electrodes) surrounding the beam in a vessel cross-section. The transverse position is corrected using kicker striplines [24, Ch. 7.2.7] and the longitudinal

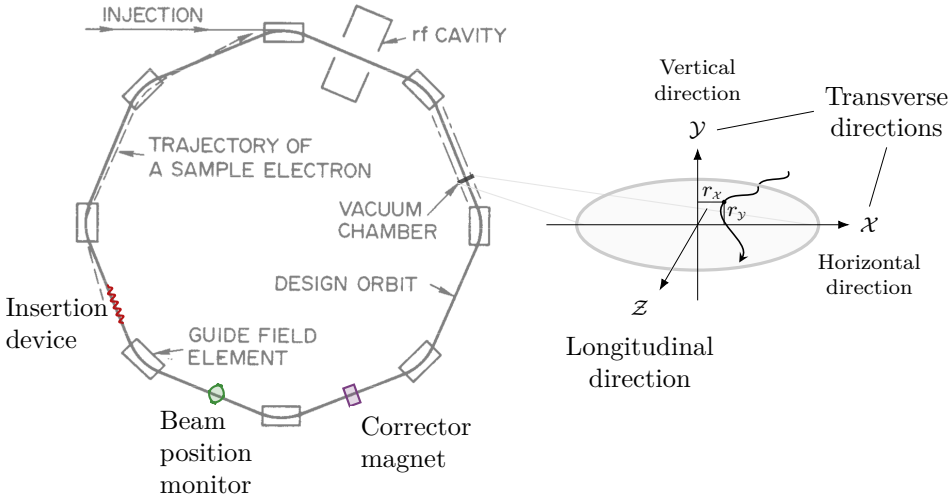


Figure 1.2: Overview of machine components in the storage ring adapted from [126] (left) with cross-sectional coordinate system (right).

Table 1.1: Feedback systems and actuation frequencies in the Diamond Light Source storage ring with the transverse and longitudinal multi-bunch feedback systems (TMBF and LMBF), slow and fast orbit feedback (SOFB and FOFB) and the vertical emittance feedback (VEFB).

Feedback	Actuators	Feedback Signal	Frequency	
TMBF [3]	Kicker striplines	Bunch position	500.0	MHz
LMBF [108]	Kicker cavity	Bunch position	500.0	MHz
FOFB [65]	Correctors	Beam position	10.0	kHz
VEFB [102]	Skew-quadrupoles	Beam size	5.0	Hz
Tune [100]	Quadrupoles	Tune	1.0	Hz
SOFB [162]	Correctors	Beam position	0.2	Hz
Dispersion [67]	RF frequency	Corrector currents	0.1	Hz

position using a kicker cavity [108]. The vertical emittance feedback (VEFB), which operates at 5 Hz, stabilises the vertical emittance and compensates for perturbations due to changes in ID gaps or long-term drifts [102]. The tune feedback runs at 1 Hz and applies corrections to the quadrupoles to maintain the betatron tune [100]. The dispersion correction feedback runs at 0.1 Hz and minimises disturbances introduced by the corrector magnets of the (slow or fast) orbit feedback [67].

Even though the electron beam is guided and confined by the magnetic lattices,

its trajectory is perturbed by disturbances caused by electromagnetic radiation, girder and machine component vibrations, or by machine operations, such as injection or ID gap changes [9]. Because the beam stability at source points of synchrotron radiation is critical for the preservation of low emittance [120], the orbit feedback attenuates the effect of disturbances onto the beam using position measurements and corrector magnets. To avoid saturation of the corrector magnets, the orbit feedback is separated into the slow orbit feedback (SOFB) running at a (configurable) frequency of 0.2 Hz and the fast orbit feedback (FOFB) running at 10 kHz [66]. The SOFB is used to reduce the beam displacement below 150 µm, before activating the FOFB.

The horizontal and vertical beam position is measured using sensors referred to as beam position monitors (BPMs), which, analogous to the multi-bunch measurements, consist of 4 capacitive button pick-ups in the vacuum vessel surrounding the electron beam in the transverse plane [24, Ch. 7.4.2]. The pick-ups sense the electric field of the beam and the 4 signals are multiplexed and processed to compute the horizontal and vertical beam positions, which are measured as the deviation from a pre-defined trajectory [66].

The trajectory of the beam is adjusted by specifying the current through the corrector magnets, whose magnetic fields deflect the beam in either horizontal or vertical direction. The coupling between the directions is negligible and horizontal and vertical directions are treated as two separate control problems [49]. At Diamond, each of the 24 cells initially contained 7 BPMs and 7 corrector magnets, but some BPMs and magnets have been added or removed for experimental reasons [10]. At present, the FOFB can use a configurable number of 172 BPMs and 2×168 corrector magnets for the vertical and horizontal directions.

The aim of the FOFB is to maintain the root means square (RMS) variation of the electron beam centroid within a maximum value typically specified as 10 % of the beam size at source points. For Diamond and in the centre of the insertion devices in the standard straight sections, this implies a maximum RMS variation of 12.3 µm in the horizontal and 0.6 µm in the vertical plane [124]. These

specifications exclude long-term disturbances, such as drifts caused by changes in air temperature and humidity.

1.2.1 Electron Beam Dynamics

In each of the transverse directions, the trajectory of the electrons is a phase and amplitude modulated oscillation of the form [165, Ch. 3.1]

$$r_{(\cdot)}(\mathcal{Z}) \coloneqq \sqrt{\beta_{(\cdot)}(\mathcal{Z})c_1}\cos\big(\phi_{(\cdot)}(\mathcal{Z}) + c_2\big) \tag{1.1}$$

where $c_1, c_2 \in \mathbb{R}_+$ are constants, $\beta_{(\cdot)} : \mathbb{R} \mapsto \mathbb{R}_+$ the *betatron function* [165, Ch. 2.1.3], $\phi_{(\cdot)} : \mathbb{R} \mapsto \mathbb{R}_+$ the phase advance, $(\cdot) = \{\mathcal{X}, \mathcal{Y}\}$ refers to the horizontal and vertical directions and $r_{(\cdot)}(\mathcal{Z})$ to the coordinates in Fig. 1.2. Superimposing the corrector magnets yields a linear map that describes the instantaneous effect of a corrector at longitudinal coordinate $\mathcal{Z}_1$ onto the beam position at $\mathcal{Z}_2$. Evaluating the linear map at the coordinates of the $n_{y,(\cdot)}$ BPMs and $n_{u,(\cdot)}$ correctors, where subscripts y and u refer to outputs and inputs, yields the *orbit response matrices* (ORM) $R_{\mathcal{X}} \in \mathbb{R}^{n_{y,\mathcal{X}} \times n_{u,\mathcal{X}}}$ and $R_{\mathcal{Y}} \in \mathbb{R}^{n_{y,\mathcal{Y}} \times n_{u,\mathcal{Y}}}$. Because the horizontal and the vertical motions are mutually independent, the subscripts $\mathcal{X}$ and $\mathcal{Y}$ will be dropped in the following and the ORM denoted by $R \in \mathbb{R}^{n_y \times n_u}$.

In practice, a high-level controller computes the corrector setpoints in Amperes and low-level power supply controllers drive the magnet currents to the desired setpoints producing a magnetic field in a particular direction. These and additional dynamics, such as Eddy currents in the vacuum chamber, are modelled to obtain the actuator model $g : \mathbb{C} \mapsto \mathbb{C}$, which for Diamond is given by the transfer function [52]

$$g(s) = \frac{a_{(\cdot)}}{s + a_{(\cdot)}}\mathrm{e}^{-\tau_d s}, \tag{1.2}$$

where $s \in \mathbb{C}$ is the Laplace variable, $\tau_d \coloneqq 900\,\mu\mathrm{s}$ the time delay, $a_{\mathcal{X}} \coloneqq 2\pi \times 500\,\mathrm{rad\,s^{-1}}$ for the horizontal direction and $a_{\mathcal{Y}} \coloneqq 2\pi \times 700\,\mathrm{rad\,s^{-1}}$ for the vertical direction [47, Tab. 2.6]. Combining the actuator models with the ORM yields the electron beam dynamics:

$$y(s) = Rg(s)u(s) + d(s), \tag{1.3}$$

where $u : \mathbb{C} \mapsto \mathbb{C}^{n_u}$ are the magnet setpoints (inputs), $y : \mathbb{C} \mapsto \mathbb{C}^{n_y}$ the BPM position measurements (outputs) and $d : \mathbb{C} \mapsto \mathbb{C}^{n_y}$ the (unknown) disturbances. In the form (1.3), the ORM is measured in µm/mA. The aim of the FOFB is to minimise the effect of the disturbances $d(s)$ onto the beam position $y(s)$ by computing inputs $u(s)$, while considering slew-rate and amplitude constraints of the corrector magnets. The disturbances have significant frequency components up to 140 Hz [9] and a non-zero steady-state value, which requires an integrating controller.

System (1.3) is referred to as a (single-array) *cross-directional* (CD) system [153]. For CD systems, the controller synthesis can be split into two parts: First, the design of a static compensator that considers R, and second, a scalar feedback controller $c : \mathbb{C} \mapsto \mathbb{C}$ that is determined from $g(s)$. Other examples of CD systems are found in strip metal rolling [39], paper making [63], plastic film extrusion [70] and coating [22]. Such applications often have many inputs and outputs with possibly varying plant configurations for which optimisation-based synthesis, such as $\mathcal{H}_\infty$ or $\mathcal{H}_2$ optimal control, can be impractical. Moreover, if the control system is operated at a high sampling frequency, controllers that involve real-time optimisation are difficult to realise in practice [50].

The controller design for CD systems can be greatly simplified by substituting the thin *singular value decomposition* (SVD) $R = U\Sigma V^{\mathrm{T}}$ in (1.3), yielding a system of the form

$$y(s) = U \begin{bmatrix} \sigma_1 g(s) & & \\ & \ddots & \\ & & \sigma_{n_y} g(s) \end{bmatrix} V^{\mathrm{T}} u(s) + d(s), \tag{1.4}$$

which can be diagonalised after left-multiplication with U^{T} and redefining variables. This approach, which is referred to as *modal decomposition* [61] and used by several synchrotrons [121], decouples the multi-input multi-output (MIMO) system (1.3) into a set of independent SISO systems and therefore allows SISO techniques to be used for the controller design.

For most synchrotrons, the condition number $\kappa(R) := \|R\|_2 \|R^{-1}\|_2$ is large, due to (spatial) oversampling or non-ideal placements of BPMs and correctors in synchrotrons [16], [117], [149]. In such cases, disturbances aligned with the left

singular vectors of R that are associated with small singular values will produce large control inputs. In other words, the gain margins arising from inverting small-magnitude singular values are small, which makes the resulting control system prone to modelling uncertainties that can cause instabilities and actuator saturation [129]. To account for the large condition number, a common approach is to substitute a regularised inverse for the standard (pseudo-)inverse of Σ [55, Ch. 6.1.4], which results in effective damping of the controller gains for modes associated with small-magnitude singular values [47].

1.3 Diamond-II Upgrade

Diamond Light Source is a third-generation light source that was inaugurated in 2007. In 2016, the MAX-IV synchrotron facility in Lund, Sweden, initiated the era of fourth-generation light sources. By reducing the emittance using a new lattice design, the MAX-IV synchrotron increased the brightness by one to two orders of magnitude compared to third-generation light sources, which represented a step-change for beamline technologies and hence for end-users [127].

Like several other synchrotrons, Diamond proposed to upgrade its facility to a fourth-generation light source and published a conceptual design report in 2019 [4]. This was followed in 2022 by the publication of a technical design report [2], and Diamond received a £81.5M funding confirmation for the first phase of Diamond-II [58]. The upgrade will increase the electron beam energy from 3 GeV to 3.5 GeV and reduce the (horizontal) emittance by a factor of 20, resulting in an increase of brightness by one to two orders of magnitude.

In order to meet the increased performance requirements of Diamond-II, most machine components must be redesigned, including the FOFB and interfacing machine components [2, Ch. 2.11.7]. Due to the lower target emittance and smaller beam sizes, the increase in performance requirements for the FOFB is substantial and given in the top half of Table 1.2. In contrast to the 10 % at Diamond, the RMS deviation must be maintained within 3 % of the beam size at Diamond-II, which amounts to 0.97 µm in the horizontal and 0.18 µm in the vertical direction

Table 1.2: Comparison of FOFB specifications and parameters for Diamond [66] and Diamond-II [2, Ch. 2.6.1]. The absolute RMS values refer to BPMs on standard straights.

	Closed-loop bandwidth (Hz)	Max. relative RMS (%)	Max. horiz. RMS (µm)	Max. vertical RMS (µm)
Diamond	140	10	12.3	0.6
Diamond-II	1000	3	0.97	0.18
	Sample/actuation frequency (kHz)	Time delay (µs)	Number of BPMs (-)	Number of correctors (-)
Diamond	10	700	172	168
Diamond-II	100	100	252	396

on mid straights [2, Tab. 2.6.1]. Because new beamlines are able to resolve beam motion above 100 Hz, the closed-loop bandwidth is raised to 1 kHz.

In order to meet these new requirements, the FOFB capabilities are raised as shown in the bottom half of Table 1.2. The number of sensors (BPMs) is increased from 172 to 252 and the number of correctors from 168 to 396. At the same time, the time delay is reduced from 700 µs to 100 µs and the sample frequency increased from 10 kHz to 100 kHz, which requires to upgrade the BPMs [2, Ch. 2.6.2].

To achieve the required closed-loop bandwidth, new corrector magnets and power supplies will be used for Diamond-II. At Diamond, all correctors are identical and produce a (medium-strength) magnetic field from 0 Hz to 500 Hz or 700 Hz, which – considering the time delay – suffices for achieving the 140 Hz closed-loop bandwidth and a zero steady-state error under persisting disturbances. At Diamond-II, the correctors will need to respond to frequencies up to 8 kHz to achieve the 1 kHz closed-loop bandwidth, but designing magnets and magnet power supplies with a bandwidth ranging from 0 kHz to 8 kHz is technically difficult [69]. First, eddy currents in the steel walls of the vacuum chamber limit the bandwidth of the effective magnetic field, which requires using different wall materials for low- and high-bandwidth correction (and consequently different actuator models). Second, high-bandwidth power supplies (amplifiers) typically use alternating current as opposed to direct current required for operating at low frequencies. Finally, the computing infrastructure for the magnet setpoints needs to support a large range

of currents, from nA to A, while guaranteeing a sufficiently fine resolution. For these reasons, Diamond-II will use two separate types of correctors: slow correctors with strong magnetic fields for low-bandwidth correction and fast correctors with weak magnetic fields for high-bandwidth correction.

The Diamond-II upgrade has a number of consequences for the FOFB infrastructure, and, in particular, the control algorithm. Introducing two types of corrector magnets changes the single-array system (1.3) to a *two-array* or *dual-rate* system:

$$y(s) = R_{\mathrm{s}} g_{\mathrm{s}}(s) u_{\mathrm{s}}(s) + R_{\mathrm{f}} g_{\mathrm{f}}(s) u_{\mathrm{f}}(s) + d(s), \tag{1.5}$$

where $u_{\mathrm{s}} : \mathbb{C} \mapsto \mathbb{C}^{n_{\mathrm{s}}}$ and $u_{\mathrm{f}} : \mathbb{C} \mapsto \mathbb{C}^{n_{\mathrm{f}}}$ are the inputs for the n_{s} slow and n_{f} fast correctors, $R_{\mathrm{s}} \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\mathrm{s}}}$ and $R_{\mathrm{f}} \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\mathrm{f}}}$ the corresponding ORMs, and $g_{\mathrm{s}} : \mathbb{C} \mapsto \mathbb{C}$ and $g_{\mathrm{f}} : \mathbb{C} \mapsto \mathbb{C}$ the low- hand high-bandwidth actuator models. As for the single-array case, R_{s} and R_{f} are ill-conditioned. The second actuator array in (1.5) precludes the application of the SVD-based modal decomposition used in the single-array case (see Chapter 4). Although extensions of the modal decomposition have been proposed [40], [41], [51], these methods introduce restrictive assumptions on R_{s} and R_{f} or leave the decoupling process unspecified when $\mathrm{range}(R_{\mathrm{s}}) \supseteq \mathrm{range}(R_{\mathrm{f}})$, and cannot be applied at Diamond-II.

Other approaches exist that exploit the bandwidth difference of the two actuator arrays, but when the slow and fast correctors are driven by two separate feedback loops that share a common frequency range it has been shown that the two systems interact and eventually cause instabilities [69]. One approach for two-array systems therefore introduces a frequency deadband between the slow and fast feedback systems [131], although this leaves a portion of the disturbance spectrum uncorrected. Other approaches cascade the loop and introduce a reference signal for the fast actuators that considers the contribution from the slow actuators, but these approaches have been shown to be prone to instabilities [69].

1.4 Aims and Contributions

The aim of this thesis is the design of electron beam stabilisation controllers for the two-array CD system (1.5) of Diamond-II, for which two control algorithms of differing complexity are proposed. Theoretical and simulation results are validated using tests on the existing storage ring and show that the controllers meet the existing Diamond specifications in terms of disturbance attenuation and computing speed. The proposed design techniques are used to extrapolate the results to the Diamond-II case, which provide a basis for the Diamond-II FOFB design and hardware specifications [2, Ch. 2.11.7].

The first control algorithm is a controller that uses the *generalised singular value decomposition* (GSVD) [55, Ch. 6.1.6] to decouple the two-array system (1.5). Since the GSVD is an extension of the SVD to two matrices, the novel GSVD-based decoupling can be interpreted as an extension of modal decomposition. Analogous to the single-array case, the proposed approach compensates for the ill-conditioned ORMs using static gain matrices. In contrast to existing beam stabilisation controllers for two-array systems [69], [115], [131], [166], the GSVD-based approach is a parameter-free method and does not require the two arrays to be treated as separate control problems. For the controller dynamics, a mid-ranging approach is proposed [6], but any other controller could be used instead, such as PID, $\mathcal{H}_2$ or $\mathcal{H}_\infty$ control. This control approach has been adopted for Diamond-II and included in the Diamond-II technical design report [2, Ch. 2.11.7].

In the early design phase of Diamond-II, a three-array actuator model was considered that included two types of fast corrector magnets. With the aim of extending the GSVD-based decomposition to a three-array system, Chapter 5 of this thesis focuses on the *higher-order generalised singular value decomposition* (HO-GSVD) [119], which is an extension of the GSVD to three or more matrices. Since the original HO-GSVD framework is restricted to matrices with full column rank, this contribution extends the HO-GSVD to the rank-deficient case, enabling the application of a HO-GSVD based modal decomposition to multi-array systems.

The second control algorithm considered in this thesis is model predictive control (MPC) [46], which is an algorithm that produces control inputs by repeatedly solving a constrained quadratic program (CQP). MPC is versatile in the sense that it can cope with an arbitrary number actuator arrays and constraints, but it requires solving a CQP in real-time and is therefore difficult to implement at 10 kHz for the large number of inputs and outputs [72]. After comparison with the alternating direction of multipliers method (ADMM), the CQP is solved using the fast gradient method [110, Ch. 6.1.3] for a horizon consisting of a single time-step. For larger horizons, the fast gradient method is combined with Dykstra's algorithm [21]. The ill-conditioned plant is accommodated by tuning the MPC to mimic the existing controller and providing systematic tuning procedures. The MPC algorithm is tested on the existing storage ring, which represents a first-of-its-kind application of MPC to the electron beam stabilisation problem.

In addition to the above, the thesis investigates symmetries in the arrangement of storage ring components that produce ORMs with structural symmetries [164, Ch. 10.2.4, p. 329], which can be used to (block-)diagonalise multi-array CD systems. In contrast to previous publications that focus on the controller analysis [48], [106], [167], this thesis focuses on computational advantages and shows how combining several symmetries can sparsify the gain matrices and significantly increase the computation speed of the controller. At a sampling frequency of 100 kHz, the time delay associated with computing the control inputs makes up for 20 % of the total latency [2, Table 2.11.10] and has a significant impact on the closed-loop bandwidth.

In pactice, most systems do not have exact structural symmetries [28], and Chapter 3 is devoted to CD systems with approximate structural symmetries. After fixing the controller structure, this thesis proposes an optimisation-based approach to find an approximation of the plant that has the structural symmetries and optimises a robust performance and stability criterion, thereby exploiting the sparsity of structural decompositions.

Apart from the studies on structural symmetries that are evaluated using hardware-in-the-loop simulations, the results and outcomes of this thesis have

been successfully implemented and tested on the existing Diamond Light Source storage ring under various operating conditions, including moving ID gaps and Wiggler ramping. For these tests, a new centralised computing device – a Vadatech AMC540 [152] with Xilinx Virtex-7 FPGA and two digital signal processors (DSPs) – has been integrated in the existing storage ring. A real-time control system has been implemented and both control algorithms parallelised on the DSPs in C language. For the Diamond-II upgrade, it is planned to implement the GSVD-based controller on a different computing device, but the present implementation is re-used for additional studies on the existing Diamond storage ring.

In summary, the contributions of this thesis and related publications are:

- Control of two-array CD systems using the GSVD [88] and results from the Diamond storage ring[1].
- Design of a GSVD-based controller for Diamond-II [2]; [86][1].
- The higher-order GSVD for rank-deficient matrices [85].
- MPC using the fast gradient method and Dykstra's algorithm [84]; [89].
- MPC implementation and results from the Diamond storage ring[1].
- Structural symmetries of the Diamond-II ORM for computational efficiency [90].
- Control of CD systems with approximate structural symmetries[1].
- ADMM for block-circulant MPC [83].

1.5 Thesis Outline

The following paragraphs summarise the thesis chapters, each of which starts and ends with separate introductions and conclusions. The last section of this chapter, Section 1.6, summarises existing results on single-array controllers for electron beam stabilisation.

Chapter 2 By analysing the periodicity and the reflection properties of the Diamond-II betatron function, it is shown that the Diamond-II ORM inherits a

[1]Pre-print/draft.

block-centrosymmetric and a block-circulant structure. Using hardware-in-the-loop simulations on the AMC540, it is shown that the structural symmetries can be used to obtain different computationally efficient decompositions of the controller.

[90] I. Kempf, P. J. Goulart, S. R. Duncan, *et al.*, "Symmetry exploitation in orbit feedback systems of synchrotrons for computational efficiency," *IEEE Trans. Nucl. Sci.*, vol. 68, no. 3, pp. 258–269, Mar. 2021

Chapter 3 The structural symmetries of a CD system with approximate symmetries are recovered by substituting a structural approximation for the original plant model, thereby introducing an approximation error. The nominal stability, performance and robust stability of the closed loop are analysed and it is proposed to obtain robust structural approximations from semidefinite programming problems.

[87] I. Kempf, P. Goulart, and S. Duncan, *Control of cross-directional systems with approximate symmetries*, Jun. 2023. arXiv: 2306.17565 [eess.SY]

Chapter 4 Based on the GSVD, the generalised modal decomposition for two-array CD systems is defined and a two-array controller developed, which uses static gain matrices to compensate for ill-conditioned ORMs. The controller is tested on the existing Diamond storage ring and applied to preliminary Diamond-II data.

[88] I. Kempf, S. R. Duncan, P. J. Goulart, *et al.*, "Multi-array electron beam stabilization using block-circulant transformation and generalized singular value decomposition," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Jeju Island, Republic of Korea, Dec. 2020, pp. 3431–3436

[2] M. G. Abbott *et al.*, "Diamond-II technical design report," Diamond Light Source, Didcot, UK, Tech. Rep., Aug. 2022. https://www.diamond.ac.uk/Home/News/LatestNews/2022/14-10-22.html

[86] I. Kempf, *Diamond-II fast orbit feedback: Controller design report*, Apr. 2023. https://github.com/kmpape/DII_controller_design

Chapter 5 The original HO-GSVD framework is modified to accommodate rank-deficient matrices and the notion of common HO-GSVD subspaces is extended to isolated subspaces. The HO-GSVD is demonstrated on an image classification dataset.

[85] I. Kempf, P. J. Goulart, and S. R. Duncan, "A higher-order generalized singular value decomposition for rank deficient matrices," *SIAM J. Matrix Anal. Appl.*, 2023, to appear

Chapter 6 An MPC approach is formulated for the electron beam stabilisation problem and the corresponding observer and regulator are tuned to match the performance of the existing single-array CD controller. The fast gradient method is compared with ADMM and then implemented on the AMC540. The convergence of the MPC implementation is analysed and the effect of actuator saturation

investigated. MPC is tested on the existing Diamond storage ring for different tuning parameters and the results are compared with a single-array CD controller.

[83] I. Kempf, P. J. Goulart, and S. R. Duncan, "Alternating direction method of multipliers for block circulant model predictive control," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Nice, France, Dec. 2019, pp. 4311–4316
[84] I. Kempf, P. J. Goulart, and S. R. Duncan, "Fast gradient method for model predictive control with input rate and amplitude constraints," in *Proc. IFAC World Congr.*, Berlin, Germany, Jul. 2020, pp. 6542–6547
[89] I. Kempf, P. J. Goulart, S. R. Duncan, *et al.*, "Model predictive control for electron beam stabilization in a synchrotron," in *Proc. Eur. Contr. Conf. (ECC)*, London, UK, Jul. 2022, pp. 814–819

Chapter 7 A real-time control system system is implemented on the AMC540, which is integrated into the existing FOFB infrastructure of the Diamond storage ring. The control algorithms are parallelised in C language on the TI C6678 DSP [145] and the implementation is optimised to meet the 10 kHz target frequency.

[78] I. Kempf, Jul. 2021. `https://github.com/kmpape/fofb-amc540`
[79] I. Kempf, Jul. 2021. `https://github.com/kmpape/fofb-amc540-codegen`
[80] I. Kempf, Nov. 2022. `https://github.com/kmpape/fofb-amc540-startup`
[81] I. Kempf, Dec. 2022. `https://github.com/kmpape/fofb-amc540-hil`

1.6 Technical Background

1.6.1 Modal Decomposition

The modal transformation, or modal decomposition, [61] decouples the single-array CD system (1.3) using the thin SVD of R:

$$R = U\Sigma V^{\mathrm{T}}, \qquad U \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\mathrm{y}}}, \qquad V \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\mathrm{u}}}, \qquad \Sigma = \operatorname{diag}\left(\sigma_1, \ldots, \sigma_{n_{\mathrm{y}}}\right) \succ 0, \tag{1.6}$$

where $U^{\mathrm{T}}U = I$, $V^{\mathrm{T}}V = I$ and $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_n > 0$ assuming that

$$\operatorname{rank}(R) = n_y \leq n_u. \tag{1.7}$$

If $\operatorname{rank}(R) < n_y$ then the system is uncontrollable, which is not considered further. Substituting (1.6) in (1.3) and defining the modal variables as

$$\hat{y}(s) \coloneqq U^{\mathrm{T}} y(s), \qquad \hat{u}(s) \coloneqq V^{\mathrm{T}} u(s), \qquad \hat{d}(s) \coloneqq U^{\mathrm{T}} d(s), \tag{1.8}$$

yields the modal representation of (1.3) [62] as

$$\hat{y}(s) = \Sigma g(s) \hat{u}(s) + \hat{d}(s). \tag{1.9}$$

In modal space, the dynamics are given by a set of uncoupled single-input single-output (SISO) systems. Because the matrices U and V of the modal transformation (1.8) are orthonormal, i.e. $U^{\mathrm{T}}U = I$ and $V^{\mathrm{T}}V = I$, it holds that [55, Ch. 2.3.5]

$$\|\hat{y}(s)\|_2 = \|y(s)\|_2, \qquad \|\hat{u}(s)\|_2 = \|u(s)\|_2, \tag{1.10}$$

and likewise in the time-domain, so that stability properties and 2-norm based upper bounds on performance and robustness measures of the control loop are retained when transforming the modal system back to the original space [153].

1.6.2 Internal Model Control

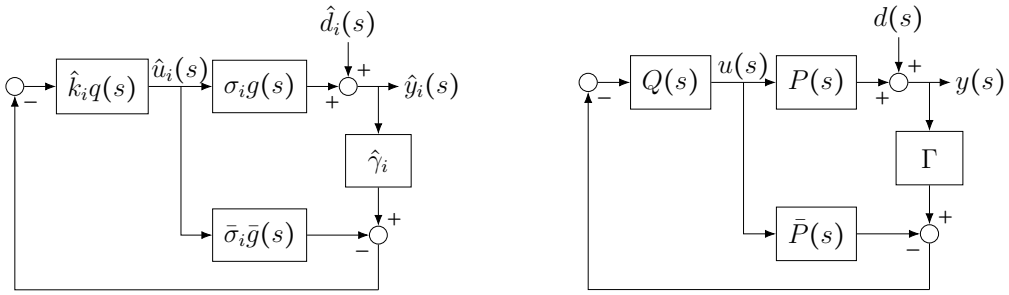
In modal space, the MIMO system reduces to (1.10), which reads component-wise as

$$\hat{y}_i(s) = \sigma_i g(s)\hat{u}_i(s) + \hat{d}_i(s), \tag{1.11}$$

for $i = 1, \ldots, n_y$ and where $\hat{y}_i(s)$ denotes component i of $\hat{y}(s)$. Based on the decoupled system (1.11), a SISO controller can be designed for each mode separately. At most synchrotrons, the mode-by-mode controller is determined using a *proportional-integral-derivative* (PID) controller and the standard feedback structure. Diamond instead uses IMC, which is particularly suitable for systems with large time delays [45]. The IMC structure used at Diamond is shown in Fig. 1.3a in modal space, where $\sigma_i g(s)$ and $\bar{\sigma}_i \bar{g}(s)$ are the plant and the plant model, $\hat{k}_i q(s)$ the mode-by-mode IMC filter and $\hat{\gamma}_i$ a *regularisation* gain, which is also referred to as output compensator in the following. If not otherwise noted, it is assumed that the plant model is accurate, i.e. $\bar{g}(s) = g(s)$ and $\bar{R} = R$ so that $\bar{\sigma}_i = \sigma_i \ \forall i = 1, \ldots, n_y$.

By fixing the IMC filter structure as $\hat{k}_i q(s)$, the dynamic part $q(s)$ is chosen to be identical for each mode, which yields a computationally advantageous controller (in original space) of the form "matrix × scalar transfer function". The static part of the IMC filter, $\hat{k}_i$, is set to $\hat{k}_i := 1/\sigma_i$, and the dynamic part is determined by

$$q(s) = \frac{T_{\mathrm{m}}(s)}{g(s)}, \tag{1.12}$$



(a) Modal space (SISO). **(b)** Original space (MIMO).

Figure 1.3: IMC structure in modal space and original space with $\sigma_i g(s)$ ($P(s) = Rg(s)$) and $\bar{\sigma}_i \bar{g}(s)$ ($\bar{P}(s) = \bar{R}\bar{g}(s)$) denoting the plant and the plant model, $\hat{\gamma}_i$ (Γ) the regularisation gain and $\hat{k}_i q(s)$ ($Q(s) = Kq(s)$) the controller.

where $T_\mathrm{m} : \mathbb{C} \mapsto \mathbb{C}$ is the *complementary sensitivity* that includes the non-minimum phase part of $g(s)$ and is chosen as

$$T_\mathrm{m}(s) \coloneqq \frac{\lambda}{s+\lambda} \mathrm{e}^{-\tau_d s}. \tag{1.13}$$

Without regularisation ($\hat{\gamma}_i = 1$) and model uncertainty ($\bar{\sigma}_i \bar{g}(s) = \sigma_i g(s)$), the control law from Fig. 1.3a is obtained as

$$\hat{u}_i(s) = -\hat{k}_i q(s)(\hat{\gamma}_i y(s) - \sigma_i g(s)\hat{u}_i(s))) = -\hat{k}_i q(s)\hat{d}_i(s), \tag{1.14}$$

and substituting in (1.11), this yields the closed loop transfer function of each mode as

$$\hat{y}_i(s) = (1 - T_\mathrm{m}(s))\hat{d}_i(s) =: S_\mathrm{m}(s)\hat{d}_i(s), \tag{1.15}$$

where

$$S_\mathrm{m}(s) = 1 - \frac{\lambda}{s+\lambda} \mathrm{e}^{-\tau_d s} \tag{1.16}$$

is the output sensitivity. To avoid large output sensitivity overshoots, the bandwidth of $T_\mathrm{m}(s)$ is chosen as $\lambda = 1/\tau_d$ [107, Ch. 4.6]. At Diamond the time delay depends on the storage ring configuration and is either 700 µs or 900 µs (Section 7.1.3), which results in bandwidth choices of 227 Hz or 177 Hz. Note that as long as $\hat{k}_i q(0)\bar{\sigma}_i \bar{g}(0) = 1$, the IMC structure implements an integrator guaranteeing that

$S_{\mathrm{m}}(0) = 0$. In the process control literature, controllers of the form (1.12)-(1.13) are referred to as Dahlin or *lambda* controllers [107, Ch. 4.5].

After designing the mode-by-mode controllers in mode space, the IMC filters are diagonally concatenated and the modal transformation (1.8) inverted to obtain the MIMO filter $Q : \mathbb{C} \mapsto \mathbb{C}^{n_u \times n_y}$ as

$$Q(s) \coloneqq V \operatorname{diag}(\hat{k}_1 q(s), \ldots, \hat{k}_{n_y} q(s)) U^{\mathrm{T}} q(s) = V \Sigma^{-1} U^{\mathrm{T}} q(s), \tag{1.17}$$

which is embedded in the structure of Fig. 1.3b, where the plant and the plant model are defined as $P(s) \coloneqq Rg(s)$ and $\bar{P}(s) \coloneqq \bar{R}\bar{g}(s)$. Because the *thin* SVD is used in (1.6), the matrix $V\Sigma^{-1}U^{\mathrm{T}} =: R^{\dagger}$ corresponds to the Moore-Penrose pseudoinverse [55, P5.5.2] of R.

The controller design in discrete-time follows the same procedure, but with $g(s)$ replaced by (6.2) and $T_{\mathrm{m}}(s)$ by

$$T_{\mathrm{m}}(z) = \frac{1 - b_\lambda}{z - b_\lambda} z^{-n_d}, \tag{1.18}$$

where $b_\lambda = \exp(-\lambda T_s)$ is the discrete-time closed-loop pole [66].

1.6.3 Regularisation

The control law (1.14) shows that the mode-by-mode controller gains are proportional to $1/\sigma_i$. For Diamond, the condition numbers are $\kappa(R_{\mathcal{X}}) = 12,281$ and $\kappa(R_{\mathcal{Y}}) = 9837$, so that disturbances aligned with left singular vectors of R associated with small σ_i can cause large actuator inputs. In order to limit the transient magnitude of these inputs, one approach is to view (1.3) as a static problem in discrete-time [49] and to find the input u_k for each $k \in \mathcal{Z}$ from

$$\underset{u_k \in \mathbb{R}^{n_u}}{\text{minimise}} \quad \|y_k + Ru_k\|_2^2 + \mu \|u_k\|_2^2, \tag{1.19}$$

where $\mu \in \mathbb{R}_+$ is a *regularisation parameter* and chosen as $\mu = 1$ at Diamond [54]. Minimising (1.19) with respect to u_k yields the solution

$$u_k = -\left(R^{\mathrm{T}} R + \mu I\right)^{-1} R^{\mathrm{T}} y_k, \tag{1.20}$$

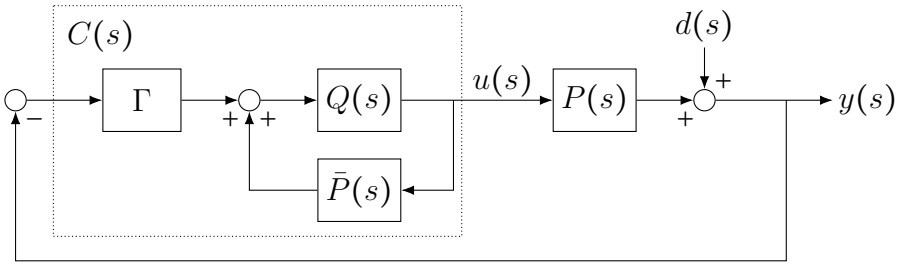


Figure 1.4: Standard feedback structure in original space (MIMO) obtained from rearranging Fig. 1.3b.

suggesting the replacement of the pseudo-inverse on the right-hand side of (1.17) with the regularised pseudo-inverse, which can be achieved by choosing the output compensator Γ as

$$\Gamma \coloneqq \left(RR^{\mathrm{T}} + \mu I\right)^{-1} RR^{\mathrm{T}} = U\left(\Sigma^2 + \mu I\right)^{-1}\Sigma^2 U^{\mathrm{T}}, \tag{1.21}$$

from which the mode-by-mode regularisation gains are obtained as $\hat{\gamma}_i \coloneqq \sigma_i^2/(\sigma_i^2 + \mu)$. According to Fig. 1.3a, the (continuous time) transfer function from $\hat{y}_i(s)$ to $\hat{u}_i(s)$ becomes

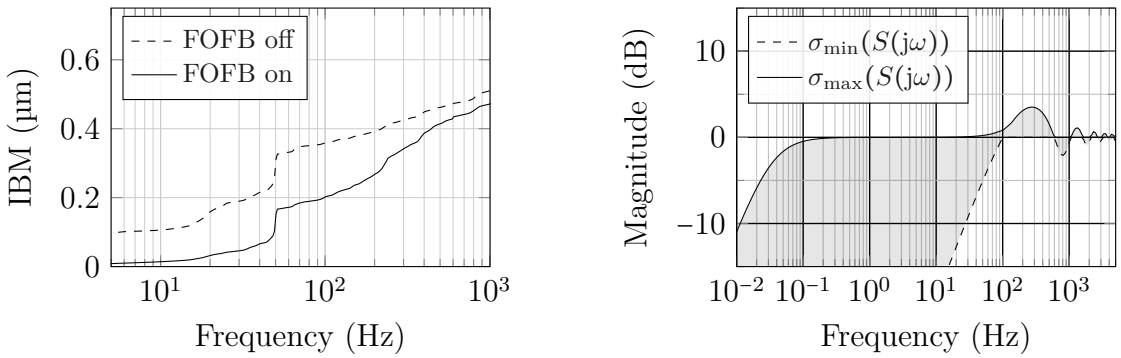
$$\hat{u}_i(s) = -\hat{\gamma}_i \frac{\hat{k}_i q(s)}{1 - \hat{k}_i q(s)\sigma_i g(s)}\hat{y}_i(s) = -\frac{\sigma_i}{\sigma_i^2 + \mu}\frac{q(s)}{1 - T_{\mathrm{s}}(s)}\hat{y}_i(s), \tag{1.22}$$

which describes the mode-by-mode controller in standard feedback form and shows how the output compensator modifies the open-loop gain as a function of the singular values. For $\sigma_i^2 \ll \mu$, $\sigma_i/(\sigma_i^2 + \mu) \approx 0$, so that control action associated with small σ_i is effectively damped, whereas for $\sigma_i^2 \gg \mu$, $\sigma_i/(\sigma_i^2 + \mu) \approx 1/\sigma_i$, i.e. the controller gain is left unchanged.

1.6.4 Standard Feedback Structure

Even though the controller at Diamond is *designed* using the IMC structure, it is implemented using the standard feedback structure from Fig. 1.4, which does not require explicit computing of the model path output from Fig. 1.3b. The standard controller $C : \mathbb{C} \mapsto \mathbb{C}^{n_u \times n_y}$ is obtained as

$$C(s) \coloneqq \left(I - Q(s)\bar{P}(s)\right)^{-1} Q(s)\Gamma = Kc(s), \tag{1.23}$$



(a) IBM on BPM 1.

(b) Overall sensitivity.

Figure 1.5: Measured IBM on BPM 1 and theoretical sensitivity gains for the existing controller (vertical direction, $\tau_d = 900\,\mu\mathrm{s}$).

where the static controller gain $K \in \mathbb{R}^{n_u \times n_y}$ is given by

$$K := V \operatorname{diag}\left(\frac{\sigma_1}{1+\sigma_1^2}, \ldots, \frac{\sigma_{n_y}}{1+\sigma_{n_y}^2}\right) U^{\mathrm{T}} = \left(R^{\mathrm{T}} R + \mu I\right)^{-1} R^{\mathrm{T}}, \tag{1.24}$$

and the scalar dynamics $c : \mathbb{C} \mapsto \mathbb{C}$ by

$$c(s) := \frac{\lambda}{a} \frac{s+a}{s+\lambda(1-\mathrm{e}^{-s\tau_d})}. \tag{1.25}$$

Note that (1.23) shows that the IMC structure implements integrating behaviour as long as $Q(0)\bar{P}(0) = I$, even for $\bar{P}(s) \neq P(s)$.

The performance of the controller is usually measured using the integrated beam motion (IBM), which is defined as the square root of $\sum_{f=0}^{F} \frac{2}{F^2} |y_i(f)|^2$, where $y_i(f)$ is the discrete Fourier transform of BPM signal i and F the number of Fourier samples used [47, Ch. 3.1]. The IBM measures the maximum displacement that can occur up to a certain frequency when the corresponding Fourier terms of the disturbance are in phase. Fig. 1.5 shows the IBM measured at BPM 1 and the minimum and maximum singular values of the corresponding output sensitivity $S(\mathrm{j}\omega) = I - P(\mathrm{j}\omega)Q(\mathrm{j}\omega)$ in Fig. 1.5b. Fig. 1.5a also shows the IBM caused by the disturbance, which is measured when the FOFB is disabled.

2
Structural Symmetries

In most synchrotrons the magnetic lattices, the BPMs and the corrector magnets are placed in repeated patterns around the storage ring [164, Ch. 10.2.4, p. 329]. These repeated sections are usually referred to as *superperiods* or *cells*, and their pattern invokes a *circulant* symmetry. Often an additional symmetry is introduced by mirror-reflecting the lattice at the middle of one cell [164, Ch. 10.2.4, p. 331], which generates a *centrosymmetry*. The circulant pattern considerably simplifies the design of the synchrotron, while the mirror-reflection cancels out non-linear effects introduced by quadrupole and sextupole magnets. Although this symmetry is intentionally created in the design phase of the synchrotron, it is generally not considered during the synthesis of the orbit feedback system.

Most orbit feedback systems use modal decomposition to map the dynamics (1.3) to modal space and synthesise the controller on a mode-by-mode basis, which results in the control law (Section 1.6)

$$u(s) = -Kc(s)y(s), \tag{2.1}$$

where K is pre-computed offline and possibly regularised to account for an ill-conditioned orbit response matrix (ORM). The modal decomposition is applicable

This chapter is based on [90] I. Kempf, P. J. Goulart, S. R. Duncan, *et al.*, "Symmetry exploitation in orbit feedback systems of synchrotrons for computational efficiency," *IEEE Trans. Nucl. Sci.*, vol. 68, no. 3, pp. 258–269, Mar. 2021.

to any kind of synchrotron but does not exploit the symmetric structure. Moreover, if actuator limitations such as slew-rate or amplitude constraints are present, then (2.1) must be extended with an anti-windup scheme [50]. Other control algorithms, such as model predictive control (MPC), allow optimal handling of actuator constraints while achieving the same or better trajectory error correction [83]. Such an algorithm, however, uses real-time optimisation so exploiting symmetry would considerably reduce the computational complexity. Other approaches exist that neglect the coupling between different input-output pairs or the small-magnitude singular values and also benefit the computation speed. For increasing controller gains, however, these approaches might lead to instabilities of the closed-loop system [135].

As an alternative to the SVD, several approaches have used a Discrete Fourier Transformation (DFT) to map the dynamics (1.3) to the spatial Fourier domain. In [167], a feedback system based upon a harmonic analysis of the dynamics was implemented that aimed at controlling the dominant harmonics of the beam displacement. This approach only yields speed advantages if the underlying system has an accurate circulant symmetry, in which case the individual harmonics are independent, or if a sufficiently large number of harmonics are omitted. A similar approach was used in [48] to diagonalise the system dynamics (1.3) and concentrate the betatron tune uncertainty in the Fourier coefficients, which considerably simplified the subsequent robustness analysis. These results were extended in [105], [106], which observed the block-circulant property of R. Because the observed symmetry was only approximate, a circulant approximation was proposed and the block-circulant transformation – the matrix equivalent of the DFT – subsequently applied to an approximation of R.

Symmetry is accompanied by redundancies in the mathematical representation of the system [30], and, most importantly, exploiting symmetry has the potential to speed up controller computations and reduce memory requirements [83], which would allow for the use of more complex control algorithms. For the Diamond-II sampling frequency of 100 kHz, the time associated with the controller computation represents 20 % of the total latency and has therefore a significant impact on the

closed-loop bandwidth. In this chapter, the symmetry properties of the betatron function – a property of the magnetic lattices – are mathematically related to the (controller) matrices, by showing that R inherits the symmetry properties of the betatron function.

In practice only a few physical systems have accurate symmetry properties and applying a symmetric decomposition requires an approximation. This chapter therefore also addresses the common case of broken symmetry and extends the results presented in [105]. For each of the block-circulant, centrosymmetric and the combined-symmetry cases, formulae for structured approximations are derived that minimise the Frobenius norm error. It is shown how the asymmetry of the ORM can be concentrated in certain elements of the symmetrical decomposition, which is used in a subsequent analysis in Chapter 3. The decompositions are illustrated using the Diamond-II ORM [4] and the approximation error compared to the ESRF-EBS, MAX IV and ALBA synchrotrons. It is shown that even when the original matrix is only approximately centrosymmetric, the orbit feedback successfully reduces the trajectory error. The analysis is concluded by demonstrating the main advantage of exploiting structural symmetries, which is increased computation-speed. Using a C-language implementation on the hardware used for feedback experiments at Diamond (Chapter 7), the standard approach (2.1) is compared with the different symmetrical decompositions. It is shown that, in exchange for an insignificant performance decrease of the trajectory error correction, the computational speed of the controller can be significantly improved. Accelerating the controller computations allows faster sampling rates to be used or the deployment of more advanced control algorithms, which is demonstrating by combining MPC with the results from this chapter in related work [83].

This chapter is structured as follows. In Section 2.1, block-circulant and centrosymmetric matrices are briefly outlined, with more details included in Appendix 2.B. Section 2.2 presents the results on the symmetric structure of the ORM and its decompositions and Section 2.4 addresses the case of broken symmetry. In Section 2.5, the results are summarised in a case study of the Diamond-II

synchrotron, in which the controller is simulated using structured approximations, the nominal stability is verified and the speed advantages of a controller that exploits the symmetric structures are demonstrated.

2.1 Matrices with Structural Symmetries

In an orbit feedback system, the calculation of optimal set-points for the corrector magnets at each sampling instant requires a matrix-vector multiplication. It will be shown that the matrix inherits certain symmetry properties from the storage ring and that these properties can be used to simplify the computations needed for the orbit feedback system. The analysis of matrices with structural symmetries has its origins in group theory and the linear representations of finite groups [94, Ch. 1], and a characterisation of structural symmetries in terms of permutation matrices is used here [15, Ch. 5]. A square matrix is $\Pi \in \mathbb{R}^{n\times n}$ is a *permutation matrix* if exactly one element in each row and column is equal to 1 and all other elements are zero [68, Ch. 0.9.5]. Among other properties, it follows that $\Pi^{\mathrm{T}}\Pi = I$ and $\Pi^k = I$ for some $k \in \mathbb{Z}_{++}$ [68, Ch. 0.9.5].

Definition 2.1. A set of matrices with a structural symmetry is defined as the subspace $\mathcal{S} := \{A \in \mathbb{R}^{n\times n} \mid A\Pi = \Pi A\} \subseteq \mathbb{R}^{n\times n}$ and associated with a permutation matrix $\Pi \in \mathbb{R}^{n\times n}$.

Because Π is orthonormal, there exists $\mathcal{T} \in \mathbb{C}^{n\times n}$ with $\mathcal{T}^*\mathcal{T} = I$ that diagonalises Π [55, Ch. 2.5]. It follows that $A \in \mathcal{S}$ iff $\mathcal{T}^*A\mathcal{T}$ is diagonal [68, Thm. 1.3.12]. The matrices in $\mathcal{S}$ form a commutative algebra, i.e. $A + B \in \mathcal{S}$, $AB \in \mathcal{S}$ and $AB = BA$ iff $A, B \in \mathcal{S}$. In addition, $A^{-1} \in \mathcal{S}$ if $A \in \mathcal{S}$ is invertible, which is applied to the limit definition of the Moore-Penrose pseudoinverse of A, $A^\dagger := \lim_{\delta\to 0}(A^{\mathrm{T}}A + \delta I)^{-1}A^{\mathrm{T}}$, in Lemma 2.2.

Lemma 2.2. *For $A \in \mathbb{R}^{n_y\times n_u}$, $A^\dagger \in \mathcal{S}$ if $A \in \mathcal{S}$.*

Proof. Suppose that $A \in \mathcal{S}$, so that $(A^{\mathrm{T}}A + \delta I)^{-1} \in \mathcal{S}\ \forall \delta \in \mathbb{R}$. Using the limit definition of $A^{\dagger}$ [55, P5.5.2]:

$$A^{\dagger}\Pi = \lim_{\delta \to 0}\left(A^{\mathrm{T}}A + \delta I\right)^{-1} A^{\mathrm{T}}\Pi \overset{\text{Def. 2.1}}{=} \Pi \lim_{\delta \to 0}\left(A^{\mathrm{T}}A + \delta I\right)^{-1} A^{\mathrm{T}} = \Pi A^{\dagger}.$$

□

Definition 2.3. The *orthogonal complement* $\mathcal{S}^{\perp}$ of $\mathcal{S}$ is defined as

$$\mathcal{S}^{\perp} := \left\{B \in \mathbb{R}^{n \times n} \mid \operatorname{trace}(B^{\mathrm{T}}A) = 0 \quad \forall A \in \mathcal{S}\right\}.$$

From $\operatorname{trace}(B^{\mathrm{T}}A) = \operatorname{trace}((\mathcal{T}^{*}B^{\mathrm{T}}\mathcal{T})(\mathcal{T}^{*}A\mathcal{T}))$, it follows that $B \in \mathcal{S}^{\perp}$ iff $\mathcal{T}^{*}B\mathcal{T}$ is *hollow*, i.e. a matrix with zero diagonal elements. Since $\mathcal{T}^{*}B\mathcal{T}$ and $B \in \mathcal{S}^{\perp}$ are similar, it holds that $\operatorname{trace}(B) = 0$.

The concept of structural symmetry can be straightforwardly extended to *block structural symmetry* by considering matrices $A \in \mathbb{R}^{n_y \times n_u}$ that satisfy $A(\Pi \otimes I_{b_u}) = (\Pi \otimes I_{b_y})A$. The matrices $(\mathcal{T}^{*} \otimes I_{b_y})A(\mathcal{T} \otimes I_{b_u})$ and $(\mathcal{T}^{*} \otimes I_{b_y})B(\mathcal{T} \otimes I_{b_u})$ are then block-diagonal and block-hollow, respectively. In the following, a "scalar" symmetry from Def. 2.1 will not be distinguished from a block symmetry, e.g. writing $R \in \mathcal{S}$ for $R \in \mathbb{R}^{n_y \times n_u}$ may imply that $\exists n, b_y, b_u \in \mathcal{Z}_{++}$ such that $n = n_y/b_y = n_u/b_u$ and $R(\Pi \otimes I_{b_u}) = (\Pi \otimes I_{b_y})R$ with $\Pi \in \mathbb{R}^{n \times n}$ being associated with $\mathcal{S}$.

2.1.1 Block-Circulant Matrices

The set of *circulant* matrices of order n, $\mathcal{C}(n)$, is formed by those matrices that commute with the *cyclic shift matrix* $\Omega_n \in \mathbb{R}^{n \times n}$,

$$\Omega_n := \begin{bmatrix} 0 & I_{n-1} \\ 1 & 0 \end{bmatrix}, \tag{2.2}$$

which satisfies $\Omega_n^n = I$. Circulant matrices are extended to block-circulant matrices in Def. 2.4.

Definition 2.4 (Block-circulant matrices)**.** Let $\mathcal{BC}(n, p, m) \subset \mathbb{R}^{np \times nm}$ denote the set of *block-circulant matrices* of order n that have the form

$$B = \begin{bmatrix} b_0 & b_1 & \dots & b_{n-1} \\ b_{n-1} & b_0 & \dots & b_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ b_1 & b_2 & \dots & b_0 \end{bmatrix}, \quad b_i \in \mathbb{R}^{p \times m}, \tag{2.3}$$

which satisfy [33]

$$B(\Omega_n \otimes I_m) = (\Omega_n \otimes I_p)B. \tag{2.4}$$

The *Fourier matrix* $F_n \in \mathbb{C}^{n\times n}$, defined as

$$F_n := \frac{1}{\sqrt{n}}\begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & w & \dots & w^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & w^{n-1} & \dots & w^{(n-1)(n-1)} \end{bmatrix}, \tag{2.5}$$

with $w = \mathrm{e}^{i\frac{2\pi}{n}}$ and $F_n^* F_n = I_n$, block-diagonalises block-circulant matrices as [33],

$$\hat{B} = (F_n^* \otimes I_p)B(F_n \otimes I_m) = \mathrm{diag}(\nu_0, \dots, \nu_{n-1}), \tag{2.6}$$

where $\nu_j \in \mathbb{C}^{p\times m}$ and $(F_n^* \otimes I_p)(F_n \otimes I_p) = I_{np}$. Equivalently, the block ν_j can also be obtained from

$$\nu_j = \sum_{k=0}^{n-1} b_k \mathrm{e}^{-i\frac{2\pi jk}{n}}. \tag{2.7}$$

The product $F_n x$ yields the coefficients of the discrete Fourier transformation of the vector x. Because the Fourier matrix appears in (2.6), the computation speed of a matrix-vector multiplication Bx can be increased significantly by transforming it to the Fourier domain, i.e. by computing $Bx = (F_n \otimes I_p)\hat{B}(F_n^* \otimes I_m)x$. The computational efficiency arises from the possibility to employ m parallel Fast Fourier Transformations (FFT) for computing products like $(F_n^* \otimes I_m)x$ and the fact that $\hat{B}$ is block-diagonal. For the case that all elements of B are non-zero, the computation time is reduced by[1]

$$\mathcal{O}\big((npm + (p+m)n\log_2 n)/\big(n^2 pm\big)\big), \tag{2.8}$$

which represents the ratio of counts of operations required to perform the product in the Fourier domain versus original domain.

[1]Formula (2.8) neither distinguishes between real- or complex-valued operations nor considers redundancies of complex conjugates in (2.6).

2.1.2 Centrosymmetric Matrices

Definition 2.5. Let $\mathcal{CS}(q,t) \subset \mathbb{R}^{2q\times 2t}$ denote the set of *centrosymmetric matrices* of the form

$$R = \left[\begin{array}{c|c} r_1 & r_2 \\ \hline J_q r_2 J_t & J_q r_1 J_t \end{array}\right], \quad r_i \in \mathbb{R}^{q\times t}, \tag{2.9}$$

where $J_k = \begin{bmatrix} & & 1 \\ & \iddots & \\ 1 & & \end{bmatrix} \in \mathbb{R}^{k\times k}$ and with $RJ_{2t} = J_{2q}R$.

A centrosymmetric ($\mathcal{CS}$) matrix is block-diagonalised by [163]

$$\hat{R} = T_q^{\mathrm{T}} R T_t = \operatorname{diag}(r_1 - r_2 J_t, r_1 + r_2 J_t), \tag{2.10}$$

where the centrosymmetric transformation is defined as

$$T_k := \frac{1}{\sqrt{2}} \begin{bmatrix} I_k & I_k \\ -J_k & J_k \end{bmatrix} \in \mathbb{R}^{2k\times 2k}, \tag{2.11}$$

with $T_k^{\mathrm{T}} T_k = I_{2k}$. As in the case of $\mathcal{BC}$ matrices, the computation speed of a matrix-vector multiplication Rx can be increased significantly by transforming it to the centrosymmetric domain, and it can be shown that (2.8) holds for $n = 2$, $p = q$ and $m = t$ (see Appendix 2.B).

Definition 2.6. Let $\mathcal{SCS}(q,t) \subset \mathbb{R}^{2q\times 2t}$ denote the set of *skew-centrosymmetric matrices* of the form

$$D = \left[\begin{array}{c|c} d_1 & d_2 \\ \hline -J_q d_2 J_t & -J_q d_1 J_t \end{array}\right], \quad d_i \in \mathbb{R}^{q\times t}, \tag{2.12}$$

with $DJ_{2t} = -J_{2q}D$.

In contrast to $\mathcal{CS}$ matrices, a skew-centrosymmetric ($\mathcal{SCS}$) matrix is block anti-diagonalisable by (2.10), i.e.

$$\hat{D} = T_q^{\mathrm{T}} D T_t = \begin{bmatrix} 0 & d_1 - d_2 J_t \\ d_1 + d_2 J_t & 0 \end{bmatrix}. \tag{2.13}$$

2.2 Properties of the Orbit Response Matrix

The element on row m and column n of R, $R_{(m,n)}$, is characterised by the *betatron function* $\beta : \mathbb{R} \mapsto \mathbb{R}_+$ and given by [167, eq. (2)]

$$R_{(m,n)} \coloneqq \frac{\sqrt{\beta_m^B \beta_n^C}}{2\sin(\pi Q)} \cos\big(\pi Q - \big|\phi_m^B - \phi_n^C\big|\big), \tag{2.14}$$

where $\beta_k^{(\cdot)} \coloneqq \beta(\ell_k^{(\cdot)}), \phi_k^{(\cdot)} \coloneqq \phi(\ell_k^{(\cdot)})$ with $\ell \in [0, L]$ representing the distance around the storage ring starting from an arbitrary reference point ($\mathcal{Z}$-axis in Fig. 1.2), L the circumference of the orbit and B and C refer to BPMs and corrector magnets, respectively. The betatron function is obtained from the solutions of non-linear differential equations and is a property of the magnetic lattice [165, Ch. 3.4]. The *phase advance* $\phi : \mathbb{R} \mapsto \mathbb{R}_+$ is defined as

$$\phi(\ell) \coloneqq \int_0^\ell \beta^{-1}(z) dz, \tag{2.15}$$

and for a stable electron beam, the *betatron tune* $Q \coloneqq \phi(L)/2\pi$ is always a non-integer number [103].

As a prerequisite for the analyis of this chapter, it is assumed that the storage ring is divided into S sections of equal length L/S and that each section contains N_B BPMs and N_C correctors, such as stated in Assumption 2.7.

Assumption 2.7 (Prerequisites for structural symmetry)**.** The storage ring is divided into S sections of length L/S with each section containing N_B BPMs and N_C correctors, which results in an ORM $R \in \mathbb{R}^{SN_B \times SN_C}$, $S, N_B, N_C \in \mathbb{Z}_{++}$.

For demonstrating the $\mathcal{BC}$ and $\mathcal{CS}$ properties of the ORM, it will be assumed that the β-function is periodic with period L/S and centrosymmetric with respect to $L/2$, i.e. β can be mirror-reflected about the middle of the storage ring. For convenience, it will be assumed that S, the number of corrector magnets per section and the number of monitors per section, are all even, as it is the case for Diamond-II. The following decompositions can also be applied to storage rings that have an odd number of cells and/or monitors and/or corrector magnets, such as the ESRF-EBS [115]. An odd S results in a different structure of the decomposed $\mathcal{BC}$

matrix (see Section 2.B), while an odd number of monitors or magnets results in a different structure of the decomposed $\mathcal{CS}$ matrix [163]. The numerical results in Section 2.5 would not differ substantially in the case of odd S.

Theorem 2.8 (Block-circulant R)**.** *Suppose that Assumption 2.7 holds and that* $\beta(\ell) = \beta(\ell + L/S)$*. Suppose that the BPMs and corrector magnets of section* $k = 1$ *are placed at ring locations* $\ell_1^B, \ldots, \ell_{N_B}^B$ *and* $\ell_1^C, \ldots, \ell_{N_C}^C$*, and that this arrangement is repeated for sections* $k = 2, 3, \ldots, S$ *as* $\ell_1^B + kL/S, \ldots, \ell_{N_B}^B + kL/S$ *and* $\ell_1^C + kL/S, \ldots, \ell_{N_C}^C + kL/S$*. Then* $R \in \mathcal{BC}(S, N_B, N_C)$*.*

Proof. See Appendix 2.A. □

Theorem 2.9 (Centrosymmetric R)**.** *Suppose that Assumption 2.7 holds and that* $\beta(L/2 + \ell) = \beta(L/2 - \ell)$*. In addition, suppose that the position of the monitors and magnets is reflection-symmetric as well, i.e. for each* $\ell_k^{(\cdot)}$ *there is a* $\ell_p^{(\cdot)}$ *s.t.* $\ell_p^{(\cdot)} = L - \ell_k^{(\cdot)}$ *for* $(\cdot) = \{B, C\}$*. Then* $R \in \mathcal{CS}(SN_B/2, SN_C/2)$*.*

Proof. See Appendix 2.A. □

Theorems 2.8 and 2.9 are intuitive results. If the magnetic lattices, BPMs and corrector magnets are arranged in a symmetric pattern, then the ORM inherits the same symmetric pattern. The $\mathcal{BC}$ property means that a circulant shift of N_B and N_C elements can be applied to the beam displacement y and magnet inputs u in (1.3) without changing the system behavior, while the $\mathcal{CS}$ property means that each vector can be mirror-reflected about its middle. If Theorems 2.8 and 2.9 simultaneously hold, each superperiod is centrosymmetric and the ORM inherits additional properties. These additional properties allow the $\mathcal{BC}$ decomposition to be split into real and imaginary parts and will further simplify calculations involving R.

Corollary 2.10 (Centrosymmetric and block-circulant R)**.** *Suppose that the conditions in Theorems 2.8 and 2.9 all hold, so that* $R \in \mathcal{BC}(S, N_B, N_C)$ *and* $R \in \mathcal{CS}(SN_B/2, SN_C/2)$*, and let* $R_k \in \mathbb{R}^{N_B \times N_C}$*,* $k = 0, \ldots, S-1$*, be the block-circulant partitioning of* R *from Def. 2.4. Then:*

2.10a $R_0, R_{S/2} \in \mathcal{CS}(N_B/2, N_C/2)$

2.10b $R_{S/2+k} J_{N_C} = J_{N_B} R_{S/2-k}$

Proof. See Appendix 2.A. □

2.3 Decompositions from Structural Symmetries

The $\mathcal{BC}$ property ensures that R can be block-diagonalised by pre- and post-multiplication with the discrete Fourier matrix F_S given in (2.5). By defining $\hat{y}(s) \coloneqq (F_S^* \otimes I_{N_B}) y(s)$, $\hat{u}(s) \coloneqq (F_S^* \otimes I_{N_C}) u(s)$ and $\hat{d}(s) \coloneqq (F_S^* \otimes I_{N_B}) d(s)$, the dynamics (1.3) can be mapped into the discrete (spatial) Fourier domain as

$$\hat{y}(s) = \hat{R} g(s) \hat{u}(s) + \hat{d}(s), \tag{2.16}$$

where $\hat{R} \coloneqq \operatorname{diag}\big(\hat{R}_0, \ldots, \hat{R}_{S-1}\big)$. When a vector is mapped into the Fourier domain as in $\hat{y}(s) = (F_S^* \otimes I_{N_B}) y(s)$, the Kronecker product between F_S^* and I_{N_B} means that the kth displacements of each superperiod are grouped. The Fourier transform is then applied to equidistant samples at $\ell_m^B + kL$, $k = 0, \ldots, S-1$. This yields the Fourier coefficients for the spatial frequencies $\vartheta_k = 2\pi k/S$. The block-diagonal structure of $\hat{R}$ means that the spatial Fourier coefficients of the displacements at frequency ϑ_k are not modified by magnetic inputs at frequency ϑ_j for $k \neq j$. The Fourier coefficients are, however, influenced by other Fourier coefficients of the same spatial frequency that have a different starting point ℓ_n^C for the equidistant samples $\ell_n^C + kL$.

Analogous to the $\mathcal{BC}$ case, the dynamics (1.3) can be mapped to the $\mathcal{CS}$ domain by defining $\hat{y}(s) \coloneqq T_{SN_B/2}^{\mathrm{T}} y(s)$, $\hat{u}(s) \coloneqq T_{SN_C/2}^{\mathrm{T}} u(s)$ and $\hat{d}(s) \coloneqq T_{SN_B/2}^{\mathrm{T}} d(s)$. The resulting $\hat{R} \coloneqq T_{SN_B/2}^{\mathrm{T}} R T_{SN_C/2}$ is block-diagonal if and only if R is $\mathcal{CS}$. The transformation $\hat{y}(s) = T_{SN_B/2}^{\mathrm{T}} y(s)$ groups elements k and $k + SN_B/2$ of $y(s)$ and assigns their sum and differences to $\hat{y}(s)$. The block-diagonalised $\hat{R}$ reflects the fact that the sum (difference) of the displacements, is solely modified by the sum (difference) of the effects of the magnets.

When R is *both* $\mathcal{BC}$ and $\mathcal{CS}$, the matrices $\hat{R}$ and $\hat{R}$ can both be further decomposed. For the decomposition of $\hat{R}$, the complex-valued blocks of the $\mathcal{BC}$ decomposition $\hat{R}$ are rewritten using (2.7) as

$$\hat{R}_n = R_0 + (-1)^n R_{S/2} + \sum_{k=1}^{S/2-1} \left(R_k \mathrm{e}^{-i\frac{2\pi nk}{S}} + J_{N_B} R_k J_{N_C} \mathrm{e}^{i\frac{2\pi nk}{S}} \right),$$

where the second part of Theorem 2.10 was used after reformulating it as $R_{S-k} = J_{N_B} R_k J_{N_C}$. Separating the real and imaginary parts of $\hat{R}_n$ yields

$$\begin{aligned} \mathrm{Re}\big(\hat{R}_n\big) &= R_0 + (-1)^n R_{S/2} + \sum_{k=1}^{S/2-1} \cos(2\pi nk/S)(R_k + J_{N_B} R_k J_{N_C}), \\ \mathrm{Im}\big(\hat{R}_n\big) &= \sum_{k=1}^{S/2-1} \sin(2\pi nk/S)(J_{N_B} R_k J_{N_C} - R_k). \end{aligned}$$

A common feature of matrices with symmetric structures, such as $\mathcal{BC}$ or $\mathcal{CS}$ matrices, is that they form an algebra (see Appendix 2.B). By pre- and post-multiplying with J_{N_B} and J_{N_C}, respectively, it can be shown that $\mathrm{Re}\big(\hat{R}_n\big)$ is $\mathcal{CS}$, while $\mathrm{Im}\big(\hat{R}_n\big)$ is $\mathcal{SCS}$. Each of the Fourier blocks $\hat{R}_n$ can therefore be pre- and post-multiplied by $T^{\mathrm{T}}_{N_B/2}$ and $T_{N_C/2}$, which will separate the real and imaginary part because $\mathcal{CS}$ matrices are block-diagonalised, while $\mathcal{SCS}$ matrices are block anti-diagonalised by the transformation (2.11).

The decomposition of the $\mathcal{CS}$ decomposition $\hat{R}$ can be found in Appendix 2.B, where it is shown that if R is $\mathcal{BC}$ as well, then each of the blocks of $\hat{R}$ is $\mathcal{CS}$ and can be decomposed using (2.10). Note that the $\mathcal{BC}$ structure is a more stringent requirement than needed, i.e. the doubly $\mathcal{CS}$ decomposition only requires that the β-function is $\mathcal{CS}$ with respect to $L/4$. Table 2.1 summarises the results from this section and characterises the $\mathcal{BC}$, $\mathcal{CS}$ and their further decompositions by showing the formulae for block-diagonalisation and the resulting sparsity patterns of the block-diagonalised matrices. The table also addresses the structured approximations $R^{\mathrm{F}}_{\mathcal{S}} \in \mathcal{S}$ for the case that the ORM is given by $R \coloneqq R^{\mathrm{F}}_{\mathcal{S}} + \Delta$, where $\Delta = R - R^{\mathrm{F}}_{\mathcal{S}} \in \mathcal{S}^{\perp}$ (Section 2.4.1). In principle, the symmetric decompositions could be applied to any matrix $R \notin \mathcal{S}$. The transformed matrix $\hat{R}$, however, is block-diagonal iff the original matrix has the corresponding symmetric structure, i.e. $R \in \mathcal{S}$.

Table 2.1: Symmetric decompositions $\hat{R}_{\mathcal{S}} = \mathcal{T}^* R \mathcal{T}$ for $S = 6$ and $\mathcal{S} \in \{\mathcal{CS}, \mathcal{CS} - \mathcal{BC}, \mathcal{BC}, \mathcal{BC} - \mathcal{CS}\}$. The table also shows the Frobenius norm approximations $R^{\mathrm{F}}_{\mathcal{S}}$ and the sparsity pattern of the approximation error $\Delta := \mathcal{T}^*(R^p - R^{\mathrm{F}}_{\mathcal{S}})\mathcal{T}$ in symmetric domain.

Centrosymmetric decomposition ($\mathcal{S} = \mathcal{CS}$)	
Diagonalisation	$\hat{R}_{\mathcal{S}} = T^{\mathrm{T}}_{SN_B/2} R T_{SN_C/2}$
Approximation†	$R^{\mathrm{F}}_{\mathcal{S}} = \frac{1}{2}(R^p + J_{SN_B} R^p J_{SN_C})$
Sparsity patterns of $\hat{R}_{\mathcal{S}}$ and $\hat{\Delta}_{\mathcal{S}}$	
Decomposition of the $\mathcal{CS}$ decomposition ($\mathcal{S} = \mathcal{CS} - \mathcal{BC}$)	
Diagonalisation	$\hat{R}_{\mathcal{S}} = \Big(I_2 \otimes T^{\mathrm{T}}_{SN_B/4}\Big) T^{\mathrm{T}}_{SN_B/2} R T_{SN_C/2} \Big(I_2 \otimes T_{SN_C/4}\Big)$
Approximation†	$R^{\mathrm{F}}_{\mathcal{S}} = \frac{1}{2S} \sum_{k=0}^{S-1} (\Omega_S^k \otimes I_{N_B})^{\mathrm{T}} (R^p + J_{SN_B} R^p J_{SN_C})(\Omega_S^k \otimes I_{N_C})$
Sparsity patterns of $\hat{R}_{\mathcal{S}}$ and $\hat{\Delta}_{\mathcal{S}}$	
Block-circulant decomposition ($\mathcal{S} = \mathcal{BC}$)	
Diagonalisation	$\hat{R}_{\mathcal{S}} = (F_S^* \otimes I_{N_B}) R (F_S \otimes I_{N_C})$
Approximation†	$R^{\mathrm{F}}_{\mathcal{S}} = \frac{1}{S} \sum_{k=0}^{S-1} (\Omega_S^k \otimes I_{N_B})^{\mathrm{T}} R^p (\Omega_S^k \otimes I_{N_C})$
Sparsity patterns of $\hat{R}_{\mathcal{S}}$ and $\hat{\Delta}_{\mathcal{S}}$	
Decomposition of the $\mathcal{BC}$ decomposition ($\mathcal{S} = \mathcal{BC} - \mathcal{CS}$)	
Diagonalisation	$\hat{R}_{\mathcal{S}} = \Big(I_S \otimes T^{\mathrm{T}}_{N_B/2}\Big)(F_S^* \otimes I_{N_B}) R (F_S \otimes I_{N_C})\Big(I_S \otimes T_{N_C/2}\Big)$
Approximation†	$R^{\mathrm{F}}_{\mathcal{S}} = \frac{1}{2S} \sum_{k=0}^{S-1} (\Omega_S^k \otimes I_{N_B})^{\mathrm{T}} (R^p + J_{SN_B} R^p J_{SN_C})(\Omega_S^k \otimes I_{N_C})$
Sparsity patterns of $\hat{R}_{\mathcal{S}}$ and $\hat{\Delta}_{\mathcal{S}}$	

Light-gray, dark-gray and striped blocks refer to real, purely imaginary and complex-valued numbers, respectively. †The matrix Ω_S is defined in (2.2).

2.4 Broken Symmetry

If the symmetric structure of the synchrotron is to be exploited for the controller, then the matrix K defined in (1.24) must have the same symmetry properties. If R is $\mathcal{BC}$ and/or $\mathcal{CS}$, the gain matrix K will necessarily have the symmetry properties, because each of the symmetric structures form an algebra. In practice, the regular arrangement of magnetic lattices, BPMs and corrector magnets is compromised by space constraints, e.g. there will be one section where the injection device – the entry point for the electrons – will need to be fitted. This will lead to an

asymmetric placement of one or more of the these components or to an asymmetry of the β-function. Because the symmetry of the ORM is based on the symmetry of the betatron function as well as on the placement of monitors and magnets, the ORM R will inherit any asymmetry. In these cases, the symmetry can be recovered by approximating the ORM. In some synchrotrons, the symmetry is also broken because additional monitors or magnets are inserted into individual cells, e.g. due to special requirements of a beamline, which leads to a partial symmetry of R. Even when R is block-wise symmetric, it is still possible to exploit the structure of the symmetric block.

2.4.1 Structured Approximation

Given a perturbed ORM $R \notin \mathcal{S}$ that does *not* satisfy the symmetry conditions, a matrix $R_{\mathcal{S}}^{\mathrm{F}} \in \mathcal{S}$ can be computed that approximates R and has the $\mathcal{BC}$ and/or $\mathcal{CS}$ properties. This problem can be formulated as an optimisation problem,

$$R_{\mathcal{S}}^{\mathrm{F}} := \underset{X \in \mathcal{S}}{\arg\min} \|X - R\|_{\mathrm{F}}^2, \tag{2.17}$$

where $\mathcal{S} \in \{\mathcal{BC}, \mathcal{CS}, \mathcal{BC} \cap \mathcal{CS}\}$. The Frobenius norm was used because it leads to closed-form solutions, which are analysed in more detail in Chapter 3. In [27], a solution is derived for the circulant symmetry and the approximation is applied to an orbit correction scheme in [105]. In Appendix 2.C, the proofs are extended for $\mathcal{S} \in \{\mathcal{BC}, \mathcal{CS}, \mathcal{BC} \cap \mathcal{CS}\}$. The results obtained are summarised in Table 2.1. They essentially consist of averaging over the sub-blocks of R according to the corresponding structure of $\mathcal{S}$, e.g. when $\mathcal{S} = \mathcal{BC}$ the diagonal block $R_{\mathcal{BC},0}^{\mathrm{F}}$ is obtained from averaging over all sub-blocks of R lying on the diagonal.

2.4.2 Approximation Error

When the gain matrix K is computed using the approximation $R_{\mathcal{S}}^{\mathrm{F}}$ instead of R, the stability properties of the resulting closed-loop system might be considerably affected, i.e. the system might be stable if K is computed using R but unstable when computed using $R_{\mathcal{S}}^{\mathrm{F}}$. To quantify the amount of asymmetry, an approximation error

is defined as $\Delta := R - R^{\mathrm{F}}_{\mathcal{S}}$. For a structured approximation problem, such as (2.17), the structure of Δ can be determined by transforming the optimisation (2.17) into the structural domain $\mathcal{S}$, i.e. by rewriting the norm in (2.17) using the symmetric transformations $\mathcal{T}_{\mathcal{S},l}, \mathcal{T}_{\mathcal{S},r}$ as

$$\|\mathcal{T}^*_{\mathcal{S},l}(X - R)\mathcal{T}_{\mathcal{S},r}\|_{\mathrm{F}} = \|\hat{X} - \hat{R}_{\shortparallel} - \hat{R}_{\perp}\|_{\mathrm{F}}, \tag{2.18}$$

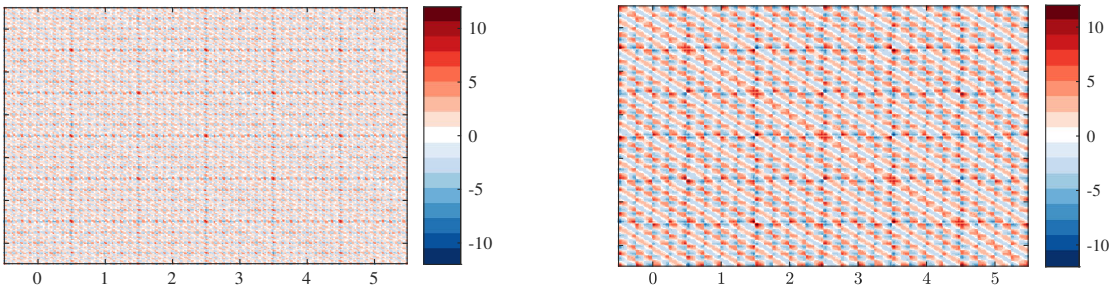
where $\hat{X} = \mathcal{T}^*_{\mathcal{S},l} X \mathcal{T}_{\mathcal{S},r}$, $\hat{R}_{\shortparallel} + \hat{R}_{\perp} = \mathcal{T}^*_{\mathcal{S},r} R \mathcal{T}_{\mathcal{S},r}$ and where $\hat{R}_{\shortparallel}$ has the same block-diagonal structure as $\hat{X}$ has such that

$$\mathrm{Re}\big(\hat{R}_{\perp}\big) \circ \mathrm{Re}\big(\hat{R}_{\shortparallel}\big) = 0, \qquad \mathrm{Im}\big(\hat{R}_{\perp}\big) \circ \mathrm{Im}\big(\hat{R}_{\shortparallel}\big) = 0. \tag{2.19}$$

Note that the Frobenius norm is invariant with respect to multiplication by an orthonormal matrix [55, Ch. 2.3.5] and that (2.19) would not necessarily hold if a different norm was used in (2.17). From (2.18), it becomes clear that $R^{\mathrm{F}}_{\mathcal{S}} = \hat{R}_{\shortparallel}$ and $\hat{\Delta} = \hat{R}_{\perp}$ and the solution to (2.17) could be found by setting $R^{\mathrm{F}}_{\mathcal{S}} = \mathcal{T}_{\mathcal{S},l} \hat{R}^{\mathrm{F}}_{\mathcal{S}} \mathcal{T}^*_{\mathcal{S},r}$. The sparsity patterns of $\hat{\Delta}$ for $\mathcal{S} \in \{\mathcal{BC}, \mathcal{CS}, \mathcal{BC} \cap \mathcal{CS}\}$ are depicted in Table 2.1. Note that for the doubly $\mathcal{CS}$ decomposition (column $\mathcal{CS} - \mathcal{BC}$ in Table 2.1) the approximation that yields $R^{\mathrm{F}}_{\mathcal{S}} \in \mathcal{BC} \cap \mathcal{CS}$ is used and (2.19) therefore does not hold.

2.4.3 Partial Symmetry

Some synchrotrons, such as the ESRF-EBS or MAX IV 3 GeV synchrotron, insert additional magnets or monitors into certain cells. In these cases, the number of monitors or magnets differ between different cells, which prohibits the symmetric decomposition of the entire ORM. If, for example, the number of additional magnets is small, it is still possible to exploit the symmetric structure. Let $R = \begin{bmatrix} R_{\mathrm{sym}} & R_{\mathrm{asym}} \end{bmatrix}$ be the ORM, where R_{asym} represents the asymmetric part of R associated to the additional magnets. The input transformation matrix can be redefined as $\hat{\mathcal{T}}_{\mathcal{S},r} := \mathrm{diag}(\mathcal{T}_{\mathcal{S},r}, I)$, where the dimension of I matches the number of additional magnets. The ORM is transformed as $\hat{R} = \mathcal{T}^*_{\mathcal{S},l} R \hat{\mathcal{T}}_{\mathcal{S},r} = \begin{bmatrix} \hat{R}_{\mathrm{sym}} & \hat{R}_{\mathrm{asym}} \end{bmatrix}$, where $\hat{R}_{\mathrm{sym}}$ is block-diagonal and $\hat{R}_{\mathrm{asym}}$ is dense. Instead of computing the gain matrix



(a) Horizontal plane. **(b)** Vertical plane.

Figure 2.1: Diamond-II orbit response matrices with $S = 6$ sections. The colors represent the magnitude of the matrix elements of R.

$\hat{K} = \left(\hat{R}^*\hat{R} + \mu I\right)^{-1}\hat{R}^*$, which would not result in a block-diagonal matrix, it is possible to avoid the pseudo-inversion of $\hat{R}$ by solving the linear system

$$\left(\begin{bmatrix} \hat{R}^*_{\text{sym}}\hat{R}_{\text{sym}} & \hat{R}^*_{\text{sym}}\hat{R}_{\text{asym}} \\ \hat{R}^*_{\text{asym}}\hat{R}_{\text{sym}} & \hat{R}^*_{\text{asym}}\hat{R}_{\text{asym}} \end{bmatrix} + \mu I\right)\hat{u} = \begin{bmatrix} \hat{R}^*_{\text{sym}} \\ \hat{R}^*_{\text{asym}} \end{bmatrix}\hat{y}. \tag{2.20}$$

The left-hand side matrix is Hermitian and in block-arrowhead form. There exist efficient and sparsity-preserving methods for solving these types of linear systems [56]. If the dimensions of $\hat{R}_{\text{asym}}$ are small compared to $\hat{R}_{\text{sym}}$ and the left-hand side matrix is pre-factorised offline, solving (2.20) is expected to be faster than the matrix-vector product in (2.1).

2.5 Case Study: Diamond-II

2.5.1 Structured Approximations

The Diamond-II storage ring will be arranged in $S = 6$ superperiods with $N_B = 44$ BPMs and $N_C = 66$ corrector magnets per superperiod [2, Ch. 2.1.1.2] and for this case study, the ORM for the vertical plane is used, which is shown in Fig. 2.1b. It can be seen that the Diamond-II storage ring has an accurate $\mathcal{BC}$ pattern. In addition, the lattices of each cell are mirror-reflected at the centre of each cell, which yields a β function with a $\mathcal{CS}$ pattern [4, Fig. 2.3]. The corrector magnets are realised by using additional windings in the sextupole magnets [4, Ch. 2.7.3.5] and the monitors are arranged in a $\mathcal{CS}$ pattern. According to Theorems 2.8-2.10, this leads to $\mathcal{CS}$, $\mathcal{BC}$ and $\mathcal{BC} \cap \mathcal{CS}$ properties of the ORM. In practice, the $\mathcal{BC}$ property is broken

Table 2.2: Approximation error $\Delta = R - R^{\mathrm{F}}_{\mathcal{S}}$ relative to R computed as $\|\Delta\|_2/\|R\|_2$ (2-norm), $\sum_{i,j}|\Delta_{(i,j)}|/\sum_{i,j}|R_{(i,j)}|$ (mean) and $\max_{i,j}|\Delta_{(i,j)}|/\max_{i,j}|R_{(i,j)}|$ (max-norm).

Synchrotron	$\mathcal{S}$	2-norm (%)	Mean (%)	Max-norm (%)
DLS-II	$\mathcal{BC}$	0.006	0.004	0.008
	$\mathcal{CS}$	7.764	5.083	12.552
	$\mathcal{BC} \cap \mathcal{CS}$	7.764	5.083	12.553
MAX IV	$\mathcal{BC}$	0.589	0.255	2.337
	$\mathcal{CS}$	2.646	2.192	3.36
	$\mathcal{BC} \cap \mathcal{CS}$	2.662	2.243	3.517
ESRF-EBS	$\mathcal{BC}$	4.523	1.072	30.809
	$\mathcal{CS}$	2.663	1.119	6.386
	$\mathcal{BC} \cap \mathcal{CS}$	4.692	1.932	30.839
ALBA	$\mathcal{BC}$	2.007	1.957	6.269
	$\mathcal{CS}$	4.234	2.849	14.998
	$\mathcal{BC} \cap \mathcal{CS}$	4.339	3.309	15.526

by the injection device, and irregular placements of BPMs and corrector magnets cause small deviations from the $\mathcal{CS}$ pattern in each superperiod [2, Ch. 2.1.1.2]. To recover the advantages of structural symmetries, the ORM is approximated using the formulae given in Table 2.1.

The first rows of Table 2.2 show the Diamond-II approximation error relative to R, i.e. $\|R - R^{\mathrm{F}}_{\mathcal{S}}\|/\|R\|$, taking as the matrix norm the standard matrix 2-norm (maximum singular value), the mean absolute entry, or the maximum absolute entry. The errors can be compared to the uncertainty introduced during the measurement of the ORM (see e.g. [47, Fig. 4.8]), which lies between 3 % and 8 %. The $\mathcal{BC}$ errors are 3 orders of magnitude smaller than the measurement uncertainty, which indicates that $\mathcal{BC}$ symmetry is an accurate assumption. Because the $\mathcal{BC}$ error is small, the approximation errors for the $\mathcal{CS}$ and $\mathcal{BC} \cap \mathcal{CS}$ symmetries are almost identical. These errors are significantly larger than for the $\mathcal{BC}$ symmetry, but remain within the same order of magnitude as the measurement uncertainty. For the stability of the controller, it is particularly relevant which modes are affected by uncertainty [47, Fig. 4.16]. For the $\mathcal{CS}$ and the $\mathcal{BC} \cap \mathcal{CS}$ approximations, the approximation error is concentrated in the first 5 low-order modes associated with

large singular values, which are less prone to instabilities than higher-order modes. Even though the $\mathcal{CS}$ and the $\mathcal{BC} \cap \mathcal{CS}$ errors seem large in magnitude, the particular distribution of the error onto the modes suggests that such an error is tolerable when the system is controlled in a feedback loop.

Table 2.2 also shows the approximation errors for the vertical planes of: the MAX IV 3 GeV storage ring, which has $S = 20$ cells and $p = m = 10$ monitors and magnets per cell; the ESRF-EBS storage ring with $S = 32$, $p = 10$ and $m = 9$; the ALBA storage ring with $S = 4$ and $p = m = 22$. The MAX IV ring has one additional monitor that breaks the symmetry and must be included using the procedure outlined in Section 2.4.3. Compared to Diamond-II, the error of the MAX IV ring for the $\mathcal{BC}$ symmetry is significantly larger, whereas the error for the $\mathcal{CS}$ symmetry is three times smaller. Similar observations can be made for the ESRF-EBS storage ring, whose symmetry is broken in one of the superperiods. This causes a 30 % error for the maximum norm of the $\mathcal{BC}$ approximation, but the average error remains at 1 %. These matrices will not be examined further, but preliminary simulations showed that the trajectory error correction would barely be affected by the structured approximations. Compared to Diamond-II, the larger number of cells at ESRF-EBS and MAX IV would also yield larger speed improvements.

2.5.2 Orbit Feedback Controller

As a proof of concept, the following analysis is focused on how the standard controller (2.1) performs when K is obtained using a structured approximation, i.e. $K = ((R^{\mathrm{F}}_{\mathcal{S}})^{\mathrm{T}} R^{\mathrm{F}}_{\mathcal{S}} r + \mu I)^{-1} (R^{\mathrm{F}}_{\mathcal{S}})^{\mathrm{T}}$, while the process model is given by the asymmetric Diamond-II ORM $R \notin \mathcal{S}$. The results are presented for the vertical plane and are comparable for the horizontal plane.

As opposed to the FOFB at Diamond, the FOFB at Diamond-II will use 252 slow and 144 fast corrector magnets. Even though the *locations* of the 396 slow and fast magnets produce an overall ORM R that has an accurate $\mathcal{BC}$ symmetry, the *distribution* of slow and fast magnets onto these locations and across the $S = 6$ superperiods is irregular. In other words, after selecting the columns of R to

obtain R_s and R_f from the two-array system (1.5), the $\mathcal{BC}$ symmetry is lost and the matrices R_s and R_f cannot be separately decomposed any more. For the following simulations, it is therefore assumed that all corrector magnets are identical and are modelled by the actuator model (1.2).

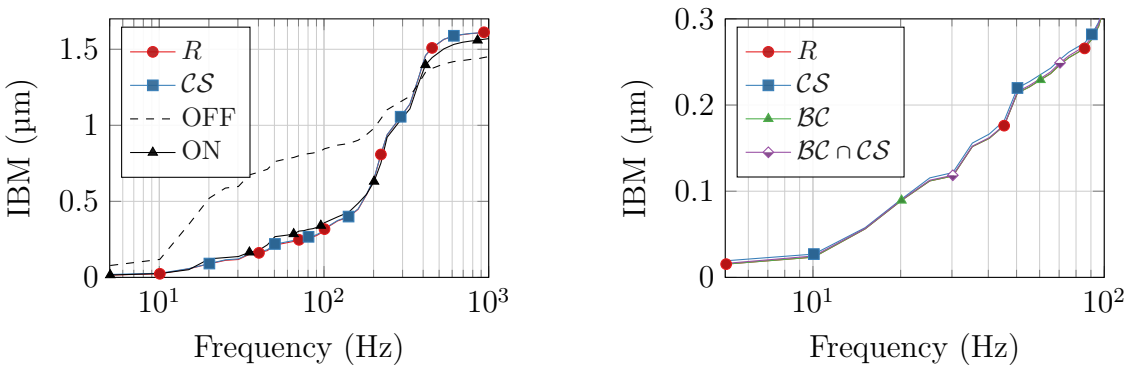
The simulation requires the disturbances $d(t)$ as an input. Because no such measurements are yet available for Diamond-II, measurements from the Diamond storage ring are used. The disturbance vector is augmented to fit the dimensions of Diamond-II. Firstly, the $80 = 252 - 172$ monitor outputs, which are lying opposite of the ring-half that contains the injection device, are copied and appended to the 172 measurements. Secondly, the augmented disturbances are transformed into mode-space using an SVD of the Diamond-II ORM and the power spectrum of the modes plotted, such as in [47, pp. 68–72]. The disturbance spectrum is then scaled to obtain a power spectrum comparable to [47, Fig. 3.11], where the modes associated with large-magnitude singular values show a larger amplitude. The resulting disturbance profile is depicted in Fig. 2.2a (labeled by Measured, off).

The performance of the controller is measured using the integrated beam motion (IBM), which is shown in Fig. 2.2. Fig. 2.2a shows the average IBM across all BPMs of the storage ring for K computed using R and the $\mathcal{CS}$ approximation of R. For clarity, the simulation results for the $\mathcal{BC}$ and $\mathcal{BC} \cap \mathcal{CS}$ approximations are omitted in Fig. 2.2a, but are shown in the close-up in Fig. 2.2b. The results show that the controller performance is only slightly worse when a structured approximation is used. In Fig. 2.2b, it can be seen that the $\mathcal{CS}$ approximation yields a slightly larger average trajectory error, which is related to the larger approximation error.

The nominal stability of the controller, i.e. when no uncertainty is present, can be verified by calculating the poles of the closed-loop transfer functions. The closed-loop transfer functions of the standard feedback structure are given by

$$y(s) = (I + Rg(s)Kc(s))^{-1} d(s), \tag{2.21a}$$

$$u(s) = Kc(s)(I + Rg(s)Kc(s))^{-1} d(s). \tag{2.21b}$$



(a) $\mathcal{CS}$ approximation. **(b)** $\mathcal{CS}$, $\mathcal{BC}$ and $\mathcal{BC} \cap \mathcal{CS}$ approximations.

Figure 2.2: Comparison of average IBM for disabled/enabled feedback with simulations of the controller using the $\mathcal{CS}$, $\mathcal{BC}$ and $\mathcal{BC} \cap \mathcal{CS}$ approximation for the vertical plane.

where $Rg(s)$ is the asymmetric plant. The gain K is given (1.24) and rewritten as

$$K = \left(X^{\mathrm{T}}X + \mu I\right)^{-1} X^{\mathrm{T}}, \tag{2.22}$$

where $X = R_{\mathcal{S}}^{\mathrm{F}}$ if the Frobenius norm approximation is used and $X = R$ otherwise. The feedback system is stable if all poles of (2.21a) and (2.21b) have negative real parts. The poles do not significantly differ in magnitude for K computed using R as well as using the structured approximations and have negative real parts, which indicates that all systems are nominally stable.

2.5.3 Performance for Implementation

The Diamond FOFB is implemented on 24 processors, which are distributed around the storage ring, while for Diamond-II the computations will be centralised and are tested and performed on a board that features a field programmable gate array (FPGA) and two DSPs, which are discussed in more detail in Chapter 7. The following simulation study will be performed on the DSP only. The clock-frequency of the DSP is 1.4 GHz and the targeted operating frequency of 100 kHz therefore allows for 14,000 processor cycles. Without symmetric decomposition, the controller computations require $396 \times 252 \approx 100,000$ multiply-accumulate operations, which equals the number of operations required for the standard approach (2.1). The dynamic part of the controller $c(s)$ requires $3 \times 396 \approx 2,400$ operations and

these are neglected in the following. To demonstrate the speed advantages of the symmetric decomposition, the matrix-vector multiplication required by the controller has been implemented on the processor. When a decomposition is used, the control input is computed as

$$u(s) = -\mathcal{T}_{\mathcal{S},l}\,\hat{K}\,\mathcal{T}_{\mathcal{S},r}^{\mathrm{T}}\,c(s)y(s), \tag{2.23}$$

where $\mathcal{S} \in \{\mathcal{CS}, \mathcal{BC}, \mathcal{CS}\text{ - }\mathcal{BC}, \mathcal{BC}\text{ - }\mathcal{CS}\}$ refers to the decompositions in Table 2.1 and $\mathcal{T}_{\mathcal{S},l}, \mathcal{T}_{\mathcal{S},r}$ to the corresponding transformation, and $\hat{K}$ is the decomposed gain matrix. As for the standard approach (2.1), the gain matrix $\hat{K}$ is computed using a regularised inverse, but because the decomposed matrices $\hat{R}$ are block-diagonal, $\hat{K}$ will have the same structure. The controller matrices $\hat{K}$ are pre-computed offline, while for the symmetric approaches the computation-efficient vector transformations $\mathcal{T}_{\mathcal{S},l}$ and $\mathcal{T}_{\mathcal{S},r}$ are applied online.

Fig. 2.3 shows the results that were obtained for the implementation of (2.23) on a single core of the processor. The performance is measured as $1/t$, where t is the time required to execute one matrix-vector multiplication, and the horizontal gray bars refer to the theoretical speed-up (2.8) that was calculated relative to the leftmost timing, which does not use a symmetric decomposition and corresponds the standard approach. While the $\mathcal{BC}$ decomposition speeds up the computations by a factor of 5.5, the computation frequency for the $\mathcal{BC}$ - $\mathcal{CS}$ decomposition is more than eleven times faster than without symmetric decomposition, which corresponds to a time-delay reduction of 82 μs compared to the standard approach, and exceeds the targeted operating frequency of 100 kHz. The significant performance difference between the $\mathcal{BC}$ and the $\mathcal{BC} \cap \mathcal{CS}$ cases is due to the fact that the $\mathcal{BC} \cap \mathcal{CS}$ decomposition separates the real and imaginary parts and obviates the need for complex arithmetic. The results also show that for the $\mathcal{BC}$ - $\mathcal{CS}$ decomposition the speed-up of the implementation is significantly larger than the theoretical prediction. The reason is that, in addition to the number of operations required, the performance of the processor is limited by memory operations, e.g. the time needed to transport the matrix data from the memory to the core. The reduced memory requirements of the decomposition indirectly benefit the computation time.

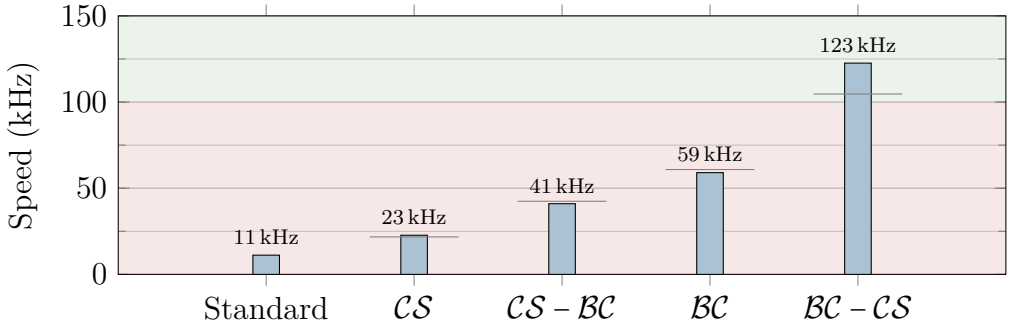


Figure 2.3: Performance measurement of the controller on a single core of the TI C6678 DSP @ 1 GHz. The gray bars refer to the theoretical speed-up. The leftmost measurement referes to the standard approach (2.1).

2.6 Conclusions

In this chapter, it was shown that the ORM of a synchrotron inherits the mirror-reflective and periodic properties of the betatron function. The algebra formed by matrices with structural symmetries guarantees that the gain matrix K of the SVD-based approach (2.22) inherits the structural symmetries allowing for K to be decomposed and the control inputs to be computed in the symmetric domain. The matrix-vector multiplication in the symmetric domain requires far fewer multiply-accumulate operations than in the original domain. In practice, the symmetry of the ORM might be broken due to irregular placements of monitors or magnets. To recover the symmetric structure, an optimisation problem was formulated in which a matrix was sought that has the corresponding symmetry properties and approximates the asymmetric ORM.

The chapter was concluded with a case study of the Diamond-II synchrotron. The Diamond-II ORM was approximated and the approximation error compared to other synchrotrons. It was shown that, despite larger approximation errors for the centrosymmetric case, there is only a small difference in trajectory error correction between the gain matrices computed using the approximations and using the asymmetric ORM. The case study was completed using a single-core implementation of the controller computations and compared the standard approach, which does not take advantage of the symmetry of the storage ring, to controllers that exploit the symmetry. A significant improvement of the computation-speed

was demonstrated when the matrix-vector multiplication was carried out in the symmetric domain. The combination of the block-circulant and centrosymmetric decompositions was 23 kHz above the targeted operating frequency of 100 kHz and one order of magnitude faster than the standard approach.

The complexity of the control algorithm is dominated by a matrix-vector multiplication, which can be implemented on parallel units, but the parallelisation has not been considered in this study. The processor used at Diamond-II has 8 cores and, without considering the cost of parallelisation, the distribution of the algorithm onto all cores would lead to a computation frequency of approximately 1 MHz. This could allow for the use of more advanced control algorithms that involve iterative schemes, such as MPC, which would benefit from executing large matrix-vector multiplications in the symmetric domain.

While it was shown how structured approximations allow an increase in the computation speed, the robust stability and performance of the resulting controllers were not addressed in detail, such as treated in [48], [106], where the closed-loop system was analysed under the influence of parameter uncertainty. The assumptions on the symmetric structure of the storage ring raise several questions. How can the degree of symmetry of a system be appropriately measured and to which extent can structured approximations be applied to asymmetric systems? For a given control approach, which level of asymmetry is acceptable before the trajectory error correction starts to deteriorate? This chapter attempted to measure the degree of symmetry by computing different norms of the approximation error. Even though these measures were significantly different for the block-circulant and centrosymmetric cases, the resulting trajectory error attenuation was almost identical, which suggests that these norms were not appropriately capturing the degree of symmetry. It was shown that the structured approximation error has a particular structure when it is mapped to the symmetric domain, and this observation will be used in Chapter 3 with the aim of combining the approximation and controller synthesis problems.

Appendix

2.A Proofs

Proof of Theorem 2.8. To show that $R \in \mathcal{BC}(S, N_B, N_C)$, partition the first N_B rows of $R \in \mathbb{R}^{SN_B \times SN_C}$ into $N_B \times N_C$ blocks. Let $\overline{\text{mod}}(\cdot)$ denote the modulo operation that is formulated as $\overline{\text{mod}}(n + kN_C) = (n + kN_C - 1 \mod SN_C) + 1$. According to the $\mathcal{BC}$ structure (2.3), it must be shown that $R_{m+kN_B,\overline{\text{mod}}(n+kN_C)} = R_{m,n}$ for $k = 1, \ldots, S-1$, $m = 1, \ldots, N_B$ and $n = 1, \ldots, N_C$. From the definition (2.14):

$$\begin{aligned} R_{m+kN_B,\overline{\text{mod}}(n+kN_C)} &= \frac{\sqrt{\beta^B_{m+kL/S}\beta^C_{n+kL/S}}}{2\sin(\pi Q)} \cos\Big(\pi Q - \big|\phi^B_{m+kL/S} - \phi^C_{n+kL}\big|\Big) \\ &= \frac{\sqrt{\beta^B_m \beta^C_n}}{2\sin(\pi Q)} \cos\left(\pi Q - \left|\phi^B_m + k\frac{2\pi Q}{S/L} - \left(\phi^C_n + k\frac{2\pi Q}{S/L}\right)\right|\right) \\ &= \frac{\sqrt{\beta^B_m \beta^C_n}}{2\sin(\pi Q)} \cos\big(\pi Q - \big|\phi^B_m - \phi^C_n\big|\big) = R_{m,n}, \end{aligned}$$

where it has been used that $\phi(s + kL) = \phi(s) + k\frac{2\pi Q}{L/S}$ for a periodic β, which can be verified from (2.15). □

Proof of Theorem 2.9. To show that $R \in \mathcal{CS}(SN_B/2, SN_C/2)$, the top-half of the matrix must be a vertically and horizontally reflected version of the bottom-half of the matrix. For the top-left and bottom-right sub-blocks of the matrix, $R_{SN_B/2-n,SN_C/2-m}$ must equal $R_{SN_B/2+n+1,SN_C/2+m+1}$ for all combinations of $n = 0, \ldots, \pm(SN_B/2-1)$ and $m = 0, \ldots, \pm(SN_C/2-1)$. After setting $\hat{s}^A_k = L/2 - \ell^A_{X+k}$ and

noting that $\phi(L/2 \pm s) = \phi(L/2) \pm \phi(s)$, one obtains:

$$\begin{aligned} R_{SN_B/2-n,SN_C/2-m} &= \frac{\sqrt{\beta(L/2-\hat{s}_n^B)\beta(L/2-\hat{s}_m^C)}}{2\sin(\pi Q)} \\ &\quad \times \cos\big(\pi Q - \big|\phi(L/2-\hat{s}_n^B) - \phi(L/2-\hat{s}_m^C)\big|\big), \\ &= \frac{\sqrt{\beta(L/2+\hat{s}_n^B)\beta(L/2+\hat{s}_m^C)}}{2\sin(\pi Q)} \underbrace{\cos\big(\pi Q - \big|\phi(\hat{s}_m^C) - \phi(\hat{s}_n^B)\big|\big)}_{=\cos(\pi Q-|\phi(L/2+\hat{s}_m^C)-\phi(L/2+\hat{s}_n^B)|)}, \\ &= R_{SN_B/2+n+1,SN_C/2+m+1}, \end{aligned}$$

and analogously for the top-right and bottom-left sub-blocks. □

Proof of Corollary 2.10. To find the algebraic conditions that a simultaneously $\mathcal{BC}$ and $\mathcal{CS}$ matrix satisfies, consider the permutation matrix J_n acting on the cyclic shift matrix Ω_n (2.2):

$$J_n \Omega_n J_n = \begin{bmatrix} 0 & 1 \\ I_{n-1} & 0 \end{bmatrix} = \Omega_n^{\mathrm{T}} = \Omega_n^{n-1}, \tag{2.24}$$

where for the rightmost equality it was considered that a cyclic downwards-shift ($\Omega_n^{\mathrm{T}} x$) of a vector of length n equals a cyclic upwards-shift by $n-1$ places ($\Omega_n^{n-1} x$). Using (2.24), one obtains

$$J\Omega_n^k J = \Omega_n^{n-1} J \Omega_n^{k-1} J = \cdots = \Omega_n^{kn-k} = \left(\Omega_n^{\mathrm{T}}\right)^k = \Omega_n^{n-k},$$

where $\Omega_n^{-1} = \Omega_n^{\mathrm{T}}$ and $\Omega_n^n = I_n$ was used. Consider a $\mathcal{BC}$ matrix $X \in \mathcal{BC}(n,l,m)$ represented as in (2.26). If X is also to be $\mathcal{CS}$, then

$$\sum_{k=0}^{n-1} \Omega_n^k \otimes x_k \overset{!}{=} J_{nl}\left(\sum_{k=0}^{n-1} \Omega_n^k \otimes x_k\right) J_{nm} = \sum_{k=0}^{n-1} J_n \Omega_n^k J_n \otimes J_l x_k J_m = \sum_{k=0}^{n-1} \Omega_n^{n-k} \otimes J_l x_k J_m.$$

Note that Ω_n^j has non-zero entries where Ω_n^k, $k \neq j$, has zero entries and vice-versa. Equating the terms with the same power of Ω_n yields

$$x_{n-k} = J_l x_k J_m, \tag{2.25}$$

and in particular $x_0 = J_l x_0 J_m$ and $x_{n/2} = J_l x_{n/2} J_m$. □

2.B Properties of Matrices with Structural Symmetries

Block-Circulant Matrices From Def. 2.4, it follows that any $B \in \mathcal{BC}(n,p,m)$ can be represented as [33]

$$B = \sum_{k=0}^{n-1} \Omega_n^k \otimes b_k. \tag{2.26}$$

When the $\mathcal{BC}$ matrix B in (2.6) is real, the blocks ν_j possess additional structure that is inherited from the Fourier matrix F_n. If n is even, ν_0 and $\nu_{n/2}$ are real while for $i = 1,\ldots,n/2-1$ it holds that $\nu_i = \bar{\nu}_{n-i}$. If n is odd, the only real-valued block is ν_0 and the latter holds for $i = 1,\ldots,(n-1)/2$. The same pattern of complex conjugates is exploited during a Fast Fourier Transformation and, according to the properties of ν_j, only the first $n/2$ blocks must be considered for a matrix-vector multiplication.

Centrosymmetric and skew-centrosymmetric matrices By reversing the order of the second half of the rows and columns, a $\mathcal{CS}$ matrix can be permuted into a $\mathcal{BC}$ matrix of order 2 [163]:

Lemma 2.11. *The permutation matrices $P_l = \operatorname{diag}(I_q, J_q)$ and $P_r = \operatorname{diag}(I_t, J_t)$ permute $R \in \mathcal{CS}(q,t)$ into a $\mathcal{BC}$ matrix of order $n = 2$:*

$$P_l R P_r = \left[\begin{array}{c:c} r_1 & r_2 J_t \\ \hdashline r_2 J_t & r_1 \end{array}\right] \in \mathcal{BC}(2,q,t).$$

Proof. Evaluating the product $P_l R P_r$ yields the result. □

Lemma (2.11) shows that $\mathcal{CS}$ matrices and $\mathcal{BC}$ matrices are closely related. For our purpose, it is sufficient to use Lemma (2.11) to show that (2.8) holds for $n = 2$.

Decomposition of the centrosymmetric decomposition When R is $\mathcal{BC}$ and $\mathcal{CS}$, the $\mathcal{CS}$ decomposition $\hat{R} = T_{SN_B/2}^{\mathrm{T}} R T_{SN_C/2}$ can be further decomposed. Consider the partitioning of R into four equal-sized blocks as in Def. 2.5 such

that $\hat{R} = \operatorname{diag}\big(R_1 - R_2 J_{SN_C/2}, R_1 + R_2 J_{SN_C/2}\big)$. From the $\mathcal{BC}$ structure (2.3), R_1 and R_2 are obtained as

$$R_1 = \begin{bmatrix} R_0 & R_1 & \dots & R_{\frac{S}{2}-1} \\ R_{S-1} & R_0 & \dots & R_{\frac{S}{2}-2} \\ \vdots & \vdots & \ddots & \vdots \\ R_{\frac{S}{2}+1} & R_{\frac{S}{2}+2} & \dots & R_0 \end{bmatrix}, \qquad R_2 = \begin{bmatrix} R_{\frac{S}{2}} & R_{\frac{S}{2}+1} & \dots & R_{S-1} \\ R_{\frac{S}{2}-1} & R_{\frac{S}{2}} & \dots & R_{S-2} \\ \vdots & \vdots & \ddots & \vdots \\ R_1 & R_2 & \dots & R_{\frac{S}{2}} \end{bmatrix}.$$

The matrices R_1, R_2 have the $\mathcal{CS}$ blocks R_0 and $R_{S/2}$ on their diagonals. In addition, the blocks opposite the diagonals are R_k and R_{k-1} for R_1 and $R_{S/2+k}$ and $R_{S/2-k}$ for R_2, i.e. the opposite blocks satisfy the second part of Theorem 2.10. This entails that $R_1, R_2 \in \mathcal{CS}(SN_B/4, SN_C/4)$. Note that if R_2 is $\mathcal{CS}$, then so is $R_2 J_{S/2N_C}$. Because the sum of two $\mathcal{CS}$ matrices is also $\mathcal{CS}$, the blocks of $\hat{R}$ are $\mathcal{CS}$ and each one of the blocks can be further decomposed by pre- and post-multiplication with $T^{\mathrm{T}}_{SN_B/4}$ and $T_{SN_C/4}$, respectively. In case SN_B or SN_C are not divisible by 4, the decomposition is still possible but a different transformation matrix must be used [163].

2.C Frobenius Norm Approxmiations

Block-circulant approximation The $\mathcal{BC}$ approximation can be found in [27] and it is summarised here in support of subsequent results. For approximating a matrix $R \in \mathbb{R}^{nl \times nm}$ with a matrix $X \in \mathcal{BC}(n, l, m)$, the optimisation (2.17) is reformulated as

$$\underset{x_0,\dots,x_{n-1} \in \mathbb{R}^{l \times m}}{\text{minimise}} \; \Big\| \sum_{k=0}^{n-1} \Omega_n^k \otimes x_k - R \Big\|_{\mathrm{F}}^2, \tag{2.27}$$

where $x_k \in \mathbb{R}^{l \times m}$, X was partitioned as in (2.3) and the $\mathcal{BC}$ representation (2.26) was used. Because Ω_n^k has non-zero elements where Ω_n^j, $j \neq k$, has zero elements, i.e. $\sum_{k=0}^{S-1} \Omega_S^k = \mathbf{1}_{n,n}$, where $\mathbf{1}_{m,n} \in \mathbb{R}^{m \times n}$ is a matrix of ones, problem (2.27) can be rewritten as

$$\min_{\{x_k\}_{k=0}^{n-1}} \sum_{k=0}^{n-1} \big\| \Omega_n^k \otimes x_k - \big(\Omega_n^k \otimes \mathbf{1}_{l,m}\big) \circ R \big\|_{\mathrm{F}}^2. \tag{2.28}$$

By partitioning R into blocks $R_{i,j} \in \mathbb{R}^{l \times m}$ with $i, j = 0, \dots, n-1$, each summand in (2.28) can be rewritten as

$$\sum_{j=0}^{n-1} \big\| x_k - R_{j,\, k+j \bmod n} \big\|_{\mathrm{F}}^2, \tag{2.29}$$

for $k = 0, \dots, n-1$. Using (2.28) and (2.29), the minimization (2.27) can be reformulated as

$$\underset{x_0,\dots,x_{n-1}\in\mathbb{R}^{l\times m}}{\text{minimise}} \sum_{k=0}^{n-1}\sum_{j=0}^{n-1} \|x_k - R_{j,k+j \bmod n}\|_{\mathrm{F}}^2. \tag{2.30}$$

The minimum of (2.30) is attained where its derivative is zero. This can be done element-wise for each element of x_k which – after reconstructing blocks x_k – yields $x_k^\star = \frac{1}{n}\sum_{j=0}^{n-1} R_{j,k+j \bmod n}$. Using the cyclic shift matrix (2.2), the solution is reconstructed as $X^\star = \sum_{k=0}^{n-1}(\Omega_n^k \otimes I_l)^{\mathrm{T}} R(\Omega_n^k \otimes I_m)/n$.

Centrosymmetric approximation For approximating a matrix $R \in \mathbb{R}^{2q\times 2t}$ with a matrix $Y \in \mathcal{CS}(q,t)$, the optimisation (2.17) is reformulated as

$$\underset{y_1,y_3\in\mathbb{R}^{q\times t}}{\text{minimise}} \left\| \left[\begin{array}{c:c} y_1 & J_q y_3 J_t \\ \hdashline y_3 & J_q y_1 J_t \end{array}\right] - R \right\|_{\mathrm{F}}^2. \tag{2.31}$$

If R is partitioned as $R = \left[\begin{array}{c:c} R_1 & R_2 \\ \hdashline R_3 & R_4 \end{array}\right]$ with $R_i \in \mathbb{R}^{q\times t}$, the minimization (2.31) can be reformulated as

$$\underset{y_1,y_3\in\mathbb{R}^{q\times t}}{\text{minimise}} \|y_1 - R_1\|_{\mathrm{F}}^2 + \|y_3 - J_q R_2 J_t\|_{\mathrm{F}}^2 + \|y_3 - R_3\|_{\mathrm{F}}^2 + \|y_1 - J_q R_4 J_t\|_{\mathrm{F}}^2 \tag{2.32}$$

where $\|UXV\|_{\mathrm{F}} = \|X\|_{\mathrm{F}}$ for orthonormal U, V was used [55, Ch. 2.3.5, p. 75]. The minimum of (2.32) is attained where its derivative is zero. The minimisers $y_1^\star$ and $y_3^\star$ are obtained as $y_1^\star = (R_1 + J_q R_4 J_t)/2$ and $y_3^\star = (R_3 + J_q R_2 J_t)/2$ and $Y^\star$ is reconstructed as $Y^\star = (R + J_{2q} R J_{2t})/2$.

Block-circulant and centrosymmetric approximation For approximating a matrix $R \in \mathbb{R}^{nl\times nm}$, where $n, l, m > 1$ are even, with a matrix $Z \in \mathcal{BC}(n,l,m) \cap \mathcal{CS}(nl/2, nm/2)$, the optimisation (2.17) is reformulated as in (2.27) and the blocks

$\{z_k\}_{k=n/2+1}^{n-1}$ are substituted using (2.25), which yields

$$\begin{aligned}
&\min_{\{z_k\}_{k=0}^{n-1}} \|I_n \otimes z_0 + \Omega_n^{n/2} \otimes z_{n/2} + \sum_{k=1}^{n/2-1} \left(\Omega_n^k \otimes z_k + \Omega_n^{n-k} \otimes J_l z_k J_m\right) - R\|_{\mathrm{F}}^2, \\
&= \min_{\{z_k\}_{k=0}^{n-1}} \Bigg(\|I_n \otimes z_0 - (I_n \otimes \mathbf{1}_{l,m}) \circ R\|_{\mathrm{F}}^2 + \|\Omega_n^{n/2} \otimes z_{n/2} - \left(\Omega_n^{n/2} \otimes \mathbf{1}_{l,m}\right) \circ R\|_{\mathrm{F}}^2 \\
&\qquad + \sum_{k=1}^{n/2-1} \Bigg(\|\Omega_n^k \otimes z_k - \left(\Omega_n^k \otimes \mathbf{1}_{l,m}\right) \circ R\|_{\mathrm{F}}^2 \\
&\qquad + \|\Omega_n^{n-k} \otimes J_l z_k J_m - \left(\Omega_n^{n-k} \otimes \mathbf{1}_{l,m}\right) \circ R\|_{\mathrm{F}}^2 \Bigg) \Bigg).
\end{aligned} \tag{2.33}$$

As for the $\mathcal{BC}$ approximation, the Frobenius norms can be separated for different powers of Ω_n. The terms for z_0 and $z_{n/2}$ can be rewritten as

$$\|\Omega_n^k \otimes z_k - \left(\Omega_n^k \otimes \mathbf{1}\right) \circ R\|_{\mathrm{F}}^2 = \sum_{j=0}^{n-1} \|z_k - \rho_{k,j}\|_{\mathrm{F}}^2, \tag{2.34}$$

where $k = \{0, n/2\}$, $\rho_{k,j} \coloneqq R_{(j,\, k+j \bmod n)}$ with R partitioned as for the $\mathcal{BC}$ approximation. According to (2.25), sub-blocks z_0 and $z_{n/2}$ must be $\mathcal{CS}$. Sub-blocks z_k and $\rho_{k,j}$ are partitioned as

$$z_k = \left[\begin{array}{c|c} z_k^1 & J_{l/2} z_k^3 J_{m/2} \\ \hline z_k^3 & J_{l/2} z_k^1 J_{m/2} \end{array}\right], \quad \rho_{k,j} = \left[\begin{array}{c|c} \rho_{k,j}^1 & \rho_{k,j}^2 \\ \hline \rho_{k,j}^3 & \rho_{k,j}^4 \end{array}\right],$$

and the right-hand side of (2.34) rewritten as

$$\begin{aligned}
\sum_{j=0}^{n-1} \big(&\|z_k^1 - \rho_{k,j}^1\|_{\mathrm{F}}^2 + \|z_k^3 - J_{l/2} \rho_{k,j}^2 J_{m/2}\|_{\mathrm{F}}^2 \\
&+ \|z_k^3 - \rho_{k,j}^3\|_{\mathrm{F}}^2 + \|z_k^1 - J_{l/2} \rho_{k,j}^4 J_{m/2}\|_{\mathrm{F}}^2 \big).
\end{aligned} \tag{2.35}$$

Note the similarity between (2.35) and (2.32). Setting the derivative of (2.35) to zero, solving for z_k^1, z_k^3 and reconstructing z_k yields for $k = \{0, n/2\}$

$$z_k^\star = \frac{1}{2n} \sum_{j=0}^{n-1} (\rho_{k,j} + J_l \rho_{k,j} J_m). \tag{2.36}$$

The summands in (2.33) for $k = 1, \ldots, n/2 - 1$ are rewritten as

$$\sum_{j=0}^{n-1} \|z_k - \rho_{k,j}\|_{\mathrm{F}}^2 + \underbrace{\|J_l z_k J_m - \rho_{n-k,j}\|_{\mathrm{F}}^2}_{= \|z_k - J_l \rho_{n-k,j} J_m\|_{\mathrm{F}}^2}.$$

Setting the derivative to zero yields $z_k^\star = 1/(2n) \sum_{j=0}^{n-1} (\rho_{k,j} + J_l \rho_{n-k,j} J_m)$ for $k = 1, \ldots, n/2 - 1$, which is identical to (2.36). After reconstruction, the matrix $Z^\star$ is obtained as $Z^\star = 1/(2n) \sum_{k=0}^{n-1} (\Omega_n^k \otimes I_l)^{\mathrm{T}} (R + J_{nl} R J_{nm}) (\Omega_n^k \otimes I_m)$.

3

Approximate Structural Symmetries

In Chapter 2, it was shown that the symmetry properties of the betatron function can produce an ORM $R \in \mathbb{R}^{n_y \times n_u}$ with $n_y = nb_y$ and $n_u = nb_u$ that has the structural symmetry $\mathcal{S}$. A structural symmetry $\mathcal{S}$ is associated with a permutation matrix Π, and any matrix R that satisfies $R(\Pi \otimes I_{b_u}) = \big(\Pi \otimes I_{b_y}\big)R$ can be block-diagonalised using the transformation matrix $T \in \mathbb{R}^{n \times n}$ associated with $\mathcal{S}$. As a consequence, the beam dynamics (1.3), $y(s) = Rg(s)u(s) + d(s)$, can be decoupled into smaller systems of size $b_y \times b_u$, which simplifies controller synthesis and can increase the computational performance of the controller implementation.

However, most systems encountered in practice only adhere approximately to a structural symmetry [28], [106], meaning that for any norm $\|\cdot\|$ it holds that

$$\left\| R(\Pi \otimes I_{b_u}) - \big(\Pi \otimes I_{b_y}\big)R \right\| =: \epsilon > 0, \tag{3.1}$$

where R refers to both the plant and the plant model, i.e. $\bar{R} = R$ in Fig. 1.3b, and plant uncertainty is considered in Section 3.3. In this case, $\hat{R} = (T^* \otimes I_{b_y})R(T \otimes I_{b_u})$ is *not* block-diagonal, so that the advantages of the transformation into symmetric domain are lost. To recover the structural symmetry of the beam dynamics (1.3),

This chapter is based on [87] I. Kempf, P. Goulart, and S. Duncan, *Control of cross-directional systems with approximate symmetries*, Jun. 2023. arXiv: 2306.17565 [eess.SY].

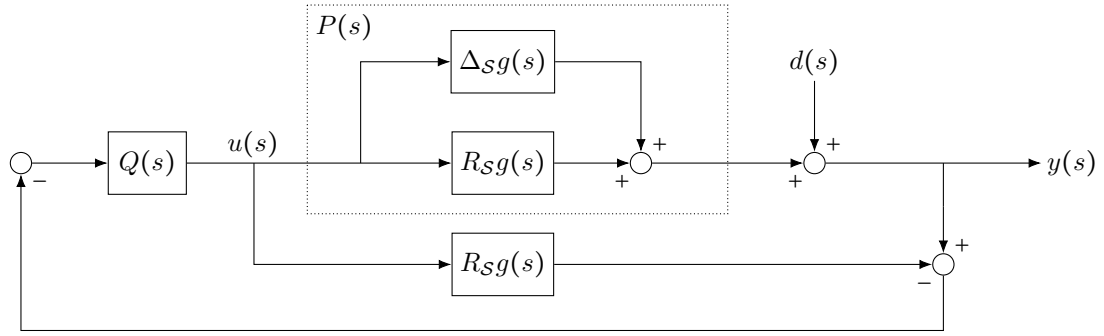


Figure 3.1: IMC structure with structured approximation $R_{\mathcal{S}}$ and approximation error $\Delta_{\mathcal{S}}$, where $P(s) = (R_{\mathcal{S}} + \Delta_{\mathcal{S}})g(s)$ is known and $\Delta_{\mathcal{S}}$ artificially introduced through $R_{\mathcal{S}}$.

one possibility is to split R as

$$R = R_{\mathcal{S}} + \Delta_{\mathcal{S}}, \tag{3.2}$$

with $R_{\mathcal{S}} \in \mathcal{S}$, thereby artificially introducing an approximation error $\Delta_{\mathcal{S}} := R - R_{\mathcal{S}} \in \mathbb{R}^{nb_y \times nb_u}$. A robust controller $Q : \mathbb{C} \mapsto \mathbb{C}^{nb_y \times nb_u}$ can then be designed using $P_{\mathcal{S}}(s) := R_{\mathcal{S}}g(s)$ and used to control the real plant $P(s) = Rg(s)$. Another possibility is to enforce the constraint $Q(s) \in \mathcal{S}\ \forall s \in \mathbb{C}$ during synthesis, which is analogous to the design of decentralised controllers [107, Ch. 12]. However, the constraint $Q(s) \in \mathcal{S}\ \forall s$ can lead to a non-convex optimisation problem [71], [123] and this method is not further considered.

In this chapter, it is assumed that R satisfies (3.1) for some $\epsilon > 0$, and the controller structure is fixed to the one from Section 1.6. The first aim is to analyse (1.3) when the structured approximation $R_{\mathcal{S}}$ (3.2) is substituted for R in (1.17), i.e. when the IMC filter is re-defined as

$$Q(s) := R_{\mathcal{S}}^{\dagger} q(s) \overset{(1.12)}{=} R_{\mathcal{S}}^{\dagger} T_{\mathrm{m}}(s)/g(s), \tag{3.3}$$

and embedded in the IMC structure from Fig. 3.1. By choosing the controller as in (3.3), the commutative algebra of the matrices in $\mathcal{S}$ allows the structure of only $R_{\mathcal{S}}$ to be constrained, which is then inherited by the controller $Q(s)$. One approximation that has been used in this setting is the *Frobenius norm approximation* (2.17) [28], [106], $R_{\mathcal{S}}^{\mathrm{F}} := \arg\min_{X \in \mathcal{S}} \|X - R\|_{\mathrm{F}}^2$, which was originally defined as a pre-conditioner for linear systems [23], but the resulting closed-loop properties have not been

analysed in detail and the choice of the Frobenius norm has not been justified. The second aim of this chapter is to characterise the Frobenius norm approximation.

The final aim of this chapter is to propose alternatives to the Frobenius approximation. It will be shown that an approximation based on the Frobenius norm can lead to unstable closed-loop dynamics even when a different structured approximation yields stable dynamics. For this reason, linear matrix inequalities (LMIs) or bilinear matrix inequalities (BMIs) are derived from the stability, performance and robustness properties of the system from Fig. 3.1 for a generic structured approximation $R_{\mathcal{S}}$. The LMIs or BMIs are then embedded in a semi-definite program (SDP) with the aim of finding a structured approximation that possibly performs better than the Frobenius norm approximation. The SDP can be formulated in the symmetric domain, where the optimisation variable is sparse, which makes this approach suitable for large-scale systems.

This chapter is organised as follows. In Sections 3.1–3.3, the stability, performance and robustness properties for the setting from Fig. 3.1 are analysed. In each section, the analysis is followed by deriving LMIs and BMIs that can be embedded in an SDP. In Section 3.4.1, the Frobenius norm approximation is revisited, before formulating SDPs for alternative approximations in Section 3.4.2. The chapter is concluded by applying the results to the ALBA synchrotron, for which it is shown that an alternative approximation yields better results than the Frobenius norm approximation.

3.1 Nominal Stability

To analyse the nominal stability, the structured approximation $R_{\mathcal{S}}$ is substituted for R in the model path of Fig. 3.1 and $Q(s)$ is formed using (3.3), while assuming that R and $g(s)$ accurately model the CD process. Compared to the IMC structure from Fig. 1.3b, the regularisation matrix Γ has been omitted, which would lead to highly non-convex constraints for the structured approximations from Section 3.4.2. However, analogous to the approach from Section 1.6, the regularisation matrix can be included *after* obtaining a structured approximation.

Using Fig. 3.1 and defining $P_{\mathcal{S}}(s) \coloneqq R_{\mathcal{S}} g(s)$, the transfer function from $d(s)$ to $u(s)$ is derived as

$$\begin{aligned} u(s) &= -(I + Q(s)(P(s) - P_{\mathcal{S}}(s)))^{-1} Q(s) d(s), \\ &= -Q(s)(I + \underbrace{(P(s) - P_{\mathcal{S}}(s))Q(s)}_{=\Delta_{\mathcal{S}} R_{\mathcal{S}}^{\dagger} T_{\mathrm{m}}(s)})^{-1} d(s), \\ &= -Q(s)(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1} d(s), \end{aligned} \tag{3.4}$$

where the push-through rule was used [130, Ch. 3.2] and the *error matrix* $\Phi_{\mathcal{S}} \in \mathbb{R}^{n_y \times n_y}$ defined as

$$\Phi_{\mathcal{S}} \coloneqq \Delta_{\mathcal{S}} R_{\mathcal{S}}^{\dagger}. \tag{3.5}$$

After substituting (3.4) in the CD system (1.3), the closed-loop transfer function from $d(s)$ to $y(s)$ is obtained as

$$\begin{aligned} y(s) &= \left(I - P(s)Q(s)(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}\right) d(s), \\ &= \left(I - T_{\mathrm{m}}(s)(R_{\mathcal{S}} + \Delta_{\mathcal{S}}) R_{\mathcal{S}}^{\dagger} (I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}\right) d(s), \\ &= \underbrace{\left(I - T_{\mathrm{m}}(s)(I + \Phi_{\mathcal{S}})(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}\right)}_{=:S(s)} d(s), \end{aligned} \tag{3.6}$$

where $R_{\mathcal{S}} R_{\mathcal{S}}^{\dagger} = I$ because $n_y \leq n_u$ and $S(s)$ is the *output sensitivity*. If $Q(s)$ is formed using (3.3) with $R_{\mathcal{S}} = R$ ($\Phi_{\mathcal{S}} = \Delta_{\mathcal{S}} = 0$), then the standard IMC closed-loop [107, Ch. 4.2] is recovered as

$$y(s) = (1 - T_{\mathrm{m}}(s)) d(s), \tag{3.7}$$

which is stable if $T_{\mathrm{m}}(s)$ is chosen as in (1.13). It holds that $y(0) = 0$ in both (3.6) and (3.7) if $T_{\mathrm{m}}(0) = 1$, from which it follows that the standard feedback equivalent of Fig. 3.1 implements n_y integrators for any $R_{\mathcal{S}}$. However, substituting the structured approximation $R_{\mathcal{S}}$ for the original R introduces an approximation error $\Delta_{\mathcal{S}}$, which in turn introduces the term $(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}$ in (3.6) that can be a source of instability. This is investigated in Theorem 3.1.

Theorem 3.1. *Suppose that $T_{\mathrm{m}}(s)$ is stable and has no poles on the imaginary axis. The system from Fig. 3.1 with $Q(s)$ defined as in (3.3) is (internally) stable iff the Nyquist plot of $\det(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s)) = \prod_i (1 + \phi_i T_{\mathrm{m}}(s))$, where ϕ_i are the eigenvalues of $\Phi_{\mathcal{S}}$, does not encircle the origin.*

Proof. No pole-zero cancellations with $\mathrm{Re}(s) > 0$ occur when forming the closed-loop transfer functions (3.4) and (3.6), which are products of stable transfer functions with $(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}$. According to the Nyquist stability criterion [130, Thm. 4.9], $(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}$ is stable iff the Nyquist plot of $\det(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))$ does not encircle the origin. □

Theorem 3.1 allows the stability of the system from Fig. 3.1 to be linked to the eigenvalues of the error matrix $\Phi_{\mathcal{S}}$. In Corollary 3.2, Theorem 3.1 is further simplified.

Corollary 3.2. *Suppose that $T_{\mathrm{m}}(s)$ is stable and has no poles on the imaginary axis. Suppose that $\Phi_{\mathcal{S}} = V \operatorname{diag}(\phi_1, \ldots, \phi_{n_y}) V^{-1}$, $\phi_i \in \mathbb{C}$ and $V \in \mathbb{C}^{n_y \times n_y}$. Then the system from Fig. 3.1 is stable iff for each $i = 1, \ldots, n_y$, none of the Nyquist plots of $1 + \phi_i T_{\mathrm{m}}(s)$ encircles the origin.*

Proof. The claim follows from diagonalising $(I + \Phi_{\mathcal{S}} T_{\mathrm{m}}(s))^{-1}$ and applying the Nyquist stability criterion. □

If all eigenvalues of $\Phi_{\mathcal{S}}$ were real, then according to Corollary 3.2, the range of ϕ_i that yields a stable system could be computed from the gain margin of $T_{\mathrm{m}}(s)$. However, since $\Phi_{\mathcal{S}} \neq \Phi_{\mathcal{S}}^*$ in general, it must be assumed that some ϕ_i are complex-valued. A more tractable but conservative condition than Corollary 3.2 is given in Corollary 3.3 [130, Thm. 4.11].

Corollary 3.3. *The system from Fig. 3.1 is stable if the spectral radius $\rho(\Phi_{\mathcal{S}}) := \max_i |\phi_i|$ satisfies $\rho(\Phi_{\mathcal{S}}) < 1$, where ϕ_i are the eigenvalues of $\Phi_{\mathcal{S}}$.*

Corollary 3.3 could also be obtained from applying standard techniques from robust control [130, Ch. 8]. Note that Theorem 3.1 and Corollaries 3.2 and 3.3 can also be formulated in the symmetric domain, i.e. by substituting $\hat{\Phi}_{\mathcal{S}}$ for $\Phi_{\mathcal{S}}$, where

$$\hat{\Phi}_{\mathcal{S}} := (T^* \otimes I_{b_y}) \Phi_{\mathcal{S}} (T \otimes I_{b_y}) = \hat{\Delta}_{\mathcal{S}} \hat{R}_{\mathcal{S}}^{\dagger}, \tag{3.8}$$

and

$$\hat{\Delta}_{\mathcal{S}} \coloneqq (T^* \otimes I_{b_y})\Delta_{\mathcal{S}}(T \otimes I_{b_u}). \tag{3.9}$$

3.1.1 Stability Conditions

With the controller $Q(s)$ being fixed as in (3.3), the nominal stability conditions depend on $T_{\mathrm{m}}(s)$ and the choice of $R_{\mathcal{S}}$. For a given $R_{\mathcal{S}}$, if the system is unstable, one possibility would be to substitute $\alpha T_{\mathrm{m}}(s)$, $0 < \alpha < 1$, for $T_{\mathrm{m}}(s)$, i.e. reducing the gain of $Q(s)$. However, according to (3.6) and (3.7), this would result in $y(0) \neq 0$ and therefore introduce an undesirable steady-state error.

Alternatively, the spectral radius of $\Phi_{\mathcal{S}}$ can be upper-bounded using matrix inequalities, which can subsequently be used to choose a structured approximation $R_{\mathcal{S}}$ that gives a favourable spectral radius of $\Phi_{\mathcal{S}}$. For that purpose, $\Phi_{\mathcal{S}}$ can be expanded as

$$\Phi_{\mathcal{S}} = \Delta_{\mathcal{S}} R_{\mathcal{S}}^{\dagger} = (R - R_{\mathcal{S}})R_{\mathcal{S}}^{\dagger} = RR_{\mathcal{S}}^{\dagger} - I, \tag{3.10}$$

where it is assumed that $R_{\mathcal{S}}R_{\mathcal{S}}^{\dagger} = I$ because $n_y \leq n_u$. Using (3.10), any upper bound on $\Phi_{\mathcal{S}}$ can be formulated in terms of $R_{\mathcal{S}}^{\dagger}$ or, after mapping (3.10) to symmetric domain, in terms of $\hat{R}_{\mathcal{S}}^{\dagger}$.

Upper bound via 2-norm. The spectral radius $\rho(\Phi_{\mathcal{S}})$ can be upper-bounded by [68, Thm. 5.6.14]

$$\rho(\Phi_{\mathcal{S}}) \leq \left\|\Phi_{\mathcal{S}}^k\right\|_2^{1/k}, \qquad k \in \mathbb{Z}_{++}, \tag{3.11}$$

with $\lim_{k\to\infty}\|\Phi_{\mathcal{S}}^k\|_2^{1/k} = \rho(\Phi_{\mathcal{S}})$. Although (3.11) uses the 2-norm, any other sub-multiplicative norm, such as the Frobenius norm, could also be used. By choosing $k = 1$ in (3.11) and substituting the right-hand side of (3.10), a sufficient condition for nominal stability is $\left\|RR_{\mathcal{S}}^{\dagger} - I\right\|_2 < 1$, which, using the Schur complement [19, Ch. 2], can be reformulated as the following linear matrix inequality (LMI):

$$\begin{bmatrix} I & RX - I \\ (RX - I)^* & I \end{bmatrix} \succ 0, \tag{NS1}$$

where $X \coloneqq R_{\mathcal{S}}^{\dagger} \in \mathcal{S}$ (cf. Lemma 2.2).

Because $\|A^2\|^{1/2} \leq \|A\|$, a possibly tighter bound can be obtained by choosing $k = 2$ in (3.11), which yields the sufficient stability condition $\|(RX - I)^2\|_2 < 1$, which can be reformulated as

$$\begin{bmatrix} I & (RX-I)^2 \\ ((RX-I)^2)^* & I \end{bmatrix} = \begin{bmatrix} I & RXRX - 2RX + I \\ (\dots)^* & I \end{bmatrix} \succ 0. \tag{NS2}$$

Constraint (NS2) is a *bilinear matrix inequality* (BMI) in X that can be solved using convexifying techniques. One possibility is to use the approach presented in [35] (Section 3.4.2), which finds a solution to the bilinear matrix inequality $F(X) \succ 0$ by solving a sequence of semidefinite programs. Note that (NS2) is never more conservative than (NS1).

Lyapunov certificate. The problem of finding X such that $\rho(RX - I) < 1$ can be recast using a discrete-time Lyapunov function approach [43, Ch. 1.4.4]. It holds that $\rho(RX - I) < 1$ iff there exists $P \in \mathbb{S}_{++}$, where $\mathbb{S}_{++}$ is the set of real symmetric positive definite matrices, such that $P - (RX - I)^* P (RX - I) \succ 0$ [43, Ch. 1.4.4]. Applying the Schur complement to the matrix inequality leads to the following constraint:

$$\begin{bmatrix} P^{-1} & RX - I \\ (\dots)^* & P \end{bmatrix} \succ 0, \tag{NS3}$$

which, after pre- and post-multiplication with $\operatorname{diag}(I, P^{-1})$, can be interpreted as a BMI in $Z \coloneqq XP^{-1}$ and P^{-1}.

In contrast to the constraints (NS1) and (NS2), constraint (NS3) introduces a dense matrix variable P and eventually becomes difficult to solve for large-scale matrices. Alternatively, one can fix P to have the same structural symmetry as X and reformulate (NS3) as

$$\begin{bmatrix} P_{\mathcal{S}}^{-1} & RZ_{\mathcal{S}} - P_{\mathcal{S}}^{-1} \\ (\dots)^* & P_{\mathcal{S}}^{-1} \end{bmatrix} \succ 0, \tag{NS4}$$

where $P_{\mathcal{S}} \in \mathcal{S}$ and $Z_{\mathcal{S}} \coloneqq XP_{\mathcal{S}}^{-1} \in \mathcal{S}$. Constraint (NS4) is an LMI in $Z_{\mathcal{S}}$ and $P_{\mathcal{S}}^{-1}$. Note that if an X is found that satisfies (NS1), then the same X satisfies (NS3) or (NS4) with $P = P_{\mathcal{S}} = I$, i.e. (NS3) and (NS4) are less conservative than (NS1).

3.2 Nominal Performance

In order to measure the impact of a structured approximation $R_{\mathcal{S}}$ on the performance, the output from the system that uses R, $y(s)$, can be compared with $y_{\mathcal{S}}(s)$, the output from the system that uses $R_{\mathcal{S}}$. Subtracting (3.6) from (3.7), the error $e(s) := y(s) - y_{\mathcal{S}}(s)$ is $e(s) = E(s)d(s)$ with

$$\begin{aligned} E(s) &:= T_{\mathrm{m}}(s)\big((I+\Phi_{\mathcal{S}})(I+T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{-1} - I\big), \\ &= T_{\mathrm{m}}(s)(1-T_{\mathrm{m}}(s))\Phi_{\mathcal{S}}(I+T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{-1}, \end{aligned} \tag{3.12}$$

where $E(0) = \lim_{\omega\to 0} E(\mathrm{j}\omega) = 0$. A structured approximation $R_{\mathcal{S}}$ that minimises $\|E(s)\|_2$ yields a similar closed-loop response to a system that uses R.

3.2.1 Performance Conditions

To obtain a more tractable form than (3.12), the term $\Phi_{\mathcal{S}}(I+T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{-1}$ is expanded using the Neumann series [68, Ch. 5.6, P26] as

$$\Phi_{\mathcal{S}}(I+T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{-1} = \Phi_{\mathcal{S}}\sum_{k=0}^{\infty}(-T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{k} = \Phi_{\mathcal{S}} - T_{\mathrm{m}}(s)\Phi_{\mathcal{S}}^2 + \mathcal{O}(\Phi_{\mathcal{S}}^3), \tag{3.13}$$

where it is assumed that $\rho(T_{\mathrm{m}}(\mathrm{j}\omega)\Phi_{\mathcal{S}}) < 1$. Combining (3.12) and (3.13), the magnitude of $E(s)$ can be upper-bounded by

$$\|E(s)\|_2 \le |T_{\mathrm{m}}(s)(1-T_{\mathrm{m}}(s))|\left\|\Phi_{\mathcal{S}} - T_{\mathrm{m}}(s)\Phi_{\mathcal{S}}^2\right\|_2 + \left\|\mathcal{O}(\Phi_{\mathcal{S}}^3)\right\|_2. \tag{3.14}$$

Ignoring higher-order terms in (3.14), $R_{\mathcal{S}}^{\dagger}$ can be chosen to minimise an upper bound $\sqrt{\alpha_\omega} \in \mathbb{R}_{++}$ on $\|\Phi_{\mathcal{S}} - T_{\mathrm{m}}(\mathrm{j}\omega)\Phi_{\mathcal{S}}^2\|_2 = \left\|RR_{\mathcal{S}}^{\dagger} - I - T_{\mathrm{m}}(\mathrm{j}\omega)(RR_{\mathcal{S}}^{\dagger} - I)^2\right\|_2$ at a particular frequency ω, which can be formulated using the Schur complement as

$$\begin{bmatrix} I & RX - I - T_{\mathrm{m}}(\mathrm{j}\omega)(RX-I)^2 \\ (\dots)^* & \alpha_\omega I \end{bmatrix} \ge 0, \tag{NP1}$$

where $X = R_{\mathcal{S}}^{\dagger}$. If (NP1) holds, then $\|E(\mathrm{j}\omega)\|_2 \le \sqrt{\alpha_\omega}|T_{\mathrm{m}}(\mathrm{j}\omega)(1-T_{\mathrm{m}}(\mathrm{j}\omega))|$. Note that (NP1) is a BMI, but in the limit $\omega \to \infty$ the following LMI is obtained:

$$\begin{bmatrix} I & RX - I \\ (\dots)^* & \alpha_\infty I \end{bmatrix} \ge 0, \tag{NP2}$$

which reduces to the nominal stability condition (NS1) for $\alpha_\omega = 1$.

3.3 Robust Stability with Additional Uncertainty

When the plant $P(s) = Rg(s)$ is approximated using $P_{\mathcal{S}}(s) = R_{\mathcal{S}}g(s)$, thereby artificially introducing the approximation error $\Delta_{\mathcal{S}}$ (3.2), it is assumed that $P(s)$ is known, i.e. $\bar{P}(s) = P(s)$ in Fig. 1.3b. In this section, it is assumed that $\bar{P}(s) \neq P(s)$ with $P(s)$ having an unknown component $\Theta : \mathbb{C} \mapsto \mathbb{C}^{n_y \times n_u}$, i.e.

$$P(s) := \bar{P}(s) + \Theta(s) = (R_{\mathcal{S}} + \Delta_{\mathcal{S}})g(s) + \Theta(s), \tag{3.15}$$

where $\bar{P}(s)$ is known and $R_{\mathcal{S}}$ is used to obtain the IMC filter $Q(s)$ (3.3).

It is assumed that a given $R_{\mathcal{S}}$ yields a stable system for $\Theta(s) = 0$ and that $\Theta(s)$ is stable. Then, for $\Theta(s) \neq 0$, the system from Fig. 3.2 is stable iff [130, Thm. 8.1]

$$\det(I - M(\mathrm{j}\omega)\Theta(\mathrm{j}\omega)) \neq 0 \quad \forall\omega, \tag{3.16}$$

where $M(s) := -Q(s)(I + T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{-1}$ is the transfer function from $u_\Theta(s)$ to $y_\Theta(s)$ (which are the auxiliary variables defined in Fig. 3.2) that equals the one from $d(s)$ to $u(s)$ (3.4). A sufficient condition for (3.16) is

$$\rho(M(\mathrm{j}\omega)\Theta(\mathrm{j}\omega)) < 1 \quad \forall\omega, \tag{3.17}$$

which, analogous to the *nominal* stability conditions from Section (3.1.1), can be upper-bounded using the 2-norm to obtain an upper bound on $\|\Theta(\mathrm{j}\omega)\|_2$ as

$$\|\Theta(\mathrm{j}\omega)\|_2 < \frac{1}{\|M(\mathrm{j}\omega)\|_2} = \frac{1}{|T_{\mathrm{m}}(\mathrm{j}\omega)/g(\mathrm{j}\omega)|\left\|R_{\mathcal{S}}^{\dagger}(I + T_{\mathrm{m}}(\mathrm{j}\omega)\Phi_{\mathcal{S}})^{-1}\right\|_2} \quad \forall\omega. \tag{3.18}$$

If, for a given uncertainty $\Theta(s)$, condition (3.18) is satisfied, then the system from Fig. 3.2 is stable. Moreover, a small $\left\|R_{\mathcal{S}}^{\dagger}(I + T_{\mathrm{m}}(\mathrm{j}\omega)\Phi_{\mathcal{S}})^{-1}\right\|_2$ allows for a large uncertainty.

3.3.1 Robust Stability Conditions

To obtain a robustness condition that can be embedded in an optimisation problem, the right-hand side term of (3.18) is expanded using a Neumann series [68, Ch. 5.6, P26] as

$$R_{\mathcal{S}}^{\dagger}(I + T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{-1} = R_{\mathcal{S}}^{\dagger}\sum_{k=0}^{\infty}(-T_{\mathrm{m}}(s)\Phi_{\mathcal{S}})^{k} \overset{(3.10)}{=} R_{\mathcal{S}}^{\dagger} + \mathcal{O}\Big(\big(R_{\mathcal{S}}^{\dagger}\big)^2\Big). \tag{3.19}$$

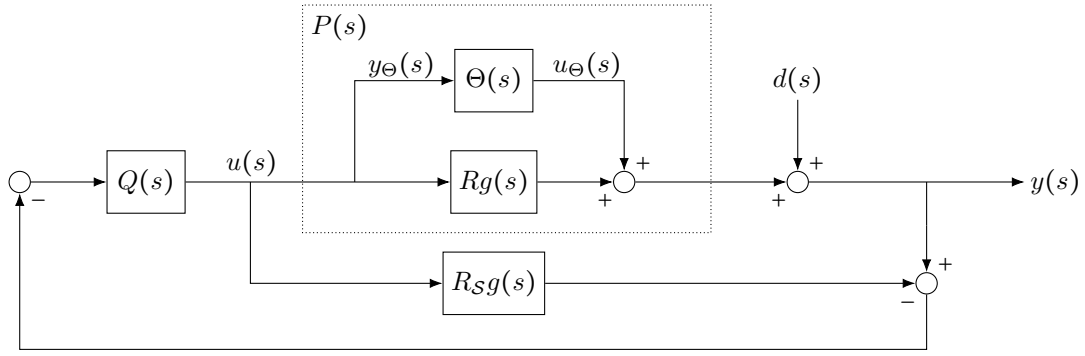


Figure 3.2: IMC structure with unknown uncertainty $\Theta(s)$ and known R and $g(s)$. The model path contains the structured approximation $R_{\mathcal{S}}$ that is used to form $Q(s)$.

An approximation $R_{\mathcal{S}}$ that yields a robust system therefore tends to make $\left\|R_{\mathcal{S}}^{\dagger}\right\|_2$ small, which is equivalent to decreasing the gain of the controller. A robust stability condition can be formulated as $\left\|R_{\mathcal{S}}^{\dagger}\right\|_2 \leq \sqrt{\beta}$ for some $\beta \in \mathbb{R}_{++}$, which can be reformulated using the Schur complement as

$$\begin{bmatrix} I & X \\ X^* & \beta I \end{bmatrix} \geq 0. \tag{RS}$$

3.4 Structured Approximations

3.4.1 Frobenius Norm Approximation

Approximations of the form

$$R_{\mathcal{S}}^{(\cdot)} = \arg\min_{X \in \mathcal{S}} \|X - R\|_{(\cdot)}^2, \tag{3.20}$$

where $(\cdot) = \{\mathrm{F}, 1, \infty\}$, have been proposed in several applications [27], [28], [31], [105], [106]. In [105], [106], the Frobenius norm is used and applied to a synchrotron orbit feedback control problem. In [28], the 1-norm and the Frobenius norm are applied to obtain structured approximations used in a robust model predictive control problem. However, in none of the applications has it been noted that considering

$$\|\Phi_{\mathcal{S}}\| \leq \|\Delta_{\mathcal{S}}\| \left\|R_{\mathcal{S}}^{\dagger}\right\| = \|R - R_{\mathcal{S}}\| \left\|R_{\mathcal{S}}^{\dagger}\right\|, \tag{3.21}$$

where $\|\cdot\|$ is an arbitrary sub-multiplicative norm, it becomes clear that an approximation of the form (3.20) minimises the upper bound $\|R_{\mathcal{S}} - R\|$ on the error matrix $\|\Phi_{\mathcal{S}}\|$. The term $\|R_{\mathcal{S}} - R\|$ can be interpreted as a first-order approximation of the nominal stability condition (3.11) and performance bounds (3.13), and the term $\left\|R_{\mathcal{S}}^{\dagger}\right\|$ as a first-order approximation of the robust stability condition (3.18).

Even though the matrix norms $(\cdot) = \{\mathrm{F}, 1, 2, \infty\}$ are *equivalent* [55, Ch. 2.3.2], it is unclear which choice of norm in (3.20) yields the best results. However, when the Frobenius norm is used the approximation error $\Delta_{\mathcal{S}}$ inherits a special structure that is characterised in Lemma 3.4:

Lemma 3.4. *If $R_{\mathcal{S}}$ is obtained from* (3.20) *with $(\cdot) = F$, then $\Delta_{\mathcal{S}}^{F} \coloneqq R - R_{\mathcal{S}}^{F} \in \mathcal{S}^{\perp}$ and $\hat{\Delta}_{\mathcal{S}}^{F} \coloneqq (T^* \otimes I_{b_y})\Delta_{\mathcal{S}}^{F}(T \otimes I_{b_u})$ is (block-)hollow.*

Proof. Because the Frobenius norm is invariant to pre- and post-multiplication with orthogonal matrices [55, Ch. 2.3.5], problem (3.20) can be reformulated for $(\cdot) = \mathrm{F}$ as $\hat{R}_{\mathcal{S}}^{\mathrm{F}} = \arg\min_{\hat{X} \in T^*\mathcal{S}T} \left\|T^*RT - \hat{X}\right\|_{\mathrm{F}}$, where $\hat{X}$ is diagonal. The minimum is attained when $\hat{X}$ equals the diagonal part of T^*RT and according to Def. 2.3, when $\hat{\Delta}_{\mathcal{S}} \in T^*\mathcal{S}^{\perp}T$ is hollow. The extension to block-structural symmetries is analogous. □

As a consequence of Lemma 3.4 and the block-diagonal property of $\hat{R}_{\mathcal{S}}$, it follows that $\hat{\Phi}_{\mathcal{S}} = \hat{\Delta}_{\mathcal{S}}\hat{R}_{\mathcal{S}}^{\dagger}$ is block-hollow too. Suppose that the *original* matrix R is mapped to the symmetric domain, giving $\hat{R} = (T^* \otimes I_{b_y})R(T \otimes I_{b_u})$, and then partitioned as

$$\hat{R} = \hat{R}_{\mathcal{S}}^{\mathrm{F}} + \hat{\Delta}_{\mathcal{S}}^{\mathrm{F}} = \begin{bmatrix} \hat{r}_{\mathcal{S},1} & & \\ & \ddots & \\ & & \hat{r}_{\mathcal{S},n} \end{bmatrix} + \begin{bmatrix} 0 & \hat{\delta}_{12} & \dots & \hat{\delta}_{1n} \\ \hat{\delta}_{21} & 0 & & \vdots \\ \vdots & & \ddots & \hat{\delta}_{(n\text{-}1)n} \\ \hat{\delta}_{n1} & \dots & \hat{\delta}_{n(n\text{-}1)} & 0 \end{bmatrix}, \tag{3.22}$$

where $\hat{r}_{\mathcal{S},i}, \hat{\delta}_{ij} \in \mathbb{C}^{b_y \times b_u}$, then the block-hollow property of $\hat{\Phi}_{\mathcal{S}}$ can be used to apply a Geršgorin-circle-type theorem for block-partitioned matrices [44] that relates (3.22) to the spectral radius $\rho(\hat{\Phi}_{\mathcal{S}})$, which, considering that (3.8) is a similarity transformation, equals $\rho(\Phi_{\mathcal{S}})$.

Theorem 3.5. *The spectral radius $\rho(\hat{\Phi}_{\mathcal{S}})$ satisfies $\rho(\hat{\Phi}_{\mathcal{S}}) \leq U$, where*

$$U \coloneqq \min\left\{ \max_{i=1,\dots,n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \left\| \hat{\delta}_{ij} \hat{r}_{\mathcal{S},i}^{\dagger} \right\|, \max_{j=1,\dots,n} \sum_{\substack{i=1 \\ i\neq j}}^{n} \left\| \hat{\delta}_{ij} \hat{r}_{\mathcal{S},i}^{\dagger} \right\| \right\},$$

for block-hollow $\hat{\Phi}_{\mathcal{S}} = \hat{\Delta}_{\mathcal{S}}^{F}(\hat{R}_{\mathcal{S}}^{F})^{\dagger} \in \mathbb{C}^{n_y \times n_y}$ partitioned as in (3.22) and any sub-multiplicative norm $\|\cdot\|$.

Proof. Each eigenvalue ϕ_k of $A \in \mathbb{C}^{n_y \times n_y}$ satisfies [44, Thm. 2]

$$\left(\left\| (A_{ii} - \phi_k I)^{-1} \right\| \right)^{-1} \leq \sum_{\substack{j=1 \\ j\neq i}}^{n} \|A_{ij}\|,$$

where A is partitioned into blocks $A_{ij} \in \mathbb{C}^{b_y \times b_y}$. If A is block-hollow, $A_{ii} = 0$ and $\left(\left\| (A_{ii} - \phi_k I)^{-1} \right\| \right)^{-1} = |\phi_k|$. It remains to substitute $\hat{\Phi}_{\mathcal{S},ij} = \hat{\delta}_{ij} \hat{r}_{\mathcal{S},i}^{\dagger}$ for A_{ij}. □

Note that the matrices $(\hat{R}_{\mathcal{S}}^{\mathrm{F}})^{\dagger}\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}$ and $\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}(\hat{R}_{\mathcal{S}}^{\mathrm{F}})^{\dagger}$ share the same non-zero eigenvalues [130, Ch. A.2.1], so that Theorem 3.5 can also be applied to $(\hat{R}_{\mathcal{S}}^{\mathrm{F}})^{\dagger}\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}$. The following Corollary 3.6 relates Theorem 3.5 to the nominal stability of the closed loop system through a block-diagonal dominance condition on the partitioning (3.22), and is in line with similar results on the decoupling of MIMO systems and decentralised control [130, Ch. 3.6.2]; [98, Ch. 4.6]; [107, Ch. 14.4.3].

Corollary 3.6. *The system from Fig. 3.1 is nominally stable if $\hat{R}_{\mathcal{S}}^{F} + \hat{\Delta}_{\mathcal{S}}^{F} = \hat{P}(0)$ is strictly column or row block diagonally dominant [44, Def. 1], e.g. if*

$$\left(\left\| \hat{r}_{\mathcal{S},i}^{\dagger} \right\| \right)^{-1} > \sum_{\substack{j=1 \\ j\neq i}}^{n} \left\| \hat{\delta}_{ij} \right\|, \tag{3.23}$$

for $i = 1, \dots, n$ and any sub-multiplicative norm $\|\cdot\|$.

Proof. By Corollary 3.3, the system from Fig. 3.1 is nominally stable if $\rho(\Phi_{\mathcal{S}}) = \rho(\hat{\Phi}_{\mathcal{S}}) < 1$. From Theorem 3.5, $\rho(\hat{\Phi}_{\mathcal{S}}) < 1$ if $\sum_{j\neq i} \left\| \hat{r}_{\mathcal{S},i}^{\dagger} \right\| \left\| \hat{\delta}_{ij} \right\| < 1$. Dividing by $\left\| \hat{r}_{\mathcal{S},i}^{\dagger} \right\|$ yields the row-wise block-diagonal dominance condition. The proof is analogous for column diagonal dominance. □

The Frobenius norm $\hat{R}_{\mathcal{S}}^{\mathrm{F}}$ (3.20) yields a block-hollow $\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}$, but it does not necessarily yield the best possible results in terms of stability of the closed loop system (3.6). To see this, suppose that the approximation is changed to $\hat{R}_{\mathcal{S}}^{\mathrm{F}}(1+\vartheta)$ with corresponding approximation error $\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}} - \vartheta\hat{R}_{\mathcal{S}}^{\mathrm{F}}$ for some scalar $\vartheta \in \mathbb{R}_+$. Since $\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}$ is hollow and $\hat{R}_{\mathcal{S}}^{\mathrm{F}} \perp \hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}$, it holds that $\left\|\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}} - \vartheta\hat{R}_{\mathcal{S}}^{\mathrm{F}}\right\|_{\mathrm{F}} \geq \left\|\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}\right\|_{\mathrm{F}}$, so $\left\|\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}} - \vartheta\hat{R}_{\mathcal{S}}^{\mathrm{F}}\right\|_{\mathrm{F}}$ is not optimal in the sense of (3.20). The spectral radius condition becomes

$$\rho\Big(\Big(\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}} - \vartheta\hat{R}_{\mathcal{S}}^{\mathrm{F}}\Big)\big(\hat{R}_{\mathcal{S}}^{\mathrm{F}}(1+\vartheta)\big)^{\dagger}\Big) = \rho\Big(\frac{1}{1+\vartheta}\Big(\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}(\hat{R}_{\mathcal{S}}^{\mathrm{F}})^{\dagger} - \vartheta I\Big)\Big) < 1.$$

If $\phi_1 \geq \cdots \geq \phi_{n_y}$ are the eigenvalues of $\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}(\hat{R}_{\mathcal{S}}^{\mathrm{F}})^{\dagger}$, then $(\phi_i - \vartheta)/(1+\vartheta)$ are the eigenvalues of $(\hat{\Delta}_{\mathcal{S}}^{\mathrm{F}}(\hat{R}_{\mathcal{S}}^{\mathrm{F}})^{\dagger} - \vartheta I)/(1+\vartheta)$. The spectral radius induced by the Frobenius norm approximation can therefore be reduced by choosing a sufficiently small ϑ satisfying $|\phi_1 - \vartheta| > |\phi_{n_y} - \vartheta|$. Such an ϑ always exists if $|\phi_1| \neq |\phi_{n_y}|$ and $|\phi_1| > 1$ or $|\phi_{n_y}| > 1$.

3.4.2 Semidefinite Programming Problems

In Section 3.4.1, it has been shown that the Frobenius norm approximation is possibly sub-optimal with respect to the spectral radius condition of Corollary 3.3. In fact, the Frobenius norm minimises one part of the upper bound (3.21) without considering $\left\|R_{\mathcal{S}}^{\dagger}\right\|$. However, for ill-conditioned systems $\left\|R_{\mathcal{S}}^{\dagger}\right\|$ might be arbitrarily large and therefore reduce the upper bound (3.18) on the admissible (unknown) uncertainty from the robust stability condition.

The stability, performance and robustness conditions from Sections 3.1-3.3, which are summarised in Table 3.1, can be used to formulate optimisation problems that lead to alternative approximations. For example, consider combining a convex objective function $f : \mathbb{R}_+ \times \mathbb{R}_+ \mapsto \mathbb{R}_+$ with the constraints (NS1), (NP2) and (RS) into the optimisation problem:

$$\underset{\substack{X \in \mathcal{S},\\ \alpha_\omega, \beta \in \mathbb{R}_{++}}}{\text{minimise}} \; f(\alpha_\omega, \beta) \quad \text{subject to (NS1), (NP2), (RS),} \tag{3.24}$$

which, if a solution exists, returns an approximation $R_{\mathcal{S}} = X^{\dagger}$ that yields a stable closed-loop satisfying performance and robustness bounds (NP2) and (RS), respectively.

Table 3.1: Overview of the programmatical constraints on nominal stability, nominal performance and robust stability. The second column refers to the equation label.

		Type	Optimisation variables	Matrix inequality
Stability	(NS1)	LMI	X	$\begin{bmatrix} I & RX-I \\ (\dots)^* & I \end{bmatrix} \succ 0$
	(NS2)	BMI	X	$\begin{bmatrix} I & (RX-I)^2 \\ (\dots)^* & I \end{bmatrix} \succ 0$
	(NS3)	BMI	X, P	$\begin{bmatrix} P^{-1} & RX-I \\ (\dots)^* & P \end{bmatrix} \succ 0$
	(NS4)	LMI	$Z_{\mathcal{S}}, P_{\mathcal{S}}^{-1}$	$\begin{bmatrix} P_{\mathcal{S}}^{-1} & RZ_{\mathcal{S}}-P_{\mathcal{S}}^{-1} \\ (\dots)^* & P_{\mathcal{S}}^{-1} \end{bmatrix} \succ 0$
Performance	(NP1)	BMI	X, α_ω	$\begin{bmatrix} I & RX-I-T_{\mathrm{m}}(\mathrm{j}\omega)(RX-I)^2 \\ (\dots)^* & \alpha_\omega I \end{bmatrix} \succeq 0$
	(NP2)	LMI	X, α_∞	$\begin{bmatrix} I & RX-I \\ (\dots)^* & \alpha_\infty I \end{bmatrix} \succeq 0$
Robustness	(RS)	LMI	X, β	$\begin{bmatrix} I & X \\ X^* & \beta I \end{bmatrix} \succeq 0$

If the objective function in (3.24) is convex and if the constraints are given by LMIs, then (3.24) is an SDP that can be solved using standard scientific software packages [8]. If some of the constraints in (3.24) are bilinear, a sub-optimal solution can be obtained by lower-bounding the BMIs using LMIs [158]. An approach that has been applied to the BMI from (NS3) is given in [35] and presented and applied to the remaining BMIs in the following paragraphs.

Table 3.2: Summary of the LMIs, $\mathcal{F}(X)+G(X,X_k) \preceq 0$, that result from upper-bounding the BMI constraints for nominal stability and performance (Appendix 3.A). In the last row, $\mathcal{R}(X)$ is used as a shorthand for $RX - I$.

(NS2)	$\begin{bmatrix} -I & R(2X + X_kRX_k - XRX_k - X_kRX) - I \\ (\dots)^* & -I \end{bmatrix} \preceq 0$
(NS3)	$\begin{bmatrix} P_k^{-1}(P - 2P_k)P_k^{-1} & -(RX - I) \\ (\dots)^* & -P \end{bmatrix} \preceq 0$
(NP1)	$\begin{bmatrix} -I & \mathcal{R}(X) + T_{\mathrm{m}}(\mathrm{j}\omega)((\mathcal{R}(X_k))^2 - \mathcal{R}(X_k)\mathcal{R}(X) - \mathcal{R}(X)\mathcal{R}(X_k)) \\ (\dots)^* & -\alpha_\omega^2 I \end{bmatrix} \preceq 0$

Convexifying algorithm

Suppose that the optimisation problem is

$$\begin{array}{ll} \underset{X \in \mathbb{R}^{n_u \times n_y}}{\text{minimise}} & f(X) \\ \text{subject to} & X \in \Omega_0, \end{array} \tag{3.25}$$

where $f : \mathbb{C}^{n_u \times n_y} \mapsto \mathbb{R}$ is a convex and first-order differentiable function bounded from below and the constraint set Ω_0 is given by

$$\Omega_0 := \{X \in \mathcal{S} \mid \mathcal{F}(X) \preceq 0, \quad F_i(X) \preceq 0, \quad i = 1, \dots, N\}, \tag{3.26}$$

where $\mathcal{F}(X) \preceq 0$ is a BMI and $F_i(X) \preceq 0$ are LMIs. The following Def. 3.7 introduces the *convexifying potential matrix functional* [35] that is used to lower-bound $\mathcal{F}(X)$.

Definition 3.7 (Convexifying potential matrix functional [35])**.** Given a BMI $\mathcal{F}(X) \preceq 0$, the *convexifying potential matrix functional* is a matrix function $G(X,Y)$ that satisfies (i) $G(X,Y) \succeq 0$, (ii) $G(X,X) = 0$, and (iii) $\nabla G(X,X) = 0 \ \forall\, X, Y$ and is such that $\mathcal{F}(X) + G(X,Y) \preceq 0$ is an LMI in X.

For each of the BMI constraints from Table 3.1, convexifying potential matrix functionals are derived in Appendix 3.A and the resulting LMIs, $\mathcal{F}(X)+G(X,Y) \preceq 0$, are listed in Table 3.2. Note that if the LMI $\mathcal{F}(X)+G(X,Y) \preceq 0$ is satisfied for some X, then according to Def. 3.7, the BMI $\mathcal{F}(X) \preceq -G(X,Y) \preceq 0$ is also satisfied.

After convexifying the BMIs, the LMIs from Table 3.1 are embedded in the iterative procedure from Algorithm 3.1 [35, Alg. 1]. Given a feasible $X_0 \in \Omega_0$, Algorithm 3.1 repeatedly solves an SDP on Line 4 to produce iterates $X_{k+1} \in \Omega_k$, where

$$\Omega_k := \{X \in \mathcal{S} \mid \mathcal{F}(X) + G(X, X_k) \preceq 0,\ F_i(X) \preceq 0,\ i = 1, \dots, N\} \tag{3.27}$$

is updated at every iteration on Line 3, and hence guarantees that the BMI is satisfied. The algorithm terminates once $\|X_{k+1} - X_k\| < \epsilon$, where $\epsilon > 0$ is fixed. If $\mathcal{F}(X)$ is a concave matrix function, Algorithm 3.1 converges to a local optimum of (3.25) [35, Thm. 5], which, as shown in Appendix 3.A, is only the case for the BMI constraint (NS3).

Algorithm 3.1 Convexifying algorithm [35] applied to problem (3.25).

Input: $X_0 \in \Omega_0$
Output: $X^\star \in \mathcal{S}$

1: $k = 0$
2: **while** $\|X_{k+1} - X_k\| \geq \epsilon$ **do**
3: $\quad \Omega_k = \{X \in \mathcal{S} \mid \mathcal{F}(X) + G(X, X_k) \preceq 0, \quad F_i(X) \preceq 0, \quad i = 1, \dots, N\}$
4: $\quad X_{k+1} = \arg\min_{X \in \Omega_k} \ f(X)$
5: $\quad k \leftarrow k + 1$
6: **end while**

If the Frobenius norm approximation yields a stable closed-loop, it can be used to initialise Algorithm 3.1 as $X_0 = (R_{\mathcal{S}}^{\mathrm{F}})^\dagger$, but when $R_{\mathcal{S}}^{\mathrm{F}}$ yields an unstable closed-loop, an alternative solution is to obtain X_0 from the solution to

$$\begin{aligned} \underset{\substack{X \in \mathcal{S}, \\ P \in \mathbb{S}_{++}, \\ \sigma \in \mathbb{R}_{++}}}{\text{minimise}} \quad & \sigma \\ \text{subject to} \quad & \begin{bmatrix} \sigma P^{-1} & RX - I \\ (\dots)^* & P \end{bmatrix} \succ 0, \end{aligned} \tag{3.28}$$

which corresponds to the Lyapunov certificate (NS3) with an additional variable $\sigma \in \mathbb{R}_{++}$ that is an upper-bound to the spectral radius $\rho(RX - I)$ [43, Ch. 1.4.4]. Problem (3.28) includes a BMI that can be convexified using the procedure from Appendix 3.A:

$$\begin{aligned} \underset{\substack{X \in \mathcal{S}, \\ P \in \mathbb{S}_{++}, \\ \sigma \in \mathbb{R}_{++}}}{\text{minimise}} \quad & \sigma \\ \text{subject to} \quad & \begin{bmatrix} \sigma_k(\sigma_k P - 2\sigma P_k) & -P_k(RX - I) \\ (\dots)^* & -P \end{bmatrix} \preceq 0. \end{aligned} \tag{3.29}$$

When Algorithm 3.1 is applied to (3.29), it must be initialised using $X_0 \in \mathcal{S}$, $P_0 \in \mathbb{S}_{++}$ and $\sigma_0 \in \mathbb{R}_{++}$ that satisfy

$$\begin{bmatrix} \sigma_0 P_0^{-1} & RX_0 - I \\ (\dots)^* & P_0 \end{bmatrix} \succ 0, \tag{3.30}$$

which can always be satisfied by choosing P_0 and σ_0 large. If on termination of Algorithm 3.1 applied to (3.29), a solution with $\sigma > 1$ is obtained, meaning that the approximation yields an unstable closed-loop, Algorithm 3.1 either converged to a local optimum or the underlying system does not allow for a structured approximation that yields a stable closed-loop with the present control approach. In the former case, Algorithm 3.1 could be repeated using a different initialisation for P_0 and σ_0.

3.5 Numerical Examples

Consider a circulant system of order $n = 3$ with $R \in \mathbb{R}^{3\times3}$ given by

$$R = F_3\Big(\hat{R}_{\mathcal{C}}^{\mathrm{F}} + \hat{\Delta}_{\mathcal{C}}^{\mathrm{F}}\Big)F_3^* = F_3\left(\begin{bmatrix} \hat{r}_1 & 0 & 0 \\ 0 & \hat{r}_2 & 0 \\ 0 & 0 & \hat{r}_2^* \end{bmatrix} + \begin{bmatrix} 0 & \hat{\delta}_1 & \hat{\delta}_1^* \\ \hat{\delta}_2 & 0 & 0 \\ \hat{\delta}_2^* & 0 & 0 \end{bmatrix}\right)F_3^*, \tag{3.31}$$

where $\hat{\delta}_i \in \mathbb{C}$, $F_3 \in \mathbb{C}^{3\times3}$ is the discrete Fourier transformation matrix (2.5) and $\mathcal{C}$ refers to the circulant symmetry. The eigenvalues of $\hat{\Phi}_{\mathcal{C}}^{\mathrm{F}} \coloneqq \hat{\Delta}_{\mathcal{C}}^{\mathrm{F}}(\hat{R}_{\mathcal{C}}^{\mathrm{F}})^{-1}$ are

$$\phi_1^{\mathrm{F}} = 0, \qquad \phi_{2,3}^{\mathrm{F}} = \pm\sqrt{\frac{\hat{\delta}_1\hat{\delta}_2}{\hat{r}_1\hat{r}_2} + \left(\frac{\hat{\delta}_1\hat{\delta}_2}{\hat{r}_1\hat{r}_2}\right)^*}. \tag{3.32}$$

For the remainder of Section 3.5, it is assumed that $T_{\mathrm{m}}(s) = 1/(s+1)$, so that according to Corollary 3.2, the resulting closed-loop system is stable if $|\phi_i| < 1\ \forall i$, and unstable if $\mathrm{Re}(\phi_i) \leq -1$ for at least one i.

Unstable Frobenius norm approximation. Choosing the values in (3.31) as

$$\hat{R} = \hat{R}_{\mathcal{C}}^{\mathrm{F}} + \hat{\Delta}_{\mathcal{C}}^{\mathrm{F}} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & -2+\mathrm{j} & 0 \\ 0 & 0 & -2-\mathrm{j} \end{bmatrix} + \begin{bmatrix} 0 & 1+\mathrm{j}0.2 & 1-\mathrm{j}0.2 \\ -4-\mathrm{j}4 & 0 & 0 \\ -4+\mathrm{j}4 & 0 & 0 \end{bmatrix}, \tag{3.33}$$

results in $\phi_3^{\mathrm{F}} = -2.5 < -1$. Note that $\hat{R}$ is *not* diagonally dominant and Corollary 3.6 is therefore not satisfied. With the aim of obtaining a stable closed-loop, a structured approximation is obtained from

$$\underset{\substack{X\in\mathcal{C},\\ \alpha_\infty,\beta\in\mathbb{R}_{++}}}{\text{minimise}}\ (\alpha_\infty+\beta) \quad \text{subject to (NS3), (NP2), (RS),} \tag{3.34}$$

where constraint (NS3) is a BMI. Problem (3.34) is therefore solved using Algorithm 3.1, which is in turn initialised using (3.29) that results after 6 iterations in an approximation $\hat{R}_{\mathcal{C}} = \mathrm{diag}(-3.9, -1.1+\mathrm{j}0.5, -1.1-\mathrm{j}0.5)$. The spectral radius is $\rho(\hat{\Phi}_{\mathcal{C}}) = 0.87$ and the system therefore stable. Using the corresponding Lyapunov certificate, one could proceed with solving (3.24) to improve performance and robustness properties of the approximation.

Stable Frobenius norm approximation. Consider again system (3.33), but divide $\hat{\Delta}_{\mathcal{C}}^{\mathrm{F}}$ by 10, so that

$$\hat{R} = \underbrace{\begin{bmatrix} 0.1 & 0 & 0 \\ 0 & -2+\mathrm{j} & 0 \\ 0 & 0 & -2-\mathrm{j} \end{bmatrix}}_{=\hat{R}_{\mathcal{C}}^{\mathrm{F}}} + \underbrace{\begin{bmatrix} 0 & 0.1+\mathrm{j}0.02 & 0.1-\mathrm{j}0.02 \\ -0.4-\mathrm{j}0.4 & 0 & 0 \\ -0.4+\mathrm{j}0.4 & 0 & 0 \end{bmatrix}}_{=\hat{\Delta}_{\mathcal{C}}^{\mathrm{F}}}. \tag{3.35}$$

Even though $\hat{R}$ is not diagonally dominant, $\rho(\hat{\Phi}_{\mathcal{C}}^{\mathrm{F}}) = 0.25$ and the Frobenius norm approximation is stable. Fig. 3.3 compares the resulting output sensitivity for $Q(s)$ (3.3) formed using $R_{\mathcal{C}}^{\mathrm{F}}$ (—■—) with the sensitivity for $Q(s)$ (3.3) formed using the original R (—●—), where it can be seen that disturbances are amplified by 12.5 dB at 100 Hz. To reduce the sensitivity peak, problem (3.24) is initialised using the Frobenius norm approximation and solved using Algorithm 3.1. After 15 iterations with constraint (NP1) evaluated at 100 Hz, Algorithm 3.1 produces

$$\hat{R} = \underbrace{\begin{bmatrix} 6.8 & 0 & 0 \\ 0 & -1+\mathrm{j}0.5 & 0 \\ 0 & 0 & -1-\mathrm{j}0.5 \end{bmatrix}}_{=:\hat{R}_{\mathcal{C}}^{\mathrm{BMI}}} + \underbrace{\begin{bmatrix} -6.7 & 0.1+\mathrm{j}0.02 & 0.1-\mathrm{j}0.02 \\ -0.4-\mathrm{j}0.4 & -1+\mathrm{j}0.5 & 0 \\ -0.4+\mathrm{j}0.4 & 0 & -1-\mathrm{j}0.5 \end{bmatrix}}_{=:\hat{\Delta}_{\mathcal{C}}^{\mathrm{BMI}}}. \tag{3.36}$$

The spectral radius is $\rho(\hat{\Phi}_{\mathcal{C}}^{\mathrm{BMI}}) = 0.99$ and the structured approximation is therefore nominally stable. The resulting output sensitivity is shown in Fig. 3.3 (—▲—),

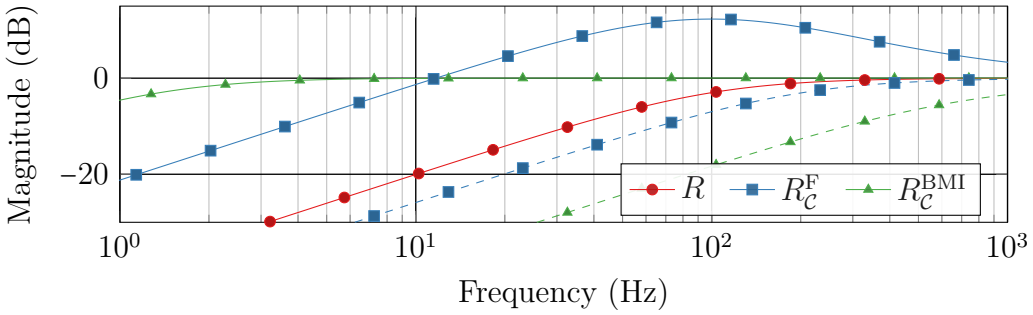


Figure 3.3: Maximum and minimum (dashed) output sensitivity gains for the stable example from Section 3.5.

where it can be seen that the sensitivity peak has been reduced. Note that this does not happen at the expense of robustness, because $1/\left\|\left(\hat{R}_{\mathcal{C}}^{\mathrm{BMI}}\right)^{-1}\right\|_2 = 1.13$ and $1/\left\|(\hat{R}_{\mathcal{C}}^{\mathrm{F}})^{-1}\right\|_2 = 0.1$.

3.6 Case Study: ALBA Synchrotron

The ALBA synchrotron is a third-generation light source located in Barcelona, Spain, that accelerates electrons to 3 GeV [42]. The electrons circulate around a 270 m circumference storage ring that accommodates 8 beamlines. The storage ring is divided into $n = 4$ superperiods and each of the four storage ring sections has $b_y = 22$ BPMs and $b_u = 22$ corrector magnets, which amounts to a total of $n_y = 88$ BPMs and $n_u = 88$ correctors. At ALBA, the sample rate of the FOFB is 10 kHz and based on the standard feedback structure and a PI controller [112], but for the following developments, it is assumed that the approach from Section 1.6 is used with the Diamond ORM replaced by the (vertical) ALBA ORM. The plant model is assumed to be $g(s) = a_y/(s + a_y)$ with $a_y = 2\pi \times 700\,\mathrm{rad\,s^{-1}}$ and for the lamba controller approach from Section 1.6.2, the complementary closed-loop sensitivity is chosen as $T_{\mathrm{m}}(s) = \lambda/(s + \lambda)$ with $\lambda = 2\pi \times 200\,\mathrm{rad\,s^{-1}}$.

As shown in Section 2.5.1, the ALBA ORM $R \in \mathbb{R}^{n_y \times n_u}$ inherits a block-circulant ($\mathcal{BC}$) symmetry and an approximate block-centrosymmetry ($\mathcal{CS}$) from the storage ring structure, but both symmetries are approximate in the sense of (3.1). In the following sections, the Frobenius norm approximation is revisited and compared

Table 3.3: Comparison of spectral radius $\rho(\Phi_{\mathcal{S}}^{\mathrm{F}})$ of the Frobenius norm approximation for the ALBA synchrotron with the approximation error $\Delta_{\mathcal{S}}^{\mathrm{F}} = R - R_{\mathcal{S}}^{\mathrm{F}}$ from Table 2.2.

$\mathcal{S}$	$\rho(\Phi_{\mathcal{S}}^{\mathrm{F}})$	2-norm (%)	Mean (%)	Max-norm (%)
$\mathcal{BC}$	1.3×10^{-6}	2.007	1.957	6.269
$\mathcal{CS}$	1.5×10^{-1}	4.234	2.849	14.998
$\mathcal{BC} \cap \mathcal{CS}$	1.5×10^{-1}	4.339	3.309	15.526

with alternative approximations obtained using the optimisation program approach from Section 3.4.2.

3.6.1 Structured Approximations

The Frobenius norm approximation for the ALBA synchrotron is computed according to Section 2.5.1. Here, the spectral radii of $\Phi_{\mathcal{S}}^{\mathrm{F}}$ are computed for $\mathcal{S} \in \{\mathcal{BC}, \mathcal{CS}, \mathcal{BC} \cap \mathcal{CS}\}$ and compared with the approximation error in Table 3.3. For all symmetries, the approximations satisfy $\rho(\Phi_{\mathcal{S}}^{\mathrm{F}}) < 1$ and are therefore closed-loop stable.

For the remainder of this chapter, the analysis is focused onto the combined $\mathcal{BC} \cap \mathcal{CS}$ symmetry, which, according to Table 3.3, results in the largest approximation error and the largest spectral radius. As an alternative to the Frobenius norm approximation $R_{\mathcal{BC} \cap \mathcal{CS}}^{\mathrm{F}}$, two approximations are derived from the SDPs from Section 3.4.2. The ORM of the ALBA synchrotron has $n_y \times n_u = 7744$ non-zero elements, which results in large SDPs. In practice, the following problems are therefore formulated in symmetric domain, where $\hat{R}_{\mathcal{BC} \cap \mathcal{CS}}^{(\cdot)}$ is sparse and has $b_y \times b_u \times (2n - 3)/2 = 1210$ non-zero elements that are purely real or purely imaginary. The SDPs are solved on a desktop computer (Intel i7-7700 CPU @ 3.1 GHz, 8 GB) using MOSEK [8].

Approximation using LMIs. The LMI constraints (NS1), (NP2) and (RS) from Table 3.1 are combined into the following SDP:

$$
\begin{aligned}
R_{\mathcal{BC} \cap \mathcal{CS}}^{\mathrm{LMI}} \coloneqq \underset{\substack{X \in \mathcal{BC} \cap \mathcal{CS}, \\ \alpha_\infty, \beta \in \mathbb{R}_{++}}}{\arg\min} \quad & \alpha_\infty / \bar{\alpha}_\infty + \beta / \bar{\beta} \\
\text{subject to} \quad & \begin{bmatrix} I & RX - I \\ (RX - I)^* & \alpha_\infty I \end{bmatrix} \succeq 0, \\
& \alpha_\infty < 1, \\
& \begin{bmatrix} I & X \\ (RX - I)^* & \beta I \end{bmatrix} \succeq 0,
\end{aligned}
\tag{3.37}
$$

where the constraint (NS1) is enforced through (NP2) with $\alpha < 1$. The objective function, $f(\alpha_\infty, \beta) = \alpha_\infty/\bar{\alpha}_\infty + \beta/\bar{\beta}$, trades off performance versus robustness, and the normalising weights $\bar{\alpha}_\infty$ and $\bar{\beta}$ are chosen as $\bar{\alpha}_\infty := \|\Phi^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}\|_2^2$ and $\bar{\beta} := \|(R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}})^\dagger\|_2^2$, where $\Phi^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$ and $R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$ stem from the Frobenius norm approximation. When formulated in the symmetric domain, the SDP (3.37) is solved within less than a minute.

Approximation using BMIs. With the aim of improving the LMI approximation, problem (3.37) is extended with the BMI constraint (NP1) to obtain the following non-convex optimisation problem[1]

$$
\begin{aligned}
R^{\mathrm{BMI}}_{\mathcal{BC}\cap\mathcal{CS}} := \quad & \underset{\substack{X\in\mathcal{BC}\cap\mathcal{CS},\\ \alpha_\infty,\alpha_\omega,\beta\in\mathbb{R}_{++}}}{\arg\min} \quad \alpha_\infty/\bar{\alpha}_\infty + \beta/\bar{\beta} + \alpha_\omega/\bar{\alpha}_\omega \\
& \text{subject to} \qquad \begin{bmatrix} I & RX-I \\ (RX-I)^* & \alpha_\infty I \end{bmatrix} \succeq 0, \\
& \qquad\qquad\qquad\qquad\quad \alpha_\infty < 1, \\
& \qquad\qquad\qquad\qquad\quad \begin{bmatrix} I & X \\ (RX-I)^* & \beta I \end{bmatrix} \succeq 0, \\
& \qquad\qquad \begin{bmatrix} I & RX-I-T_{\mathrm{m}}(\mathrm{j}\omega)(RX-I)^2 \\ (\dots)^* & \alpha_\omega I \end{bmatrix} \succeq 0,
\end{aligned}
\tag{3.38}
$$

where $\bar{\alpha}_\omega := \|\Phi^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}} - T_{\mathrm{m}}(\mathrm{j}\omega)(\Phi^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}})^2\|_2^2$ and constraint (NP1) is evaluated at $\omega = 2\pi \times 100\,\mathrm{rad\,s^{-1}}$, which will be justified in the following sections. After convexifying the last constraint of (3.38) using Table 3.2, problem (3.38) can be solved using Algorithm 3.1. For the approximation obtained from (3.38), Algorithm 3.1 was initialised using $R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$ and executed 60 iterations before reaching the stopping criteria of Algorithm 3.1 with $\epsilon = 10^{-3}$, which required 2 h of computing time.

3.6.2 Nominal Stability

Because the optimisation programs (3.37) and (3.38) enforce closed-loop stability and are solved with no constraint violations, the two approximations obtained from (3.37) and (3.38) both yield stable closed loops. However, in general there is no guarantee that feasible solutions to (3.37) and (3.38) exist, and because the constraints from Table 3.1 are only sufficient but not necessary, infeasibility of (3.37) and (3.38) does *not* prove that no stabilising structured approximation exists.

[1] $R^{\mathrm{BMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ is in fact obtained from the convexified version of (3.38) and not from (3.38) itself.

Table 3.4: Comparison of spectral radii $\rho(\Phi_{\mathcal{S}})$ and 2-norms of $\Phi_{\mathcal{S}}$, $R_{\mathcal{S}}^{\dagger}$ and $\Delta_{\mathcal{S}}$ resulting from different approximations for the $\mathcal{S} = \mathcal{BC} \cap \mathcal{CS}$ symmetry at the ALBA synchrotron.

Approximation	$\rho(\Phi^{(\cdot)}_{\mathcal{BC}\cap\mathcal{CS}})$	$\left\|\Delta^{(\cdot)}_{\mathcal{BC}\cap\mathcal{CS}}\right\|_2$	$\left\|\Phi^{(\cdot)}_{\mathcal{BC}\cap\mathcal{CS}}\right\|_2$	$\left\|(R^{(\cdot)}_{\mathcal{BC}\cap\mathcal{CS}})^{\dagger}\right\|_2$
Frobenius	0.1503	1.1414	0.4736	26.8679
LMI problem	0.5259	9.4590	0.5838	18.9985
BMI problem	0.4156	1.4024	0.4315	17.8677

The spectral radii and additional metrics resulting from (3.37) and (3.38) are compared with the Frobenius norm approximation $R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$ in Table 3.4, where it can be seen that the spectral radii $\rho(\Phi^{\mathrm{LMI}}_{\mathcal{BC}\cap\mathcal{CS}})$ and $\rho(\Phi^{\mathrm{BMI}}_{\mathcal{BC}\cap\mathcal{CS}})$ are smaller than 1, but over 2 times larger than $\rho(\Phi^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}})$. However, the spectral radius is *not* a good measure for robustness, as will be seen in Section 3.6.4.

3.6.3 Nominal Performance

The maximum and minimum (dashed) output sensitivity gains (3.6) of the system from Fig. 3.1 are shown in Fig. 3.4 for $Q(s)$ formed using the original R (—●—) and the structured approximations from Section 3.6.1. For the original R, the minimum and the maximum sensitivity gains coincide, which is a consequence of the approach from Section 1.6.2.

Compared to the original system, the closed-loop bandwidth of the systems that are controlled using structured approximations is lowered by 100 Hz. Measured by the maximum sensitivity gain, the Frobenius norm approximation $R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$ (—■—) performs best and produces a low-frequency attenuation that is roughly 3 dB higher than the low-frequency attenuation of the original system. The approximation $R^{\mathrm{LMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ (—▲—) performs worse than $R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$ and produces a worst-case low-frequency disturbance attenuation that is roughly 5 dB higher than the low-frequency attenuation of the original system.

To reduce the difference in performance associated with $R^{\mathrm{LMI}}_{\mathcal{BC}\cap\mathcal{CS}}$, the problem (3.37) is extended with the BMI constraint (NP1) evaluated at 100 Hz to obtain $R^{\mathrm{BMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ (—◆—). In Fig. 3.1, it can be seen that the addition of the BMI constraint lowers the maximum sensitivity gain by roughly 1 dB at 100 Hz. Additional BMI

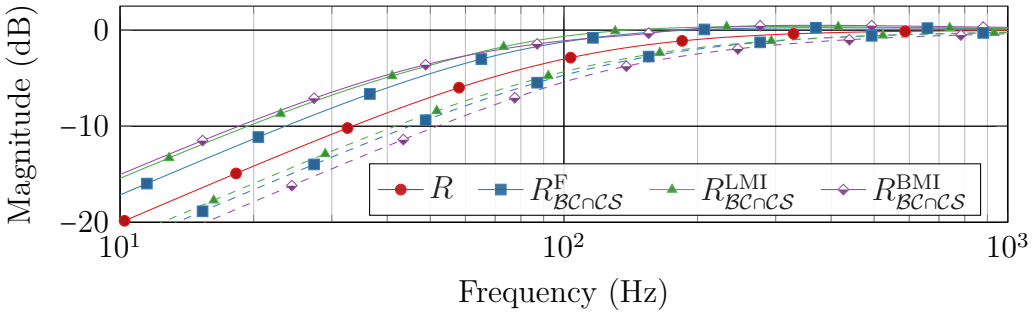


Figure 3.4: Maximum and minimum (dashed) sensitivity gains for the original system and the $\mathcal{BC}\cap\mathcal{CS}$ approximations of the ALBA synchrotron.

constraints at different frequencies could be integrated in (3.38) to further reduce the performance difference.

All structured approximations also affect the minimum output sensitivity gain, and according to Fig. 3.4, perform better for certain disturbance directions. However, the strong directionality of the system also affects the disturbance spectrum, which is more pronounced for directions associated with large singular values of R, and a detailed performance analysis therefore requires to consider the disturbance spectrum at the ALBA synchrotron.

3.6.4 Robust Stability

The robustness of the system is measured by the frequency-dependent upper bound (3.18) on the unknown additional uncertainty $\Theta(\mathrm{j}\omega)$, which is shown in Fig. 3.5 for the system that uses the original R (—●—) and the different structured approximations from Section 3.6.1.

At low frequencies ($\omega \le 2\pi \times 20\,\mathrm{rad\,s^{-1}}$), the norm of the admissible unknown uncertainty is at least $-30\,\mathrm{dB} \approx 0.03$ before the closed-loop system might become unstable, and this upper bound is of similar magnitude for all systems from Fig. 3.5, including the one that uses the original R. In the limit $\omega \to 0$, the right-hand side of the upper bound (3.18) becomes $1/\|R^\dagger\|_2$ for the system that uses the original R and $1/\left\|(R_{\mathcal{S}}^{(\cdot)})^\dagger(I+\Phi_{\mathcal{S}}^{(\cdot)})^{-1}\right\|_2$ for the structured approximation. The upper bound is dominated by $1/\|R^\dagger\|_2$ or $1/\left\|(R_{\mathcal{S}}^{(\cdot)})^\dagger\right\|_2$, which reflects the low-magnitude singular values of R or $R_{\mathcal{S}}^{(\cdot)}$ for $\mathcal{S} = \mathcal{BC}\cap\mathcal{CS}$, respectively.

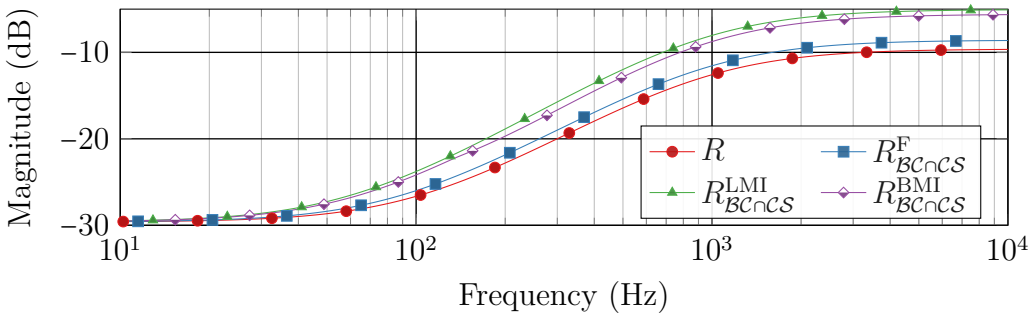


Figure 3.5: Upper bound on unkown uncertainty $\|\Theta(\mathrm{j}\omega)\|_2$ for the original system and the $\mathcal{BC}\cap\mathcal{CS}$ approximations of the ALBA synchrotron.

At high frequencies ($\omega \geq 2\pi \times 1\,\mathrm{krad\,s^{-1}}$), the structured approximations $R^{\mathrm{LMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ (—▲—) and $R^{\mathrm{BMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ (—◆—) yield significantly more robust systems than the systems that use R and $R^{\mathrm{F}}_{\mathcal{BC}\cap\mathcal{CS}}$. At 1 kHz, the admissible uncertainty is at least 6 dB $\approx$ 2 greater for the $R^{\mathrm{LMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ and $R^{\mathrm{BMI}}_{\mathcal{BC}\cap\mathcal{CS}}$ approximations, which suggests that the performance loss from Fig. 3.4 is traded against the gain in robustness from Fig. 3.5.

3.7 Conclusion

In this chapter, the IMC approach from Section 1.6.2 was applied to CD systems with approximate structural symmetries. After fixing the controller structure and substituting a generic approximation for the original plant model, the nominal stability, performance and robustness properties were analysed. Based on this analysis, conditions on stability, performance and robustness were derived that can be embedded in an SDP with the aim of finding an approximation that has exact structural symmetries.

In contrast to SDP-based approximations, the Frobenius norm approximation benefits from a closed-form solution and a block-hollow structure of the resulting approximation error. Based on the properties of the approximation error, a simple block-diagonal dominance condition was derived to verify whether a CD system is amenable to a Frobenius norm approximation. In general, row or column block-diagonal dominance of the steady-state gain matrix is sufficient to ensure closed-loop stability of the symmetric approximation.

It was also shown that the Frobenius norm approximation can be sub-optimal in the sense that it can yield an unstable system or a system with poor performance. In this case, the SDP-based approach can be used to find structured approximations with improved performance and robustness properties. For the case that the Frobenius approximation yields an unstable closed loop, it was shown how to define an optimisation problem that is able to find a stabilising approximation (if it exists). If the Frobenius norm approximation yields a stable closed loop, it can be used to initialise alternative optimisation problems that can lead to approximations with better robustness and performance properties. These optimisation problems can be solved in the symmetric domain where the matrices are sparse, which allows for large-scale systems with large optimisation problems to be investigated that would otherwise be difficult to solve if all matrices were dense.

The asymmetry of the steady-state gain matrix of a CD system has been investigated, but a possible asymmetry of the actuator dynamics has been ignored. For systems with asymmetric actuator dynamics, the nominal stability condition, which is based on evaluating the spectral radius of a static closed-loop matrix, would need to be evaluated on a frequency-by-frequency basis. It is unclear whether the block-diagonal dominance condition for stability remains sufficient. Future research could extend the methods of this chapter to allow for asymmetry in the actuator dynamics.

Certain CD systems, such as synchrotron light sources, suffer from an ill-conditioned steady-state gain matrix. In this case, the controller produces large actuator gains in direction of small-magnitude singular values and the control system becomes sensitive to modeling errors. In practice, a static regularisation matrix is added to the IMC structure, which reduces the controller gains in direction of the small-magnitude singular values. The regularisation gain has been omitted from this analysis, but could be considered in future research directions.

Appendix

3.A Bilinear Matrix Inequalities

Convexifying (NS2)

The BMI (NS2) is given as

$$\mathcal{F}(X) = -\begin{bmatrix} I & (RX - I)^2 \\ (\dots)^* & I \end{bmatrix} \preceq 0.$$

In [35], a matrix functional $\mathcal{F} : \mathbb{R}^{n\times n} \mapsto \mathbb{R}^{m\times m}$ is called "concave" iff $\mathcal{F}((1-\alpha)X + \alpha Y) \succeq (1-\alpha)\mathcal{F}(X) + \alpha\mathcal{F}(Y)$ for all $X, Y \in \mathbb{R}^{n\times n}$ and $\alpha \in [0,1]$. Here, the concavity condition is evaluated as

$$\begin{aligned} &\mathcal{F}((1-\alpha)X + \alpha Y) - (1-\alpha)\mathcal{F}(X) - \alpha\mathcal{F}(Y) \\ &\qquad = (\alpha - \alpha^2)\begin{bmatrix} 0 & (RX + RY - 2I)^2 \\ ((RX + RY - 2I)^2)^* & 0 \end{bmatrix}, \end{aligned}$$

which shows that $\mathcal{F}(X)$ is *not* a concave function. For (NS2), Algorithm 3.1 therefore only generates feasible iterates, without necessarily converging to a local optimum.

The convexifying potential matrix functional $G = G(X, X_k)$ can be chosen as

$$G(X, X_k) := \begin{bmatrix} I \\ 0 \end{bmatrix}(RX - RX_k)^2\begin{bmatrix} 0 & I \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix}\big((RX - RX_k)^2\big)^*\begin{bmatrix} I & 0 \end{bmatrix}. \tag{3.39}$$

Note that $G(X, X_k)$ is a quadratic form in X and X_k and therefore satisfies the assumptions from Def. 3.7. The sum $\mathcal{F}(X) + G(X, X_k)$ is obtained as

$$\mathcal{F}(X) + G(X, X_k) = \begin{bmatrix} -I & R(2X + X_k R X_k - X R X_k - X_k R X) - I \\ (\dots)^* & -I \end{bmatrix}, \tag{3.40}$$

which, after setting $\mathcal{F}(X) + G(X, X_k) \succeq 0$, gives an LMI in X.

Convexifying (NS3)

The matrix inequality (NS3) is given as

$$\mathcal{F}(X,P) = -\begin{bmatrix} P^{-1} & RX-I \\ (RX-I)^* & P \end{bmatrix} \prec 0, \tag{3.41}$$

which, after pre- and post-multplying with $\mathrm{diag}(P,I)$, can be interpretated as a BMI in X and P. In [35], it is shown that $\mathcal{F}(X,P)$ is a concave matrix functional and that with

$$G(P,P_k) := \begin{bmatrix} I \\ 0 \end{bmatrix} \left(P_k^{-1} - P^{-1}\right)^* P \left(P_k^{-1} - P^{-1}\right) \begin{bmatrix} I & 0 \end{bmatrix}, \tag{3.42}$$

one obtains

$$\mathcal{F}(X,P) + G(P,P_k) = \begin{bmatrix} P_k^{-1}(P-2P_k)P_k^{-1} & -(RX-I) \\ -(RX-I)^* & -P \end{bmatrix} \prec 0, \tag{3.43}$$

which, to avoid numerical difficulties with computing P_k^{-1}, can be reformulated as

$$\begin{bmatrix} (P-2P_k) & -P_k(RX-I) \\ -(RX-I)^* P_k & -P \end{bmatrix} \prec 0. \tag{3.44}$$

Convexifying (NP1)

To convexify (NP1), the Schur complement is applied to $\|T_{\mathrm{m}}(s)\Phi_{\mathcal{S}}^2 - \Phi_{\mathcal{S}}\|_2 \leq \alpha_\omega$. The matrix inequality (NP1) is then reformulated as

$$\mathcal{F}(X) = \begin{bmatrix} -I & \mathcal{R}(X) - T_{\mathrm{m}}(\mathrm{j}\omega)(\mathcal{R}(X))^2 \\ (\dots)^* & -\alpha_\omega^2 I \end{bmatrix} \preceq 0, \tag{3.45}$$

where $\mathcal{R}(X) = RX - I$. A matrix functional $G = G(X, X_k)$ that convexifies the BMI is given by

$$G(X,X_k) := \begin{bmatrix} I \\ 0 \end{bmatrix} T_{\mathrm{m}}(\mathrm{j}\omega)(RX - RX_k)^2 \begin{bmatrix} 0 & I \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} (\dots)^* \begin{bmatrix} I & 0 \end{bmatrix}, \tag{3.46}$$

and the sum $\mathcal{F}(X) + G(X, X_k)$ evaluates to the LMI

$$\begin{aligned} &\mathcal{F}(X) + G(X,X_k) = \\ &\begin{bmatrix} -I & \mathcal{R}(X) + T_{\mathrm{m}}(\mathrm{j}\omega)((\mathcal{R}(X_k))^2 - \mathcal{R}(X_k)\mathcal{R}(X) - \mathcal{R}(X)\mathcal{R}(X_k)) \\ (\dots)^* & -\alpha_\omega^2 I \end{bmatrix} \preceq 0. \end{aligned} \tag{3.47}$$

4

Dual-Rate Cross-Directional Control

In Chapters 2 and 3, it has been shown that the single-array CD system (1.3), $y(s) = Rg(s)u(s) + d(s)$, is amenable to symmetric decompositions if the ORM $R \in \mathbb{R}^{n_y \times n_u}$ has a structural symmetry $\mathcal{S}$. If each of the matrices $R_\mathrm{s} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{s}}$ and $R_\mathrm{f} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{f}}$ have the (block-)structural symmetry $\mathcal{S}$, the same decompositions also apply to *two-array* CD systems of the form

$$y(s) = R_\mathrm{s} g_\mathrm{s}(s) u_\mathrm{s}(s) + R_\mathrm{f} g_\mathrm{f}(s) u_\mathrm{f}(s) + d(s), \tag{4.1}$$

where in view of the Diamond-II upgrade, the subscripts "s" and "f" refer to *slow* and *fast*. Instead of decompositions that are based on structural symmetries, this chapter focuses on a general two-matrix factorisation and, in particular, on decoupling (4.1) with the aim of reusing the concepts from single-array CD control [62].

The motivation for considering CD systems with two or more actuator arrays is twofold. On one hand, the single-array representation (1.3) is often based on the assumption that multiple actuator arrays are independent and/or all actuator dynamics within an array are identical [153]. On the other hand, system (1.3) can be extended with an additional array of actuators to increase the fault tolerance

This chapter is based on [88] I. Kempf, S. R. Duncan, P. J. Goulart, *et al.*, "Multi-array electron beam stabilization using block-circulant transformation and generalized singular value decomposition," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Jeju Island, Republic of Korea, Dec. 2020, pp. 3431–3436.

and/or performance of the control system [4]. Analogous to the single-array case, decoupling (4.1) facilitates the controller synthesis, but the modal decomposition *cannot* be applied when two or more actuator arrays are present. To see why, substitute the standard SVDs $R_\mathrm{s} =: \bar{U}_\mathrm{s}\bar{\Sigma}_\mathrm{s}\bar{V}_\mathrm{s}^\mathrm{T}$ and $R_\mathrm{f} =: \bar{U}_\mathrm{f}\bar{\Sigma}_\mathrm{f}\bar{V}_\mathrm{f}^\mathrm{T}$ in (4.1) to obtain

$$y(s) = \bar{U}_\mathrm{s}\bar{\Sigma}_\mathrm{s}\bar{V}_\mathrm{s}^\mathrm{T} g_\mathrm{s}(s)u_\mathrm{s}(s) + \bar{U}_\mathrm{f}\bar{\Sigma}_\mathrm{f}\bar{V}_\mathrm{f}^\mathrm{T} g_\mathrm{f}(s)u_\mathrm{f}(s) + d(s). \tag{4.2}$$

Left-multiplying (4.2) with $\bar{U}_\mathrm{s}^\mathrm{T}$ and defining the variables $\bar{y}(s) \coloneqq \bar{U}_\mathrm{s}^\mathrm{T} y(s)$, $\bar{u}_\mathrm{s}(s) \coloneqq \bar{V}_\mathrm{s}^\mathrm{T} u_\mathrm{s}(s)$, $\bar{u}_\mathrm{f}(s) \coloneqq \bar{V}_\mathrm{f}^\mathrm{T} u_\mathrm{f}(s)$ and $\bar{d}(s) \coloneqq \bar{U}_\mathrm{s}^\mathrm{T} d(s)$ yields

$$\bar{y}(s) = \bar{\Sigma}_\mathrm{s} g_\mathrm{s}(s)\bar{u}_\mathrm{s}(s) + \bar{U}_\mathrm{s}^\mathrm{T}\bar{U}_\mathrm{f}\bar{\Sigma}_\mathrm{f} g_\mathrm{f}(s)\bar{u}_\mathrm{f}(s) + \bar{d}(s), \tag{4.3}$$

which shows that, using the standard SVDs of R_s and R_f, system (4.1) is decoupled iff $\bar{U}_\mathrm{s}^\mathrm{T}\bar{U}_\mathrm{f} = I$, i.e. the controllable subspaces of R_s and R_f must be entirely overlapping.

Even though extensions of the modal decomposition have been proposed [40], [41], [51], these methods rely on the analysis of the *controllable subspaces* of the slow and fast actuators arrays, which for system (4.1) equal to $\mathcal{Y}_\mathrm{s} \coloneqq \mathrm{range}(R_\mathrm{s}) \subseteq \mathbb{R}^{n_y}$ and $\mathcal{Y}_\mathrm{f} \coloneqq \mathrm{range}(R_\mathrm{f}) \subseteq \mathbb{R}^{n_y}$. Based on the principal angles between $\mathcal{Y}_\mathrm{s}$ and $\mathcal{Y}_\mathrm{f}$, a decoupling matrix is derived that transforms the original system into a set of single-input, single-output (SISO) and a set of two-inputs, single-output (TISO) systems, but when the subspace generated by the fast actuators is entirely contained in the subspace generated by slow actuators ($\mathcal{Y}_\mathrm{f} \subset \mathcal{Y}_\mathrm{s}$), the analysis of the principal angles becomes redundant and the use of heuristics unavoidable, leaving the decoupling process unspecified.

Based on the assumption that the bandwidths of $g_\mathrm{s}(s)$ and $g_\mathrm{f}(s)$ differ significantly, other approaches split the control problem (4.1) into two loops: one feedback loop for the slow array that is possibly operated at a lower sampling/actuation frequency; and a separate feedback loop for the fast array. Such a separation is implemented in most synchrotrons that use a separate sets of slow and fast correctors [69], [115], [131], but interactions at intermediate frequencies can require the introduction of a frequency deadband between the slow and fast systems, which

– depending on the disturbance spectrum – can lead to significant performance degradation [131].

To avoid introducing a frequency deadband, one solution is to subtract the predicted effect of the slow array from the feedback signal of the fast array [69]. Another solution is to periodically subtract the DC gain from each fast actuator (individually) and to import these values into the slow feedback loop, hence shifting the low-frequency action from the fast actuator array to the slow actuator array [166]. However, this approach neglects the coupling between slow and fast actuators and relies on a SISO analysis of the combined slow and fast loops. As for the single-array system (1.3), large condition numbers $\kappa(R_\mathrm{s})$ and $\kappa(R_\mathrm{f})$ of the order of 10^3 to 10^4 significantly limit the performance, and neglecting the coupling may require to reduce controller gains further [117]. None of the approaches – the frequency deadband method [69], the periodic DC method [166], or combinations of those [116], [117] – provide frameworks that allow (4.1) to be analysed as a whole and investigate the stability, performance and robustness properties of the combined feedback loops, which might be prone to instabilities due to large $\kappa(R_\mathrm{s})$ and $\kappa(R_\mathrm{f})$.

In this chapter, it is proposed to decouple (4.1) into sets of two-inputs single-output (TISO) and SISO systems using the *generalised singular value decomposition* (GSVD) [55, Ch. 6.1.6]. The GSVD factors R_s and R_f as $R_\mathrm{s} = X\Sigma_\mathrm{s}U_\mathrm{s}^\mathrm{T}$ and $R_\mathrm{f} = X\Sigma_\mathrm{f}U_\mathrm{f}^\mathrm{T}$, where X is invertible, U_s and U_f are orthogonal and Σ_s and Σ_f are diagonal and possibly padded with zeroes (Theorem 4.2). By substituting the GSVD in (4.1), each ORM is diagonalised, which is referred to as *generalised modal transformation* (Section 4.1). It is shown that the output transformation matrix X is closely related to the hypothetical modal transformation of (1.3) when $R = \begin{bmatrix} R_\mathrm{s} & R_\mathrm{f} \end{bmatrix}$.

In contrast to the modal decomposition, the mapping to the generalised modal space is defined by the non-orthogonal matrix X, so that the performance properties of the control loop are *not* retained when transforming the decoupled systems back to original space. In particular, if $\mathcal{Y}_\mathrm{f} \subset \mathcal{Y}_\mathrm{s}$, the performance of the fast actuator array, $R_\mathrm{f}g_\mathrm{f}(s)u_\mathrm{f}(s)$, may degrade for certain disturbance directions and in Section 4.2.1, it is shown how the GSVD can be used to define an optimal static compensator

for the case that an identical controller is used for each actuator array. Moreover, since any feedback signal is multiplied by X^{-1}, an ill-conditioned X leads to large controller gains in disturbance directions aligned with (standard) left singular vectors of X associated with small-magnitude singular values. The resulting control system is prone to instabilities caused by uncertainties in R [129] and analogous to the single-array case [37], it is proposed to balance the controller gains using a regularised inverse of X (Section 4.2.2).

This chapter is organised as follows. The GSVD and the generalised modal decomposition are introduced in Section 4.1, followed in Section 4.2 by the definition of the static pre- and post-compensators that account for the non-normal transformation. For the decoupled TISO and SISO systems, an internal model control based mid-ranging design is proposed in Section 4.4.2, which is motivated by the application of the GSVD-based approach to the Diamond Light Source synchrotron. In Section 4.4, real-world results from experiments in Diamond's storage ring of both single- and two-array controllers are presented that demonstrate the practical feasibility of the two-array controller for the Diamond-II upgrade. Finally, Section 4.5 summarises the application of the two-array controller to preliminary Diamond-II data.

4.1 The Generalised Modal Decomposition

To decouple the two-array system (4.1), the original GSVD formulation [55, Ch. 6.1.6] is transposed and applied to R_s and R_f. Throughout this chapter, it is assumed that the slow actuator array spans the output space and no actuator array has redundant components, which is summarised in Assumption 4.1.

Assumption 4.1. The ORMs $R_\mathrm{s} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{s}}$ and $R_\mathrm{f} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{f}}$ of system (4.1) satisfy $\operatorname{rank}(R_\mathrm{s}) = n_\mathrm{s} = n_\mathrm{y}$ and $\operatorname{rank}(R_\mathrm{f}) = n_\mathrm{f} \leq n_\mathrm{y}$.

Systems with $\operatorname{rank}(R_\mathrm{s}) > n_\mathrm{y}$ or $\operatorname{rank}(R_\mathrm{f}) > n_\mathrm{y}$ can be reformulated to satisfy Assumption 4.1, but systems with $\operatorname{rank}(R_\mathrm{s}) < n_y$ and $\operatorname{rank}(R_\mathrm{f}) < n_y$ are *uncontrollable* in the sense of [130, Def. 6.4] and not further considered.

Theorem 4.2 (GSVD [55, Ch. 6.1.6]). *Given $R_\mathrm{s} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{s}}$ and $R_\mathrm{f} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{f}}$ satisfying Assumption 4.1, the GSVD factors R_s and R_f as*

$$R_\mathrm{s} = X \begin{bmatrix} \Sigma_\mathrm{s} & 0 \\ 0 & I \end{bmatrix} U_\mathrm{s}^T, \qquad R_\mathrm{f} = X \begin{bmatrix} \Sigma_\mathrm{f} \\ 0 \end{bmatrix} U_\mathrm{f}^T, \tag{4.4}$$

where $X \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{y}}$ with $\det(X) \neq 0$ is the matrix of generalised output modes, $\Sigma_\times \in \mathbb{R}^{n_\mathrm{f} \times n_\mathrm{f}}$ with $\Sigma_\times = \operatorname{diag}(\sigma_{\times,1}, \ldots, \sigma_{\times,n_\mathrm{f}}) \succ 0$ and $\times = \{\mathrm{s}, \mathrm{f}\}$ are the matrices of generalised singular values that satisfy $\sigma_{\mathrm{s},i}^2 + \sigma_{\mathrm{f},i}^2 = 1$, and $U_\times \in \mathbb{R}^{n_\times \times n_\times}$ with $U_\times^T U_\times = I$ are the matrices of generalised input modes.

The GSVD from Theorem 4.2 is substituted in (4.1) to obtain

$$y(s) = X \begin{bmatrix} \Sigma_\mathrm{s} & 0 \\ 0 & I \end{bmatrix} U_\mathrm{s}^\mathrm{T} g_\mathrm{s}(s) u_\mathrm{s}(s) + X \begin{bmatrix} \Sigma_\mathrm{f} \\ 0 \end{bmatrix} U_\mathrm{f}^\mathrm{T} g_\mathrm{f}(s) u_\mathrm{f}(s) + d(s). \tag{4.5}$$

Left-multiplying (4.5) with X^{-1} and introducing the *generalised modal variables*

$$\tilde{y}(s) \coloneqq X^{-1} y(s), \quad \tilde{u}_\mathrm{s}(s) \coloneqq U_\mathrm{s}^\mathrm{T} u_\mathrm{s}(s), \quad \tilde{u}_\mathrm{f}(s) \coloneqq U_\mathrm{f}^\mathrm{T} u_\mathrm{f}(s), \quad \tilde{d}(s) \coloneqq X^{-1} d(s), \tag{4.6}$$

yields the *generalised modal representation* of (4.1):

$$\tilde{y}(s) = \begin{bmatrix} \Sigma_\mathrm{s} & 0 \\ 0 & I \end{bmatrix} g_\mathrm{s}(s) \tilde{u}_\mathrm{s}(s) + \begin{bmatrix} \Sigma_\mathrm{f} \\ 0 \end{bmatrix} g_\mathrm{f}(s) \tilde{u}_\mathrm{f}(s) + \tilde{d}(s). \tag{4.7}$$

Because the matrices Σ_s and Σ_f are diagonal, the MIMO representation (4.1) is decoupled into n_f TISO systems and $n_\mathrm{s} - n_\mathrm{f}$ SISO systems in (4.7). The separation between output directions that are affected by TISO systems and those affected by SISO systems is given by

$$\mathcal{Y}_{\mathrm{s}\cap\mathrm{f}} \coloneqq \mathcal{Y}_\mathrm{s} \bigcap \mathcal{Y}_\mathrm{f} = \operatorname{span}(x_1, \ldots, x_{n_\mathrm{f}}), \tag{4.8a}$$

$$\mathcal{Y}_{\mathrm{s}\setminus\mathrm{f}} \coloneqq \mathcal{Y}_\mathrm{s} \setminus \mathcal{Y}_\mathrm{f} = \operatorname{span}\big(x_{n_\mathrm{f}+1}, \ldots, x_{n_\mathrm{y}}\big), \tag{4.8b}$$

where $x_i \in \mathbb{R}^{n_\mathrm{y}}$ are the columns of X and $\mathcal{Y}_{\mathrm{s}\cap\mathrm{f}}$ and $\mathcal{Y}_{\mathrm{s}\setminus\mathrm{f}}$ are referred to as TISO and SISO subspace in the following. Note that (4.8) is a consequence of (4.1) that cannot be altered by the choice of decomposition, but compared to an arbitrary factorisation, the output basis provided by Theorem 4.2 is closely related to a hypothetical single-array system, which is shown in the following lemma.

Lemma 4.3. *Consider the factorisation of R_s and R_f from Theorem 4.2, and let $R \coloneqq \begin{bmatrix} R_\mathrm{s} & R_\mathrm{f} \end{bmatrix} \in \mathbb{R}^{n_\mathrm{y} \times (n_\mathrm{s}+n_\mathrm{f})}$. Then, the standard singular values and standard left singular vectors of R equal those of X.*

Proof. Note that according to Theorem 4.2, the generalised singular values satisfy $\Sigma_\mathrm{s}^2 + \Sigma_\mathrm{f}^2 = I$. Express R^T using (4.4) and compute $RR^\mathrm{T} = X \begin{bmatrix} \Sigma_\mathrm{s}^2 + \Sigma_\mathrm{f}^2 & 0 \\ 0 & I \end{bmatrix} X^\mathrm{T} = XX^\mathrm{T}$, from which the claim follows. □

Lemma 4.3 shows that the decomposition of the two-array system (4.1) through Theorem 4.2 relates to the modal decomposition of a hypothetical single-array system (1.3) with $n_u = n_\mathrm{s} + n_\mathrm{f}$ and $R = \begin{bmatrix} R_\mathrm{s} & R_\mathrm{f} \end{bmatrix}$ and therefore allows the TISO and SISO subspaces (4.8) to be related to the standard left singular vectors of R, which determines the spatial distribution of the disturbance spectrum (Section 4.4.3). Consider the standard SVD of R as in (1.6), then X can be formed as

$$X = U\Sigma V_X^\mathrm{T} \tag{4.9}$$

where V_X with $V_X^\mathrm{T} V_X = I$ is the matrix of standard right singular vectors of X. The mapping of a vector $y \in \mathbb{R}^{n_\mathrm{y}}$ to generalised modal space, $X^{-1}y = V_X \Sigma^{-1} U^\mathrm{T} y$, therefore consists of mapping y to mode space first, before scaling by Σ^{-1} and finally multiplying it with V_X^T. It remains unclear how V_X can be interpreted, but the following lemma characterises the gain ratio between each actuator array using a function $f : \mathcal{Y}_{\mathrm{s}\cap\mathrm{f}} \mapsto \mathbb{R}_{\geq 1}$, where $\mathbb{R}_{\geq 1} \coloneqq \mathbb{R} \cap \{x \in \mathbb{R} \mid x \geq 1\}$, that has been used for a variational formulation of the GSVD [26].

Lemma 4.4. *Consider the function $f : \mathcal{Y}_{\mathrm{s}\cap\mathrm{f}} \mapsto \mathbb{R}_{\geq 1}$,*

$$f(y) \coloneqq \frac{1}{2}\left(\frac{\|y^T R_\mathrm{s}\|_2^2}{\|y^T R_\mathrm{f}\|_2^2} + \frac{\|y^T R_\mathrm{f}\|_2^2}{\|y^T R_\mathrm{s}\|_2^2} \right), \tag{4.10}$$

and its gradient $\nabla f : \mathcal{Y}_{\mathrm{s}\cap\mathrm{f}} \mapsto \mathbb{R}^{n_\mathrm{y}}$,

$$\begin{aligned} \nabla f(y) \coloneqq \frac{1}{\|R_\mathrm{f}^T y\|_2^2} \left(R_\mathrm{s} R_\mathrm{s}^T y - \frac{\|R_\mathrm{s}^T y\|_2^2}{\|R_\mathrm{f}^T y\|_2^2} R_\mathrm{f} R_\mathrm{f}^T y \right) + \\ \frac{1}{\|R_\mathrm{s}^T y\|_2^2} \left(R_\mathrm{f} R_\mathrm{f}^T y - \frac{\|R_\mathrm{f}^T y\|_2^2}{\|R_\mathrm{s}^T y\|_2^2} R_\mathrm{s} R_\mathrm{s}^T y \right). \end{aligned} \tag{4.11}$$

It holds that $\nabla f(y) = 0$ *at* $y_i := (XX^T)^{-1}x_i$*, where* $x_1, \ldots, x_{n_y}$ *are the columns of* X *from Theorem 4.2. In addition,* $\nabla f(y_i) = 0$ *and* $f(y_i) = 1$ *if* y_i *is a shared standard left singular vector for* R_s *and* R_f *associated with an identical singular value.*

Proof. Suppose that $y_i = (XX^T)^{-1}x_i$ and express $R_\times R_\times^T$ using (4.4), so that $R_\times R_\times^T y_i = R_\times R_\times^T (XX^T)^{-1}x_i = X\Sigma_\times^2 X^{-1}x_i = \sigma_{\times,i}^2 x_i$, where $\sigma_{\times,i}^2$ is a generalised singular value and $\times = \{s, f\}$. Substituting in the first term on the right-hand side of (4.11) yields

$$R_s R_s^T y_i - \frac{\|R_s^T y_i\|_2^2}{\|R_f^T y_i\|_2^2} R_f R_f^T y_i = \sigma_{s,i}^2 x_i - \frac{\sigma_{s,i}^2}{\sigma_{f,i}^2}\sigma_{f,i}^2 x_i = 0,$$

and similarly for the second term on the right-hand side of (4.11), which follows that $\nabla f(y_i) = 0$.

For the second part of the claim, note that if x_i is a shared standard left singular vector for R_s and R_f, then it must be one for R from Lemma 4.3 and X as well. It follows that $y_i = x_i/\hat{\sigma}_i^2$, where $\hat{\sigma}_i$ is the corresponding standard singular value of R, and hence $\|y_i^T R_s\|_2^2 = \|y_i^T R_f\|_2^2$. □

The function $f(y)$ can be interpreted as a measure for the ratio of gains that are required by each actuator array to produce a correction y. Ignoring the dependency on the Laplace variable in (4.1), the actuator effort required to produce a correction is $y = R_s u_s + R_f u_f$, so that the terms $\|y^T R_s\|_2 \geq \|y^T R_s u_s\|_2/\|u_s\|_2$ and $\|y^T R_f\|_2 \geq \|y^T R_f u_f\|_2/\|u_f\|_2$ can be seen as the relative contribution of each actuator array. If a vector y_i exist that is a shared standard left singular vector associated with standard singular values $\hat{\sigma}_{s,i}$ and $\hat{\sigma}_{f,i}$ for R_s and R_f, then $f(y_i) = \frac{1}{2}\big(\hat{\sigma}_{s,i}^2/\hat{\sigma}_{f,i}^2 + \hat{\sigma}_{f,i}^2/\hat{\sigma}_{s,i}^2\big)$.

4.2 Compensators

As in the case of the single-array system (1.3), the IMC structure is used but the generalised modal decomposition could also be combined with other feedback structures. The IMC structure is shown in Fig. 4.1, where $u(s) = \begin{pmatrix} u_s(s) \\ u_f(s) \end{pmatrix}$,

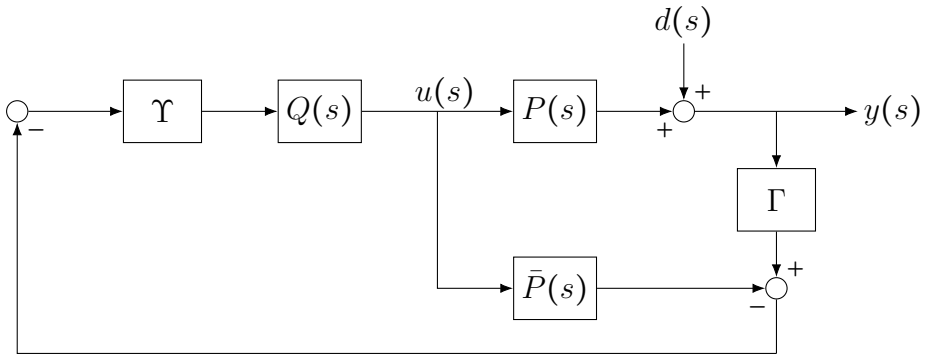


Figure 4.1: IMC structure for the two-array system (4.1) with compensators Γ and Υ.

$\Gamma \in \mathbb{R}^{n_y \times n_y}$ and

$$P(s) := \begin{bmatrix} R_s g_s(s) & R_f g_f(s) \end{bmatrix} \in \mathbb{C}^{n_y \times (n_s + n_f)}, \tag{4.12a}$$

$$\bar{P}(s) := \begin{bmatrix} \bar{R}_s \bar{g}_s(s) & \bar{R}_f \bar{g}_f(s) \end{bmatrix} \in \mathbb{C}^{n_y \times (n_s + n_f)}, \tag{4.12b}$$

$$Q(s) := \operatorname{diag}(Q_s(s),\, Q_f(s)) \in \mathbb{C}^{(n_s + n_f) \times 2n_y}, \tag{4.12c}$$

$$\Upsilon := \begin{bmatrix} \Upsilon_s \\ \Upsilon_f \end{bmatrix} \in \mathbb{C}^{2n_y \times n_y}. \tag{4.12d}$$

The matrix $\bar{P}(s)$ is a nominal model of the real plant $P(s)$ and Υ and Γ are (static) input and output compensators introduced in Sections 4.2.1 and 4.2.2. As opposed to the standard feedback structure, the main advantage of IMC is that the closed-loop properties are directly related to the open-loop transfer function and not to the inverse of the return difference. Other advantages are that the IMC structure is naturally amenable to plants with time delays [107, Ch. 3.5] and the feedback signal can be used as an input to a fault detection algorithm [160].

Before considering the input and output compensators, so that $\Gamma = I$ and $\Upsilon = \begin{bmatrix} I & I \end{bmatrix}^{\mathrm{T}}$, the feedback laws for the decoupled system (4.7) are assumed to be given by

$$\tilde{u}_s(s) = -\tilde{Q}_s(s)\tilde{d}(s) := -\begin{bmatrix} \Sigma_s^{-1} & 0 \\ 0 & I \end{bmatrix} q_s(s)\tilde{d}(s), \tag{4.13a}$$

$$\tilde{u}_f(s) = -\tilde{Q}_f(s)\tilde{d}(s) := -\begin{bmatrix} \Sigma_f^{-1} & 0 \end{bmatrix} q_f(s)\tilde{d}(s), \tag{4.13b}$$

where $q_s : \mathbb{C} \mapsto \mathbb{C}$ and $q_f : \mathbb{C} \mapsto \mathbb{C}$ are stable and realisable transfer functions. The

filters are recovered in original space as $Q_{\mathrm{s}}(s) = U_{(\cdot)}\tilde{Q}_{(\cdot)}(s)X^{-1}$, so that

$$Q_{\mathrm{s}}(s) = U_{\mathrm{s}}\begin{bmatrix}\Sigma_{\mathrm{s}}^{-1} & 0\\ 0 & I\end{bmatrix}X^{-1}q_{\mathrm{s}}(s), \tag{4.14a}$$

$$Q_{\mathrm{f}}(s) = U_{\mathrm{f}}\begin{bmatrix}\Sigma_{\mathrm{f}}^{-1} & 0\end{bmatrix}X^{-1}q_{\mathrm{f}}(s), \tag{4.14b}$$

The structure (4.13a) results in output sensitivities that are identical for each TISO and each SISO mode, i.e.

$$S_i(s) = S_{\mathrm{s}\cap\mathrm{f}}(s), \quad i = 1,\ldots,n_{\mathrm{f}}, \qquad S_j(s) = S_{\mathrm{s}\setminus\mathrm{f}}(s), \quad j = n_{\mathrm{f}}+1,\ldots,n_{\mathrm{y}}, \tag{4.15}$$

where $S_i : \mathbb{C} \mapsto \mathbb{C}$ is the transfer function from component i of $\tilde{d}(s)$ to component i of $\tilde{y}(s)$ and $S_{\mathrm{s}\cap\mathrm{f}} : \mathbb{C} \mapsto \mathbb{C}$ and $S_{\mathrm{s}\setminus\mathrm{f}} : \mathbb{C} \mapsto \mathbb{C}$ the TISO and SISO output sensitivities, respectively. Restricting the controller dynamics to a scalar function as in (4.13a) is a design constraint commonly accepted for single-array CD control [54]. For the two-array system, this restriction allows a static (frequency-independent) compensator to be designed that guarantees identical performance in original and generalised modal space (Section 4.2.1). To guarantee a zero steady-state for disturbances with non-zero offsets, it is assumed that the output sensitivities satisfy

$$S_{\mathrm{s}\cap\mathrm{f}}(0) = S_{\mathrm{s}\setminus\mathrm{f}}(0) = 0, \tag{4.16}$$

which means for $q_{\mathrm{s}}(s)$ and $q_{\mathrm{f}}(s)$ that $g_{\mathrm{s}}(0)q_{\mathrm{s}}(0) = 1$ and $g_{\mathrm{f}}(0)q_{\mathrm{f}}(0) = 0$. Substitute the inputs (4.13a) in the dynamics (4.7) to obtain the transfer function $\tilde{S} : \mathbb{C}^{n_y} \mapsto \mathbb{C}^{n_y}$ from $\tilde{d}(s)$ to $\tilde{y}(s)$,

$$\begin{aligned}
\tilde{y}(s) &= \left(I - \Sigma_{\mathrm{s}}g_{\mathrm{s}}(s)\tilde{Q}_{\mathrm{s}}(s) - \begin{bmatrix}\Sigma_{\mathrm{f}}\\ 0\end{bmatrix}g_{\mathrm{f}}(s)\tilde{Q}_{\mathrm{f}}(s)\right)\tilde{d}(s),\\
&= \left(I - \begin{bmatrix}I & 0\\ 0 & I\end{bmatrix}g_{\mathrm{s}}(s)q_{\mathrm{s}}(s) - \begin{bmatrix}I & 0\\ 0 & 0\end{bmatrix}g_{\mathrm{f}}(s)q_{\mathrm{f}}(s)\right)\tilde{d}(s)\\
&= \begin{bmatrix}S_{\mathrm{s}\cap\mathrm{f}}(s)I & 0\\ 0 & S_{\mathrm{s}\setminus\mathrm{f}}(s)I\end{bmatrix}\tilde{d}(s)\\
&=: \tilde{S}(s)\tilde{d}(s),
\end{aligned} \tag{4.17}$$

and invert the transformations (4.6) to map the output sensitivity back to the original space:

$$\begin{aligned}
y(s) &= X\tilde{S}(s)X^{-1}d(s)\\
&= \left(I - Ig_{\mathrm{s}}(s)q_{\mathrm{s}}(s) - X\begin{bmatrix}I & 0\\ 0 & 0\end{bmatrix}X^{-1}g_{\mathrm{f}}(s)q_{\mathrm{f}}(s)\right)d(s)\\
&=: S(s)d(s).
\end{aligned} \tag{4.18}$$

From (4.18), it can be seen that if $\text{rank}(R_\text{s}) = \text{rank}(R_\text{f}) = n_\text{y}$, then

$$\|S(s)\|_2 = \|\tilde{S}(s)\|_2 = \max\{|S_{\text{s}\cap\text{f}}(s)|, |S_{\text{s}\setminus\text{f}}(s)|\}, \tag{4.19}$$

but if $\text{rank}(R_\text{f}) < \text{rank}(R_\text{s}) = n_y$, $\|S(s)\|_2 \neq \|\tilde{S}(s)\|_2$ and an upper bound on $\|S(s)\|_2$ is given by

$$\|S(s)\|_2 \approx \|X\tilde{S}(s)X^{-1}\|_2 \leq \kappa(X)\|\tilde{S}(s)\|_2. \tag{4.20}$$

Hence if R is ill-conditioned, as is the case for synchrotrons, then according to Lemma 4.3, X is ill-conditioned and the performance of the control system in original space can be arbitrarily poor. In the following section, the input compensator Υ is designed that serves to remove the potential performance difference highlighted in (4.20).

4.2.1 Input Compensator

Consider the output sensitivity in original $S(s)$ space (4.18) and the IMC structure from Fig. 4.1, where for the remainder of this section it is assumed that $\Gamma = I$ and $\bar{P}(s) = P(s)$. In (4.18), the matrix in the term associated with $u_\text{f}(s)$ is

$$X\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}X^{-1} = \begin{bmatrix} X_{11} & 0 \\ X_{21} & 0 \end{bmatrix}X^{-1} =: X_{\text{s}\cap\text{f}}X^{-1}, \tag{4.21}$$

where $X_{11} \in \mathbb{R}^{n_\text{f}\times n_\text{f}}$ and $X_{21} \in \mathbb{R}^{(n_\text{s}-n_\text{f})\times n_\text{f}}$. Since (4.21) is responsible for the potential performance difference, it seems natural to set $\Upsilon_\text{s} = I$ and include Υ_f in the control law (4.13a), i.e. $\tilde{u}_\text{f}(s) = -\begin{bmatrix}\Sigma_\text{f}^{-1} & 0\end{bmatrix}\Upsilon_\text{f}q_\text{f}(s)\tilde{y}(s)$, so that $u_\text{s}(s)$ and $u_\text{f}(s)$ are given in the original space by

$$u_\text{s}(s) = -U_\text{s}\begin{bmatrix}\Sigma_\text{s}^{-1} & 0 \\ 0 & I\end{bmatrix}X^{-1}q_\text{s}(s)d(s), \tag{4.22a}$$

$$u_\text{f}(s) = -U_\text{f}\begin{bmatrix}\Sigma_\text{f}^{-1} & 0\end{bmatrix}\Upsilon_\text{f}X^{-1}q_\text{f}(s)d(s). \tag{4.22b}$$

To construct Υ_f, the following Lemma 4.5 is used.

Lemma 4.5. *Let $A^\dagger$ be the Moore-Penrose pseudoinverse [55, P5.5.2] of $A \in \mathbb{R}^{n\times n}$ with $\text{rank}(A) = r \leq n$. Then $AA^\dagger$ is symmetric and has r unity and $n-r$ zero eigenvalues.*

Proof. Let the standard SVD of A be given as $A = U\,\mathrm{diag}(\Sigma, 0)V^{\mathrm{T}}$ with $\Sigma \in \mathbb{R}^{r\times r} \succ 0$. Then, $A^{\dagger} = V\,\mathrm{diag}(\Sigma^{-1}, 0)U^{\mathrm{T}}$ and $AA^{\dagger} = U\,\mathrm{diag}(I, 0)U^{\mathrm{T}}$ with $I \in \mathbb{R}^{r\times r}$. □

If Υ_{f} is chosen as

$$\Upsilon_{\mathrm{f}} \coloneqq X_{\mathrm{s}\cap\mathrm{f}}^{\dagger} X, \tag{4.23}$$

the output sensitivity (4.18) becomes

$$y(s) = \big(I - I g_{\mathrm{s}}(s) q_{\mathrm{s}}(s) - X_{\mathrm{s}\cap\mathrm{f}} X_{\mathrm{s}\cap\mathrm{f}}^{\dagger} g_{\mathrm{f}}(s) q_{\mathrm{f}}(s)\big) d(s). \tag{4.24}$$

By setting $A = X_{\mathrm{s}\cap\mathrm{f}}$ with $\mathrm{rank}(A) = \mathrm{rank}(X_{\mathrm{s}\cap\mathrm{f}}) = n_{\mathrm{f}}$ in Lemma 4.5, it becomes clear that with Υ_{f} defined as in (4.23), the difference between performance of the original and the generalised modal space vanishes, i.e. $\|S(s)\|_2 = \|\tilde{S}(s)\|_2$. From the structure of the matrix $X_{\mathrm{s}\cap\mathrm{f}}$ in (4.21), the structure of $X_{\mathrm{s}\cap\mathrm{f}}^{\dagger}$ is

$$X_{\mathrm{s}\cap\mathrm{f}}^{\dagger} = \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & 0 \end{bmatrix},$$

where, because the non-zero columns of $X_{\mathrm{s}\cap\mathrm{f}}$ are linearly independent, the blocks $Z_{11} \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{f}}}$ and $Z_{12} \in \mathbb{R}^{n_{\mathrm{f}}\times(n_{\mathrm{s}}-n_{\mathrm{f}})}$ must satisfy

$$\begin{bmatrix} Z_{11} & Z_{12} \end{bmatrix} \begin{bmatrix} X_{11} \\ X_{21} \end{bmatrix} = Z_{11}X_{11} + Z_{12}X_{21} = I. \tag{4.25}$$

The n_{f} output directions from Lemma 4.5 that are unaffected by Υ_{f} and attenuated by $S_{\mathrm{s}\cap\mathrm{f}}(s)$ therefore lie in $\mathcal{Y}_{\mathrm{s}\cap\mathrm{f}}$, whereas there exist $n_{\mathrm{s}} - n_{\mathrm{f}}$ output directions that are zeroed out by Υ_{f} and attenuated by $S_{\mathrm{s}\setminus\mathrm{f}}(s)$.

To see the effect of Υ_{f} onto the generalised modes, consider mapping (4.24) to generalised modal space using (4.6):

$$\tilde{y}(s) = \big(I - I g_{\mathrm{s}}(s) q_{\mathrm{s}}(s) - X^{-1} X_{\mathrm{s}\cap\mathrm{f}} X_{\mathrm{s}\cap\mathrm{f}}^{\dagger} X g_{\mathrm{f}}(s) q_{\mathrm{f}}(s)\big) \tilde{d}(s). \tag{4.26}$$

According to Lemma 4.5, the 2-norm of (4.26) is identical to (4.17). However, the input compensator has the effect of coupling the TISO modes with the SISO modes. To see this, define $X_{\mathrm{s}\setminus\mathrm{f}} \coloneqq X - X_{\mathrm{s}\cap\mathrm{f}}$, and note that by the definition of the

pseudo-inverse, the last $n_s - n_f$ rows of $X^{\dagger}_{s\cap f}$ are zero. Then, the term $X^{-1}X_{s\cap f}X^{\dagger}_{s\cap f}X$ from (4.26) can be expanded as

$$X^{-1}X_{s\cap f}X^{\dagger}_{s\cap f}X = X^{-1}X_{s\cap f} + X^{-1}X_{s\cap f}X^{\dagger}_{s\cap f}X_{s\setminus f} = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & \star \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} I & \star \\ 0 & 0 \end{bmatrix},$$

where $\star$ is a non-zero block given by

$$\begin{bmatrix} Z_{11} & Z_{12} \end{bmatrix}\begin{bmatrix} X_{12} \\ X_{22} \end{bmatrix} = Z_{11}X_{12} + Z_{12}X_{22}. \tag{4.27}$$

4.2.2 Output Compensator

While the performance difference (4.20) has been removed by the input compensator, it can be seen from (4.22) that the controllers for both arrays are proportional to X^{-1}. According to Lemma 4.3, X has the same condition number as R, so the disturbance directions associated with small-magnitude singular values of R therefore cause large input gains for both actuator arrays, which can lead to actuator saturation. Moreover, if the plant model is inexact, i.e. $\bar{P}(s) \neq P(s)$, the resulting control system is likely to be prone to instabilities [129]. The output compensator is used to remedy this problem and the following section revisits its design for the single-array system, before adapting it to the two-array system.

Single-Array Systems

Consider the single-array system (1.3) and the control law in modal space,

$$\hat{u}(s) \coloneqq -\hat{Q}(s)\hat{d}(s) \coloneqq -\Sigma^{-1}q(s)\hat{d}(s), \tag{4.28}$$

where $q : \mathbb{C} \mapsto \mathbb{C}$ is such that $q(s)g(s) = T_m(s)$ with $T_m(0) = 1$, which results in a complementary output sensitivity $\hat{T}(s) \coloneqq T_m(s)I$ that describes the transfer function from the (zero) reference signal to the output $y(s)$. The standard feedback equivalent of $\hat{Q}(s)$, $\hat{C}(s) \coloneqq \left(I - \hat{Q}(s)\hat{P}(s)\right)^{-1}\hat{Q}(s)$, is given by

$$\hat{C}(s) = \Sigma^{-1}\frac{q(s)}{1 - q(s)g(s)} = \Sigma^{-1}\frac{q(s)}{1 - T_m(s)}. \tag{4.29}$$

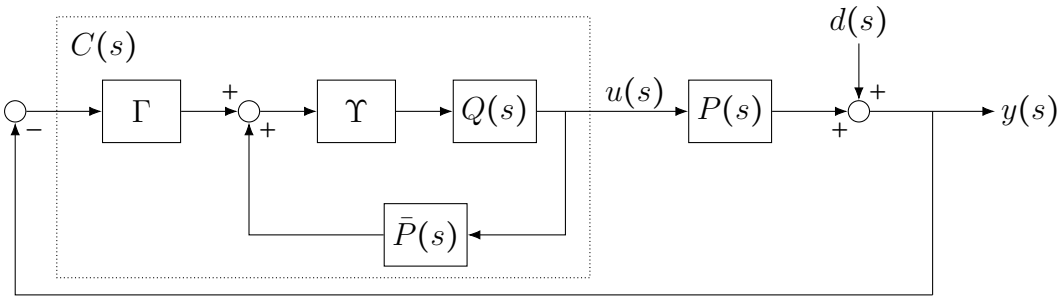


Figure 4.2: IMC structure from Fig. 4.1 rearranged into the standard feedback structure.

To accommodate systems with $\kappa(R) = \kappa(\Sigma) \gg 1$, a matrix $\hat{\Gamma} \in \mathbb{R}^{n_y \times n_y}$ is defined as follows:

$$\hat{\Gamma} := \left(\Sigma^2 + \mu I\right)^{-1}\Sigma^2 = \operatorname{diag}\left(\frac{\sigma_1^2}{\sigma_1^2+\mu}, \ldots, \frac{\sigma_{n_y}^2}{\sigma_{n_y}^2+\mu}\right), \tag{4.30}$$

where the scalar $\mu \geq 0$ is the regularisation parameter. Right-multiplying $\hat{C}(s)$ with $\hat{\Gamma}$ modifies the controller as

$$\hat{C}(s)\hat{\Gamma} = \left(\Sigma^2 + \mu I\right)^{-1}\Sigma\frac{q(s)}{1 - T_m(s)}, \tag{4.31}$$

i.e. the inverse in (4.29) has been replaced with the regularised inverse $(\Sigma^2 + \mu I)^{-1}\Sigma$, thus attenuating input gains associated with small singular values for $\mu > 0$. With the controller defined as in (4.31), the diagonal elements of the open-loop $\hat{L}(s) := \Sigma g(s)\hat{C}(s)\hat{\Gamma} =: \operatorname{diag}(\ell_1(s), \ldots, \ell_{n_y})$ become

$$\ell_i(s) = \frac{\sigma_i^2}{\sigma_i^2 + \mu}\,\frac{T_m(s)}{1 - T_m(s)}. \tag{4.32}$$

The regularisation parameter μ therefore changes the open-loop bandwidth as well as the position of the low-frequency asymptote of the Nyquist diagram of $\ell_i(s)$. Note that for $\sigma_i^2 \gg \mu$, the effect of μ is negligible, whereas for $\sigma_j^2 \ll \mu$, the open-loop gain and the closed-loop bandwidth are effectively reduced.

Two-Array Systems

Consider rearranging Fig. 4.1 into the standard feedback structure shown in Fig. 4.2. For $\Upsilon = \begin{bmatrix} I & I \end{bmatrix}^T$ and $\bar{P}(s) = P(s)$, the standard feedback controller

$C(s) = (I - Q(s)P(s))^{-1}Q(s)\Gamma$ is obtained as (Appendix 4.A)

$$C(s) = \begin{bmatrix} U_{\mathrm{s}} & 0 \\ 0 & U_{\mathrm{f}} \end{bmatrix} \begin{bmatrix} \frac{1-S_{\mathrm{s}\setminus\mathrm{f}}(s)}{g_{\mathrm{s}}(s)S_{\mathrm{s}\cap\mathrm{f}}(s)}\Sigma_{\mathrm{s}}^{-1} & 0 \\ 0 & \frac{1-S_{\mathrm{s}\setminus\mathrm{f}}(s)}{g_{\mathrm{s}}(s)S_{\mathrm{s}\setminus\mathrm{f}}(s)}I \\ \frac{S_{\mathrm{s}\setminus\mathrm{f}}(s)-S_{\mathrm{s}\cap\mathrm{f}}(s)}{g_{\mathrm{f}}(s)S_{\mathrm{s}\cap\mathrm{f}}(s)}\Sigma_{\mathrm{f}}^{-1} & 0 \end{bmatrix} X^{-1}\Gamma, \tag{4.33}$$

where there inverse of X reappears as in (4.22). To attenuate feedback signals that are aligned with directions associated with small singular values of X, the inverse X^{-1} in (4.33) can be replaced by setting

$$\Gamma := X\big(X^{\mathrm{T}}WX + \mu I\big)^{-1}(WX)^{\mathrm{T}}, \tag{4.34}$$

for some $\mu > 0$ and $W \in \mathcal{S}_{++}$. With Γ as defined in (4.34), the term $X^{-1}\Gamma$ in (4.33) is replaced by $X_\mu^{-1} := (X^{\mathrm{T}}WX + \mu I)^{-1}(WX)^{\mathrm{T}}$, which can be interpreted as the factor matrix obtained from the following regularised least squares problem [55, Ch. 6.1.5]:

$$\underset{\nu\in\mathbb{R}^{n_y}}{\text{minimise}} \quad \|W^{\frac{1}{2}}X\nu + b\|_2^2 + \mu\|\nu\|_2^2.$$

If W is chosen as $W = U\,\mathrm{diag}(w_1, \ldots, w_{n_{\mathrm{y}}})U^{\mathrm{T}}$, where U is the matrix of standard left singular vectors of X, the weights w_i can be chosen to prioritize certain (standard) modes that are particularly affected by disturbances. This follows from Lemma 4.3, which states that R and X share the same matrix of left singular vectors.

Because the matrix $X^{\mathrm{T}}WX$ is positive definite, the 2-norm of X_μ^{-1} decreases as μ increases. Let $C_0(s)$ denote the standard controller with $\Gamma = I$ ($\mu = 0$). The gain of $C(s)$ for $\Gamma \neq I$ can be upper-bounded by

$$\|C(s)\|_2 = \|C_0(s)X_\mu^{-1}\|_2 \leq \|C_0(s)\|_2\|X_\mu^{-1}\|_2 \leq \|C_0(s)\|_2,$$

from which it can be seen that the parameter μ controls the bandwidth of the open-loop transfer function $L(s) := P(s)C(s)$, which is obtained as (Appendix 4.A)

$$\begin{aligned} L(s) &= X \underbrace{\begin{bmatrix} \frac{1-S_{\mathrm{s}\cap\mathrm{f}}(s)}{S_{\mathrm{s}\cap\mathrm{f}}(s)}I & 0 \\ 0 & \frac{1-S_{\mathrm{s}\setminus\mathrm{f}}(s)}{S_{\mathrm{s}\setminus\mathrm{f}}(s)}I \end{bmatrix}}_{=:\mathrm{diag}(\ell_{\mathrm{s}\cap\mathrm{f}}(s)I,\, \ell_{\mathrm{s}\setminus\mathrm{f}}(s)I)} X^{-1}\Gamma \\ &=: L_0(s)\Gamma. \end{aligned} \tag{4.35}$$

The following proposition summarises the impact of Γ on the robustness of the closed-loop system.

Proposition 4.6. *Suppose that the system from Fig. 4.1 is stable for* $\Upsilon = [I \quad I]^T$ *and* $\Gamma = I$*. Additionally, suppose that* $Re(\det(L_0(\mathrm{j}\omega))) \geq -1/2 \ \forall\omega$*. For* $\Upsilon = [I \quad I]^T$*, the gain margin [130, Ch. 2.4.5] of the control system from Fig. 4.1 increases for increasing* μ*.*

Proof. By the generalised Nyquist theorem [130, Thm. 4.9], the closed-loop is stable iff the Nyquist plot of $\det(L(s))$ does not encircle the point $-1 + j0$. By the assumptions of Lemma 4.6, the Nyquist plot of $\det(L_0(s))$ does not encircle the point $-1 + j0$ and is located to the right of the line $-1/2 + jx$, $x \in \mathbb{R}$. The claim follows from noting that $\det(L(s)) = \det(L_0(s))\det(\Gamma)$ and that $1 > \det(\Gamma) > 0$ decreases for increasing μ. □

For $\Upsilon = [I \quad \Upsilon_{\mathrm{f}}^{\mathrm{T}}]^{\mathrm{T}}$ with Υ_{f} as in (4.23), the coupling between TISO and SISO modes complicates computing $L_0(s)$, but $L(s)$ remains proportional to Γ, so that Proposition 4.6 remains valid.

To see how Γ (4.34) affects the closed-loop poles, consider rearranging the complementary sensitivity $T(s) = (I + L(s))^{-1}L(s)$ for $W = I$ as

$$\begin{aligned} T(s) &= \left(I + X\begin{bmatrix} \ell_{\mathrm{s}\cap\mathrm{f}}(s)I & 0 \\ 0 & \ell_{\mathrm{s}\setminus\mathrm{f}}(s)I \end{bmatrix}\left(X^{\mathrm{T}}X + \mu I\right)^{-1}X^{\mathrm{T}}\right)^{-1}L(s), \\ &= X^{-\mathrm{T}}\left(\left(X^{\mathrm{T}}X\right)^{-1} + \begin{bmatrix} \ell_{\mathrm{s}\cap\mathrm{f}}(s)I & 0 \\ 0 & \ell_{\mathrm{s}\setminus\mathrm{f}}(s)I \end{bmatrix}\left(X^{\mathrm{T}}X + \mu I\right)^{-1}\right)^{-1}X^{-1}L(s), \\ &= X_\mu\left(\begin{bmatrix} (1+\ell_{\mathrm{s}\cap\mathrm{f}}(s))I & 0 \\ 0 & (1+\ell_{\mathrm{s}\setminus\mathrm{f}}(s))I \end{bmatrix} + \mu\left(X^{\mathrm{T}}X\right)^{-1}\right)^{-1}X^{-1}L(s). \end{aligned} \tag{4.36}$$

The closed-loop poles are therefore those values of $s \in \mathbb{C}$ for which

$$\det\left(\begin{bmatrix} (1+\ell_{\mathrm{s}\cap\mathrm{f}}(s))I & 0 \\ 0 & (1+\ell_{\mathrm{s}\setminus\mathrm{f}}(s))I \end{bmatrix} + \mu\left(X^{\mathrm{T}}X\right)^{-1}\right) = 0. \tag{4.37}$$

For $\mu = 0$, the closed-loop poles belong to a subset of $\pi_i(T_{\mathrm{s}\cap\mathrm{f}}) \cup \pi_i(T_{\mathrm{s}\setminus\mathrm{f}}) \cup \{0\}$ (Proposition 4.6). To examine (4.37) for $\mu > 0$, condition (4.37) is further simplified by setting $n_{\mathrm{s}} = n_{\mathrm{f}}$, so that with $\ell_{\mathrm{s}\cap\mathrm{f}}(s) = \ell_{\mathrm{s}\setminus\mathrm{f}}(s) = \lambda/s$ (4.37) becomes:

$$\begin{aligned} 0 &= \det\left(\frac{s+\lambda}{s}I + \mu\left(X^{\mathrm{T}}X\right)^{-1}\right) \\ &= \det\left(\frac{1}{s}\left(\lambda I + s\left(I + \mu\left(X^{\mathrm{T}}X\right)^{-1}\right)\right)\right) \\ &= \frac{1}{s^{n_{\mathrm{s}}}}\det\left(I + \mu\left(X^{\mathrm{T}}X\right)^{-1}\right)\det\left(\lambda\left(I + \mu\left(X^{\mathrm{T}}X\right)^{-1}\right)^{-1} + sI\right). \end{aligned} \tag{4.38}$$

The closed-loop poles are therefore a subset of the eigenvalues of $-\lambda\big(I+\mu(X^{\mathrm{T}}X)^{-1}\big)^{-1}$, which can obtained by substituting the SVD of X:

$$-\lambda\Big(I+\mu\big(X^{\mathrm{T}}X\big)^{-1}\Big)^{-1} = V_X\,\mathrm{diag}(-\lambda\frac{\sigma_i^2}{\sigma_i^2+\mu})V_X^{\mathrm{T}}. \tag{4.39}$$

where σ_i are the standard singular values of X. The poles are therefore $-\lambda\sigma_i^2/(\sigma_i^2+\mu)$, $i=1,\ldots,n_y$, and vary from $-\lambda$ for $\mu=0$ to 0 for $\mu\to\infty$ without crossing the real axis (Proposition 4.6). Note that by construction, the root locus of the two-array system with $\ell_{\mathrm{s}\cap\mathrm{f}}(s)=\ell_{\mathrm{s}\setminus\mathrm{f}}(s)=\ell(s)$ corresponds to the root locus of the single-array system.

Remark 4.7. The case $n_{\mathrm{y}}=n_{\mathrm{s}}=n_{\mathrm{f}}$ with $\mathrm{rank}(R_{\mathrm{s}})=\mathrm{rank}(R_{\mathrm{f}})=n_{\mathrm{y}}$ considerably simplifies the analysis. In this case, the input compensator Υ becomes redundant, and the open-loop (4.35) simplifies to $L(s)=\ell_{\mathrm{s}\cap\mathrm{f}}(s)XX_{\mu}^{-1}$. Using the standard SVD of X (4.9), the matrix $L(s)$ is further simplified to

$$L(s)=\ell_{\mathrm{s}\cap\mathrm{f}}(s)U\,\mathrm{diag}\left(\frac{\sigma_1^2}{\sigma_1^2+\mu},\ldots,\frac{\sigma_{n_{\mathrm{y}}}^2}{\sigma_{n_{\mathrm{y}}}^2+\mu}\right)U^{\mathrm{T}}, \tag{4.40}$$

where σ_i are the standard singular values of X. The open loop (4.40) corresponds to the open-loop transfer function of a single-array system designed using the procedure from Section 1.6. According to Lemma 4.3, the standard singular values equal those of $R=\begin{bmatrix}R_{\mathrm{s}} & R_{\mathrm{f}}\end{bmatrix}$. Hence, ignoring model uncertainty, the two-array approach yields the same closed-loop dynamics as a hypothetical single-array system with $n_{\mathrm{s}}+n_{\mathrm{f}}$ actuators, $g(s)=g_{\mathrm{f}}(s)$, and possibly with a permuted $R=\begin{bmatrix}R_{\mathrm{s}} & R_{\mathrm{f}}\end{bmatrix}$. Also, considering that R has a kernel of dimension n_{f}, this argument can be extended to single-array systems with n_{s} actuators and $\bar{R}\in\mathbb{R}^{n_{\mathrm{s}}\times n_{\mathrm{s}}}$ obtained from the economy-sized (standard) SVD of R.

4.3 Robust Stability

Suppose that the real plant is given by

$$\begin{aligned} P(s) &= \bar{P}(s)+\begin{bmatrix}\Delta_{\mathrm{s}}g_{\mathrm{s}}(s) & \Delta_{\mathrm{f}}g_{\mathrm{f}}(s)\end{bmatrix},\\ &=: \bar{P}(s)+\Delta(s), \end{aligned} \tag{4.41}$$

where $\Delta_\mathrm{s} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{s}}$ and $\Delta_\mathrm{f} \in \mathbb{R}^{n_\mathrm{y} \times n_\mathrm{f}}$ model the uncertainty. It is assumed that $g_\mathrm{s}(s)$ and $g_\mathrm{f}(s)$ reflect the actuator dynamics accurately or that any dynamic uncertainties occur at high frequencies where the controller has a small amplification.

In generalised modal space, the uncertain system is given by

$$\tilde{y}(s) = \left(\begin{bmatrix} \Sigma_\mathrm{s} & 0 \\ 0 & I \end{bmatrix} + \tilde{\Delta}_\mathrm{s}\right) g_\mathrm{s}(s)\tilde{u}_\mathrm{s}(s) + \left(\begin{bmatrix} \Sigma_\mathrm{f} \\ 0 \end{bmatrix} + \tilde{\Delta}_\mathrm{f}\right) g_\mathrm{f}(s)\tilde{u}_\mathrm{f}(s) + \tilde{d}(s), \tag{4.42}$$

where $\tilde{\Delta}_\mathrm{s} \coloneqq X^{-1}\Delta_\mathrm{s}U_\mathrm{s}$ and $\tilde{\Delta}_\mathrm{f} \coloneqq X^{-1}\Delta_\mathrm{f}U_\mathrm{f}$. In general, $\tilde{\Delta}_\mathrm{s}$ and $\tilde{\Delta}_\mathrm{f}$ are not diagonal and (4.42) shows that any uncertainty couples the modes in generalised modal space.

The IMC structure with uncertainty is shown in Fig. 4.3. For the robust stability analysis, the transfer function $M(s)$ from $u_\Delta(s)$ to $y_\Delta(s)$ (see Fig. 4.3) is

$$M(s) \coloneqq -Q(s)\Upsilon(I + (\Gamma - I)P(s)Q(s)\Upsilon)^{-1}\Gamma. \tag{4.43}$$

The system in Fig. 4.3 is stable iff [130, Thm. 8.1]

$$\det(I - M(\mathrm{j}\omega)\Delta(\mathrm{j}\omega)) \neq 0 \quad \forall\omega. \tag{4.44}$$

A sufficient condition for (4.44) is

$$\rho(M(\mathrm{j}\omega)\Delta(\mathrm{j}\omega)) = \rho(\Delta(\mathrm{j}\omega)M(\mathrm{j}\omega)) < 1 \quad \forall\omega, \tag{4.45}$$

where $\rho(\cdot)$ denotes the spectral radius. The product $\Delta(s)M(s)$ can be rearranged as

$$\Delta(s)M(s) = \begin{bmatrix}\Delta_\mathrm{s} & \Delta_\mathrm{f}\end{bmatrix}\begin{bmatrix} Ig_\mathrm{s}(s) & 0 \\ 0 & Ig_\mathrm{f}(s) \end{bmatrix} M(s). \tag{4.46}$$

For any square matrix A, the spectral radius can be upper bounded by $\rho(A) \leq \|A\|_2$ [68, Thm. 5.6.9]. A sufficient condition for robust stability is therefore

$$\left\|\begin{bmatrix}\Delta_\mathrm{s} & \Delta_\mathrm{f}\end{bmatrix}\right\|_2 \left\|\begin{bmatrix} Ig_\mathrm{s}(\mathrm{j}\omega) & 0 \\ 0 & Ig_\mathrm{f}(\mathrm{j}\omega) \end{bmatrix} M(\mathrm{j}\omega)\right\|_2 < 1 \quad \forall\omega, \tag{4.47}$$

from which an upper bound on the admissible uncertainty is obtained:

$$\left\|\begin{bmatrix}\Delta_\mathrm{s} & \Delta_\mathrm{f}\end{bmatrix}\right\|_2 < \Delta_\mathrm{max} \coloneqq \min_\omega \left(\left\|\begin{bmatrix} Ig_\mathrm{s}(\mathrm{j}\omega) & 0 \\ 0 & Ig_\mathrm{f}(\mathrm{j}\omega) \end{bmatrix} M(\mathrm{j}\omega)\right\|_2\right)^{-1}. \tag{4.48}$$

Whenever $\left\|\begin{bmatrix}\Delta_\mathrm{s} & \Delta_\mathrm{f}\end{bmatrix}\right\|_2 > \Delta_\mathrm{max}$, the control system of Fig. 4.3 *can* be unstable. The right-hand side of (4.48) can be plotted against frequency to find the smallest Δ_max.

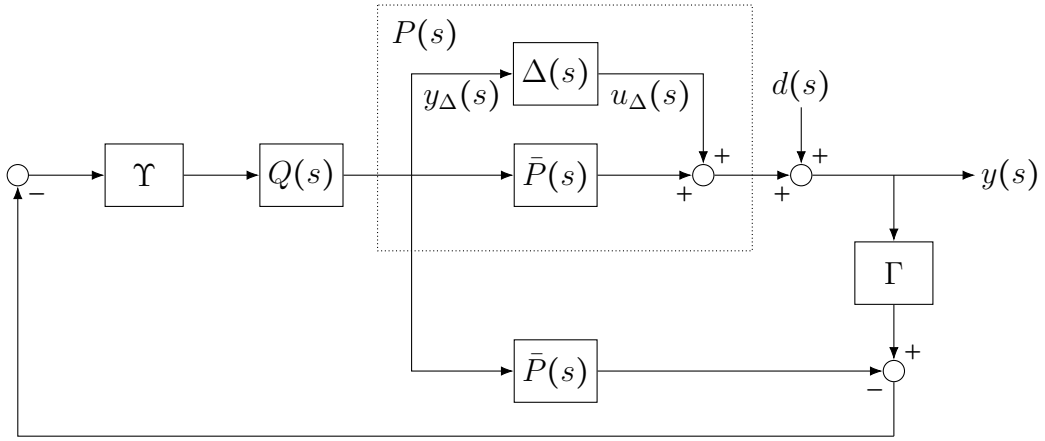


Figure 4.3: IMC structure with plant $P(s) = \bar{P}(s) + \Delta(s)$, where $\bar{P}(s)$ is known and $\Delta(s)$ models the unknown uncertainty.

4.4 Fast Orbit Feedback at Diamond Light Source

In parallel to the simulation studies for Diamond-II, the FOFB infrastructure of the existing Diamond storage ring has been upgraded with a more powerful and centralised computing node that reflects the technical setup of Diamond-II (as described in Chapter 7). The enhanced computing node allows new control algorithms to be tested, and the aim of this section is to apply the GSVD-based controller to a real-world system and evaluate it in practical circumstances. Since the Diamond control problem corresponds to a single-array system of the form (1.3), a subset of inputs and outputs is selected from (1.3) and represented as (4.1). Although $g_{\mathrm{s}}(s) = g_{\mathrm{f}}(s) = g(s)$ on the existing storage ring, a subset of the inputs is selected to represent slow actuators (and cover low frequencies), and the remaining inputs are selected to represent fast actuators (and cover high frequencies). After transforming (1.3) using the GSVD, the control effort is distributed onto designated slow and fast actuators using *mid-ranging* control [6], [53], which reflects the situation at Diamond-II. Note that both the Diamond and Diamond-II ORMs are ill-conditioned and are therefore comparable. The two-array controller is then benchmarked against a single-array controller, and the real-world results demonstrate that the GSVD is applicable in practice.

This section is organised as follows. Subsection 4.4.1 and Subsection 4.4.2 detail the design parameter of the single-array and two-array controllers, followed by the analysis of the disturbance in Section 4.4.3. Results from tests on the Diamond storage ring are presented in Section 4.4.4. If not stated otherwise, the figures and diagrams correspond to the vertical control direction direction.

4.4.1 Single-Array Controller

At Diamond, the single-array system (1.3) has $n_\mathrm{y} = 173$ BPMs and $n_\mathrm{u} = 172$ identical magnets and the fast orbit feedback is operated at 10 kHz. In practice, the synchrotron can be reconfigured, allowing any combination of $n_\mathrm{y} \leq 173$ and $n_\mathrm{u} \leq 172$ outputs and inputs. The actuator models for the horizontal and vertical control directions are

$$g(s) \coloneqq \frac{a}{s+a} \mathrm{e}^{-\tau_d s}, \tag{4.49}$$

where $a \coloneqq 2\pi \times 700\,\mathrm{rad\,s^{-1}}$ and $\tau_d \coloneqq 900\,\mu\mathrm{s}$. For the dynamic part of the controller (Section 1.6.2), the output sensitivity $S_\mathrm{m} : \mathbb{C} \mapsto \mathbb{C}$ is chosen to be

$$S_\mathrm{m}(s) = 1 - T_\mathrm{m}(s) = 1 - \frac{\lambda}{s+\lambda} \mathrm{e}^{-\tau_d s} =: 1 - T_\mathrm{m}(s), \tag{4.50}$$

where $\lambda \coloneqq 1/\tau_d = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$ is the closed-loop bandwidth, so that the IMC filter $q : \mathbb{C} \mapsto \mathbb{C}$ is given by

$$q(s) \coloneqq \frac{T_\mathrm{m}(s)}{g(s)} = \frac{\lambda}{a} \frac{s+a}{s+\lambda}. \tag{4.51}$$

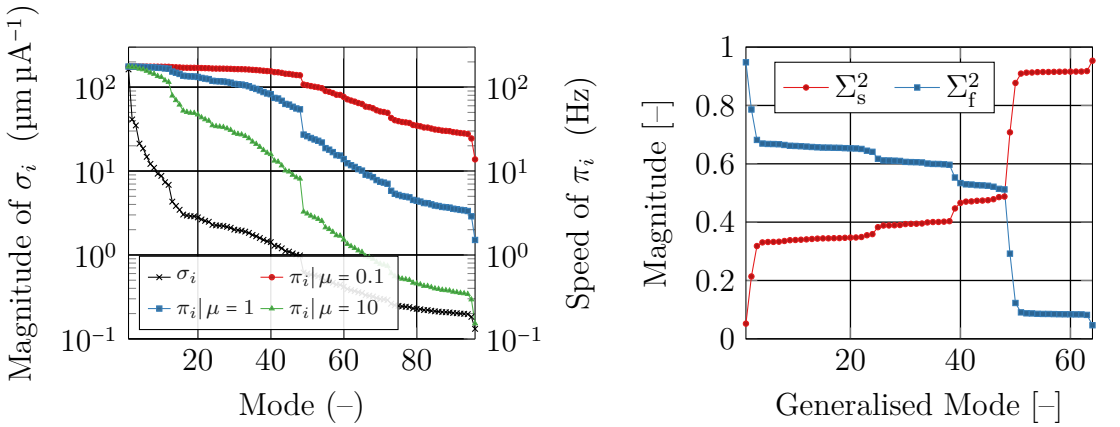
To account for the large condition number of R, the output compensator $\Gamma = (R^\mathrm{T} R + \mu I)^{-1} R^\mathrm{T}$ is embedded with $\mu = 1$ (Section 1.6.3), resulting in the overall control law

$$u(s) = -V \operatorname{diag}\left(\frac{\sigma_1}{1+\sigma_1^2}, \ldots, \frac{\sigma_{n_\mathrm{y}}}{1+\sigma_{n_\mathrm{y}}^2} \right) U^\mathrm{T} c(s) y(s), \tag{4.52}$$

where the standard feedback filter $c(s)$ is (Section 1.6.4)

$$c(s) \coloneqq \frac{\lambda}{a} \frac{s+a}{s+\lambda(1-\mathrm{e}^{-s\tau_d})}. \tag{4.53}$$

In practice, the continuous-time transfer function (4.53) is mapped to discrete-time using zero-order hold (Section 1.6) and implemented as a control law in



(a) Standard singular values σ_i and closed-loop poles π_i.

(b) Generalised singular values of R_s and R_f.

Figure 4.4: Standard and generalised singular values of the Diamond ORMs (vertical direction). Figure (a) also shows the mode-by-mode closed-loop poles for $\mu \in \{0.1, 1, 10\}$.

the form $u_t = -Kc(z^{-1})y_t$, where $K \in \mathbb{R}^{n_u \times n_y}$ is the gain matrix (1.24), z^{-1} is interpreted as the backwards shift operator and $t \in \mathcal{Z}_+$ represents the discrete-time variable. The controller $c(s)$ implements a pole at $s = 0$ and the inclusion of Γ effectively reduces the closed-loop bandwidth, which can be seen by computing the overall complementary sensitivity:

$$T(s) = U\mathrm{diag}\left(\frac{\sigma_1 k_1 \lambda \mathrm{e}^{-s\tau_d}}{s + \lambda(1-(1-\sigma_1 k_1)\mathrm{e}^{-s\tau_d})}, \ldots, \frac{\sigma_{n_y} k_{n_y} \lambda \mathrm{e}^{-s\tau_d}}{s + \lambda\big(1-(1-\sigma_{n_y} k_{n_y})\mathrm{e}^{-s\tau_d}\big)}\right) U^{\mathrm{T}}, \quad (4.54)$$

where $k_i := \sigma_i/(\mu + \sigma_i^2)$. Fig. 4.4a compares the standard singular values σ_i ($-\!\!\times\!\!-$, left-hand side axis), $i = 1, \ldots, n_y$, with the corresponding closed-loop poles π_i for $\mu \in \{0.1, 1, 10\}$ (right-hand side axis). For modes associated with $\sigma_i \gg \mu$, $\pi_i \approx \lambda = 2\pi 176\,\mathrm{rad\,s^{-1}}$, but as the singular values decrease, the closed-loop poles are slowed down, hence increasing the robustness for modes associated with small singular values. At Diamond, the choice $\mu = 1$ has proven to be effective. The controller structure would allow different values of μ to be chosen for different modes, but previous research has shown that $\mu = 1$ is near-optimal with respect to some robust performance criterion [54].

4.4.2 Two-Array Controller

To mimic the Diamond-II system, $n_\mathrm{s} = 96$ magnets are chosen to represent the slow actuators, while $n_\mathrm{f} = 64$ are chosen to represent the fast actuators. These $n_\mathrm{u} = n_\mathrm{s} + n_\mathrm{f} = 160$ magnets are controlling a selection of $n_\mathrm{y} = 96$ BPMs. The overall selection of magnets and BPMs is made based on physical arguments as well as on Assumption 4.1, which will hold for Diamond-II. The matrices R_s and R_f are obtained from extracting the corresponding rows and columns of the Diamond ORM R. To design the controller, the two-array system (4.1) is mapped to generalised modal space, which decouples (4.1), as in (4.7). The next paragraph explains the controller dynamics, which is followed by the design of input and output compensators.

Mid-Ranging Control

For the two-array controller, the output sensitivity $S_{\mathrm{s}\cap\mathrm{f}}(s) = 1 - T_{\mathrm{s}\cap\mathrm{f}}(s)$ of the TISO systems is chosen to be

$$S_{\mathrm{s}\cap\mathrm{f}}(s) = 1 - \frac{\lambda_{\mathrm{s}\cap\mathrm{f}}}{s + \lambda_{\mathrm{s}\cap\mathrm{f}}} \mathrm{e}^{-\tau_d s}, \tag{4.55}$$

where for later comparison, $\lambda_{\mathrm{s}\cap\mathrm{f}} := 1/\tau_d = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$ matches the closed-loop bandwidth of the existing single-array design (4.50). The output sensitivity of the SISO systems $S_{\mathrm{s}\setminus\mathrm{f}}(s) = 1 - T_{\mathrm{s}\setminus\mathrm{f}}(s)$ is chosen to be

$$S_{\mathrm{s}\setminus\mathrm{f}}(s) := 1 - \frac{\lambda_{\mathrm{s}\setminus\mathrm{f}}}{s + \lambda_{\mathrm{s}\setminus\mathrm{f}}} \mathrm{e}^{-\tau_d s}, \tag{4.56}$$

where some of the following experiments use $\lambda_{\mathrm{s}\setminus\mathrm{f}} := 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ and some $\lambda_{\mathrm{s}\setminus\mathrm{f}} := 2\pi \times 10\,\mathrm{rad\,s^{-1}}$. With $S_{\mathrm{s}\cap\mathrm{f}}(s)$ and $S_{\mathrm{s}\setminus\mathrm{f}}(s)$ fixed, the IMC filter for the $u_\mathrm{s}(s)$ array is

$$q_\mathrm{s}(s) := \frac{T_{\mathrm{s}\setminus\mathrm{f}}(s)}{g_\mathrm{s}(s)} = \frac{\lambda_{\mathrm{s}\setminus\mathrm{f}}}{a} \frac{s + a}{s + \lambda_{\mathrm{s}\setminus\mathrm{f}}}, \tag{4.57}$$

and the filter for the $u_\mathrm{f}(s)$ array is

$$q_\mathrm{f}(s) := \frac{T_{\mathrm{s}\cap\mathrm{f}}(s) - T_{\mathrm{s}\setminus\mathrm{f}}(s)}{g_\mathrm{f}(s)} = \frac{\lambda_{\mathrm{s}\cap\mathrm{f}} - \lambda_{\mathrm{s}\setminus\mathrm{f}}}{a} \frac{s(s + a)}{(s + \lambda_{\mathrm{s}\cap\mathrm{f}})(s + \lambda_{\mathrm{s}\setminus\mathrm{f}})}. \tag{4.58}$$

The choices (4.55)–(4.58) are also referred to as *mid-ranging* TISO controllers [6], [53]. The overall bandwidth is split between the slow and fast actuator arrays, which will allow for a higher closed-loop bandwidth at Diamond-II.

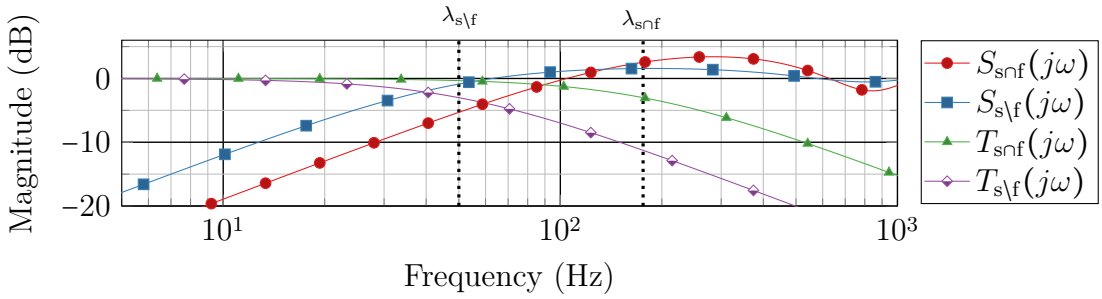


Figure 4.5: Bode plots of output sensitivities $S_{(.)}(s)$ from (4.55)–(4.56) for the decoupled TISO ($\lambda_{s\cap f} = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$) and SISO ($\lambda_{s\setminus f} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$) systems. The figure also shows complementary sensitivities $T_{(.)}(s) = 1 - S_{(.)}(s)$.

Fig. 4.5 shows the Bode magnitude plots of the output sensitivities, $S_{s\cap f}(\mathrm{j}\omega)$ (—■—) and $S_{s\setminus f}(\mathrm{j}\omega)$ (—●—), and the corresponding complementary sensitivities, $T_{s\cap f}(\mathrm{j}\omega)$ (—▲—) and $T_{s\setminus f}(\mathrm{j}\omega)$ (—◆—) for $\lambda_{s\cap f} = 2\pi\times176\,\mathrm{rad\,s^{-1}}$ and $\lambda_{s\setminus f} = 2\pi\times50\,\mathrm{rad\,s^{-1}}$. Due to the large time delay, the phase lag of $T_{s\cap f}(\mathrm{j}\omega)$ reaches 60° at 100 Hz, which significantly reduces the bandwidth of $S_{s\cap f}(\mathrm{j}\omega)$. The sensitivity peaks are $\|S_{s\cap f}(\mathrm{j}\omega)\|_\infty = 3.4\,\mathrm{dB}$ at 290 Hz and $\|S_{s\setminus f}(\mathrm{j}\omega)\|_\infty = 0.4\,\mathrm{dB}$ at 113 Hz. In the following section, the scalar filters $q_s(s)$ and $q_f(s)$ are embedded in the MIMO system.

Input and Output Compensators

The standard singular values of R are shown in Fig. 4.4a and the squared generalised singular values of R_s and R_f in Fig. 4.4b. The overall condition number for the two-array system is $\kappa(R) = \kappa(X) = 1159$, but the corresponding condition numbers for the generalised singular values are $\kappa(\Sigma_s) = 4.3$ and $\kappa(\Sigma_f) = 4.5$, and therefore allow Σ_s and Σ_f in (4.14) to be inverted.

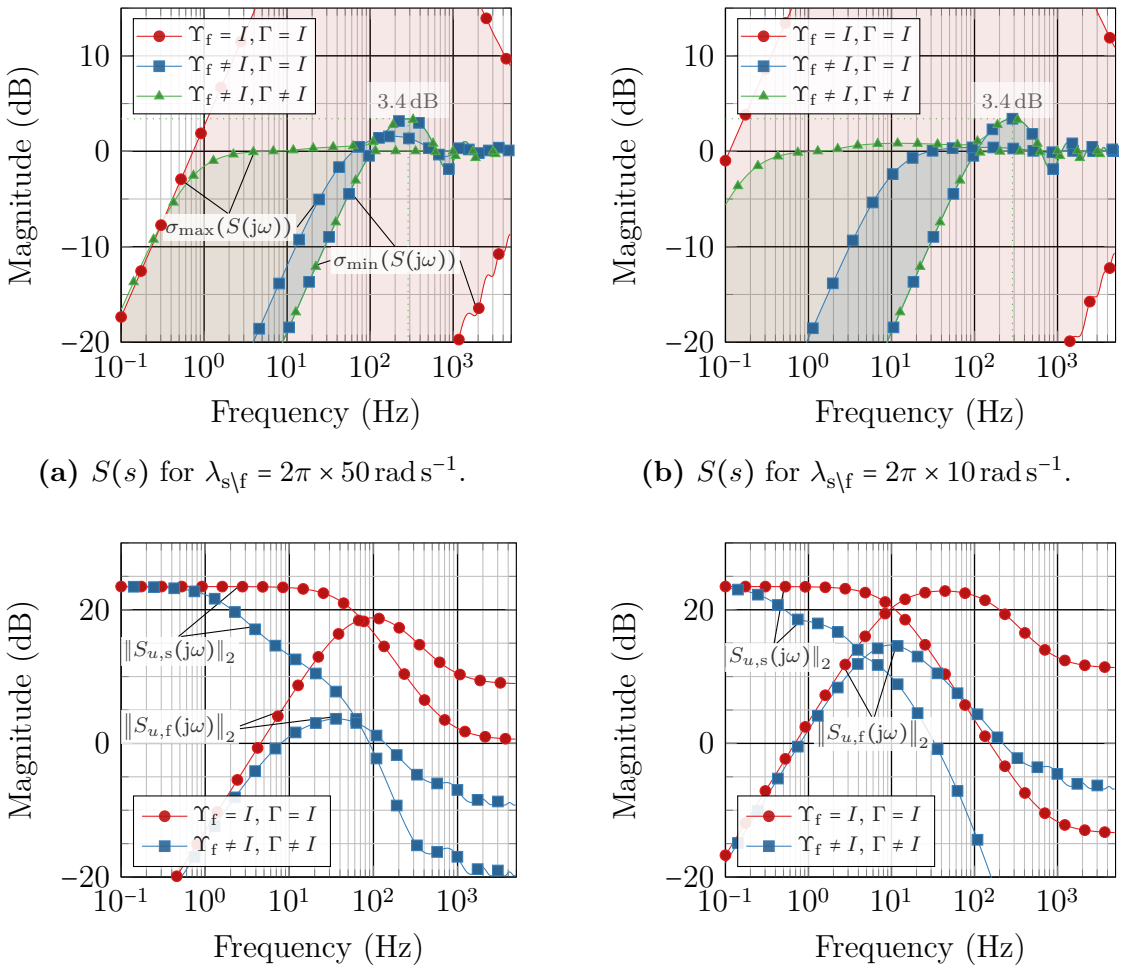
Fig. 4.6a shows the minimum and maximum gains of the output sensitivity (4.18) (—●—), $\sigma_{\min}(S(\mathrm{j}\omega))$ and $\sigma_{\max}(S(\mathrm{j}\omega))$, for $\lambda_{s\setminus f} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ and for the case that the scalar filters from Section 4.4.2 are embedded in the MIMO system without input and output compensators ($\Upsilon = \begin{bmatrix} I & I \end{bmatrix}^{\mathrm{T}}$ and $\Gamma = I$). For orthogonal X, the magnitude of the sensitivity would be enclosed by the TISO and SISO transfer functions from Fig. 4.5, but with the ill-conditioned X and $n_f < n_s$, some disturbance directions are amplified for for frequencies ranging from 1 Hz to 5 kHz. The input compensator $\Upsilon = \begin{bmatrix} I & \Upsilon_f^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ from Section 4.2.1 removes the performance difference

between generalised modal space and original space, so that the "input-compensated" sensitivity (—■— , Fig. 4.6a) equals the sensitivity from Fig. 4.5.

The output compensator Γ is designed according to Section 4.2.1 with the choice $\mu = 1$, which is adopted from the single-array design from Section 4.4.1. Analogous to the single-array design, the regularisation has the effect of reducing the closed-loop bandwidth, which can be seen from the "input- and output-compensated" sensitivity in Fig. 4.6a (—▲—).

The analysis of the output sensitivity gains is repeated in Fig. 4.6b for $\lambda_{\mathrm{s\backslash f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$, meaning that the closed-loop bandwidth of the slow actuator array is reduced by 40 Hz. Compared to Fig. 4.6a, the bandwidth of the maximum sensitivity gain $\sigma_{\max}(S(\mathrm{j}\omega))$ is reduced accordingly, while the bandwidth of $\sigma_{\min}(S(\mathrm{j}\omega))$, which is determined by the TISO systems, remains the same. Fig. 4.6b also highlights that for the "input- and output-compensated" sensitivity (—▲—), some disturbance directions are amplified between 1 Hz and 100 Hz with a local peak of roughly 1 dB at 10 Hz, which corresponds to the transition between slow and fast actuators in mid-ranging control. This peak does not appear for the "uncompensated" sensitivity (—■—) and is less pronounced in Fig. 4.6a, and therefore likely to be associated with the additonal phase lag introduced by the slow actuator array for $\lambda_{\mathrm{s\backslash f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$.

The effect of the regularisation on the inputs is illustrated in Fig. 4.6c and 4.6d, which show the maximum gain of the transfer functions from $d(s)$ to $u_{(\cdot)}(s)$, $S_{u,(\cdot)}(\mathrm{j}\omega)$, for $(\cdot) \in \{\mathrm{s}, \mathrm{f}\}$. Without compensators (—●—), the control effort is sustained up to 100 Hz. Moreover, reducing the SISO bandwidth, such as in Fig. 4.6b and 4.6d, increases the control action of the fast actuator array, which is a consequence of the mid-ranging approach. With compensators (—■—), the controller gain is reduced by 20 dB at 100 Hz for both actuator arrays in Fig. 4.6c and 4.6d. Increasing the regularisation parameter μ would have the effect of decreasing the gains further.



(a) $S(s)$ for $\lambda_{s\setminus f} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$.

(b) $S(s)$ for $\lambda_{s\setminus f} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$.

(c) $S_{u,(\cdot)}(s)$ for $\lambda_{s\setminus f} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$.

(d) $S_{u,(\cdot)}(s)$ for $\lambda_{s\setminus f} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$.

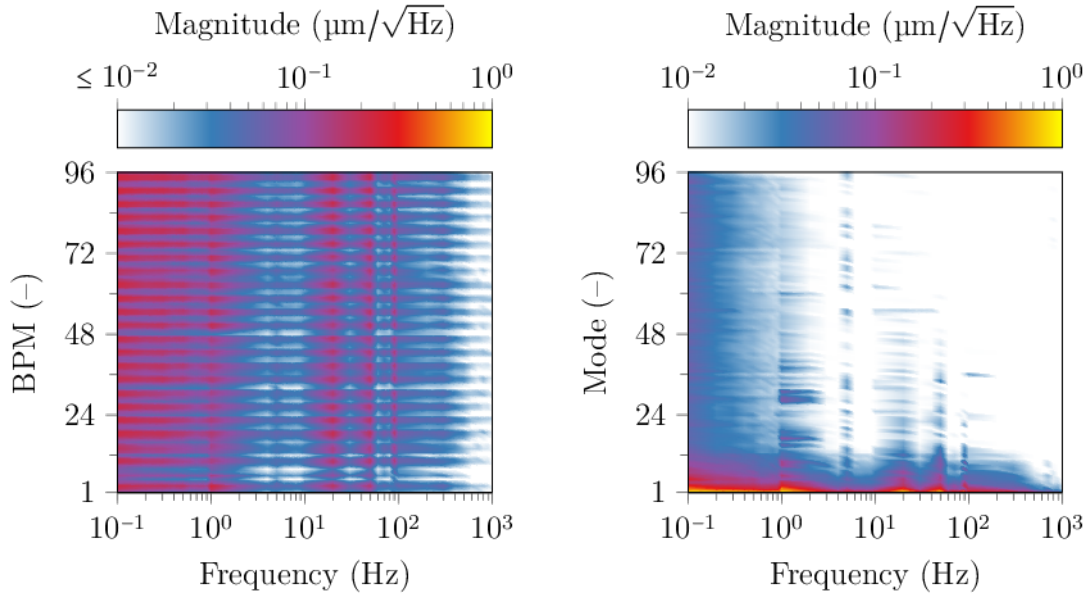
Figure 4.6: Minimum and maximum singular values $S(s)$ ($d(s) \mapsto y(s)$) and $S_{u,(\cdot)}(s)$ ($d(s) \mapsto u_{(\cdot)}(s)$) for $\lambda_{s\cap f} = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$, $\lambda_{s\setminus f} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ (a and c), $\lambda_{s\setminus f} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$ (b and d), and different compensators (vertical direction).

4.4.3 Disturbance Spectrum

For both single- and two-array models, all exogenous effects are lumped into the (output) disturbance $d(s)$. To understand the characteristic disturbance spectrum at Diamond, $d(s)$ can be split into an input disturbance $d_u(s)$ and an output disturbance $d_y(s)$:

$$d(s) = Rd_u(s) + d_y(s). \tag{4.59}$$

The contribution from $d_u(s)$ is mainly associated with ground vibrations and vibrating machine components. The vibrations are transmitted to the corrector



(a) Original space: min. = $10^{-2.6}$, max. = $10^{-0.6}$ µm/$\sqrt{\text{Hz}}$.

(b) Mode space: min. = $10^{-2.5}$, max. = $10^{0.1}$ µm/$\sqrt{\text{Hz}}$.

Figure 4.7: Measured ASD of the disturbance in original and mode space (vertical direction).

magnets by the supporting girders, which exhibit structural resonances at particular frequencies [9], [47]. The corrector magnets eventually transmit these vibrations to the electron beam producing a disturbance component that is proportional to R. The contribution from $d_y(s)$ is mainly associated with the operation of beamlines, which can introduce (slow) ramp-shaped beam movement, and with injection and insertion devices [164, Ch. 16.7].

The (output) disturbance spectrum can be estimated when the feedback is disabled, i.e. when $y_t = d_t$. Fig. 4.7a shows the *amplitude spectral density* (ASD) for BPMs $i = 1, \ldots, 96$ from 0.1 Hz to 1 kHz, which is computed from the discrete Fourier transform [96, Ch. 2.2] of the measured signal d_t, $t = 0, \ldots, N-1$, as

$$D_{(i)}(\omega_k) := \sqrt{\frac{2}{f_s}\left|\sum_{t=0}^{N-1} d_{(i),t} \mathrm{e}^{-\frac{\mathrm{j}2\pi kt}{N}}\right|^2}, \qquad \omega_k := \frac{2\pi k f_s}{N}, \qquad k = 0, \ldots, N-1, \tag{4.60}$$

where $f_s = 10\,\text{kHz}$ and $N = 100,000$, which corresponds to 1 s of data and results in a frequency resolution of $f_s/N = 0.1\,\text{Hz}$. In practice, the ASD (4.60) is computed 10 times for a signal of length $10N$ and then averaged using Welch's method [96, Ch.

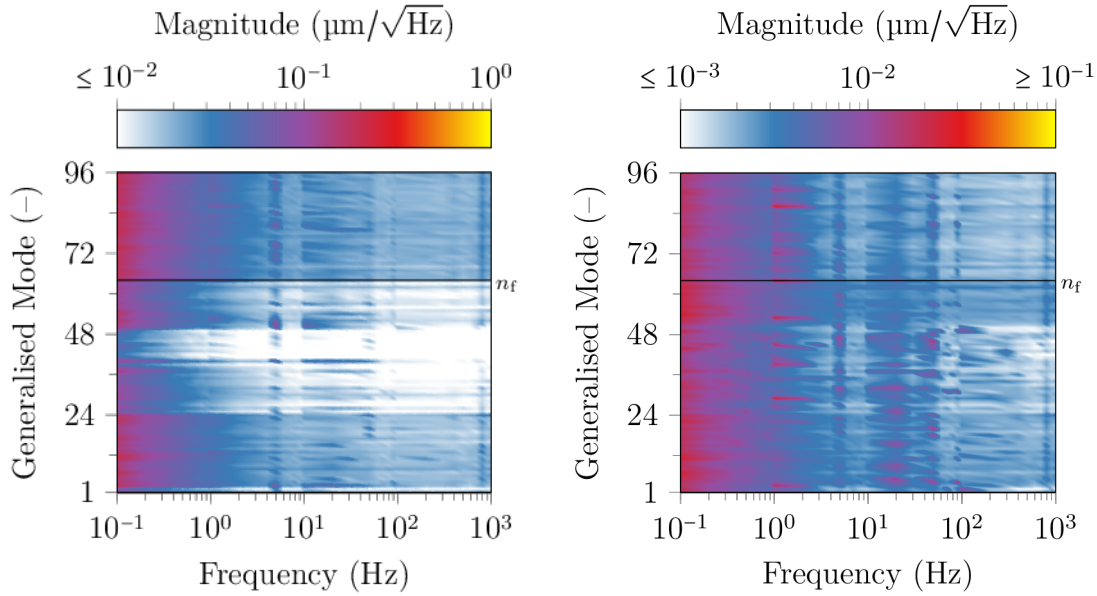
6.4]. The horizontal pattern of the ASD in Fig. 4.7a can be associated with girder resonances [9], such as the peaks at 0.2 Hz, 1 Hz, 20 Hz and 120 Hz. The vertical pattern is associated with the partitioning of the storage ring, which possesses an (approximate) 24-fold circulant symmetry [54]. Within each cell, the placement of BPMs on the girders and the distance to other devices determine the sensitivity to disturbances; some BPMs, such as those located downstream of an insertion device, are exposed to larger disturbances than other BPMs.

To demonstrate the importance of (4.59), the disturbance is mapped to mode space using the orthonormal matrix (1.8). The resulting amplitude spectral density, $\hat{D}_i(\omega)$, is shown in Fig. 4.7b, where the vertical axis refers to the ith mode with $i = 1$ being associated with the largest (standard) singular value of R. Due to the orthonormal property of the transformation matrix U, it holds that

$$\|D(\omega_k)\|_2^2 = \|\hat{D}(\omega_k)\|_2^2 \quad \forall \omega_k, \tag{4.61}$$

where the square in (4.61) is applied element-wise. Compared to Fig. 4.7a, the circulant pattern disappeared and the ASD is concentrated in the low-order modes ($i \leq 10$) instead. For the single-array system, the concentration of the disturbance in the low-order modes justifies the output compensator from Section 4.2.2, which significantly reduces the bandwidth for higher-order modes. Because the disturbance is concentrated in the low-order modes on which the regularisation has little effect, it is also expected that the realised output sensitivity will lie closer to the maximum disturbance attenuation (lower —▲— in Fig. 4.6) than to the minimum disturbance attenuation (upper —▲— in Fig. 4.6).

For the two-array case, the ill-conditioned X makes analysing the effect of the characteristic disturbance spectrum onto the performance more difficult. Fig. 4.8a shows the ASD of the disturbance in generalised modal space, $\tilde{D}_i(\omega)$, obtained from mapping the disturbance through (4.6). The vertical axis refers to the generalised modes and the TISO modes are those associated with $i = 1, \ldots, 64$, which is marked by the horizontal line in Fig. 4.8. Even though the transformation matrix X is not orthogonal and hence $\tilde{D}(\omega) \neq D(\omega)$, Fig. 4.8a shows how the disturbance is



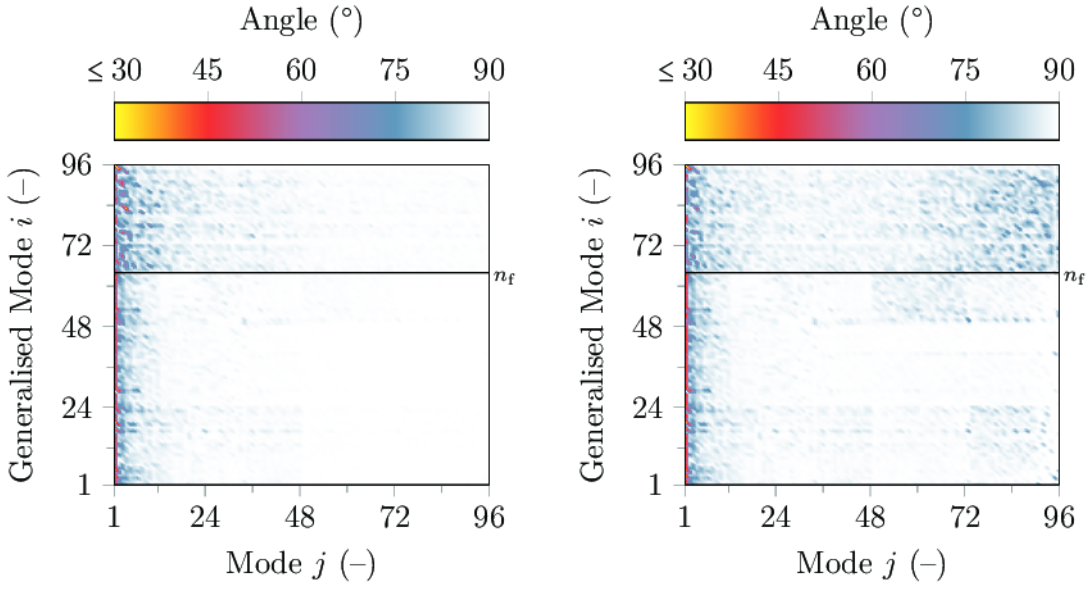
(a) Without output compensator ($\Gamma = I$): min. = $10^{-2.5}$, max. = $10^{-0.6}$ µm/$\sqrt{\text{Hz}}$.

(b) With output compensator ($\Gamma \neq I$): min. = $10^{-3.1}$, max. = $10^{-1.5}$ µm/$\sqrt{\text{Hz}}$.

Figure 4.8: Measured ASD of the disturbance at Diamond mapped to generalised modal space with (a) and without (b) output compensator. The horizontal line at $n_\text{f} = 64$ separates the TISO from the SISO modes.

distributed on SISO and TISO modes. Analogous to the mode space ASD from Fig. 4.7b, the vertical pattern of Fig. 4.7a disappeared, but the disturbance is spread onto TISO as well as SISO modes. While the ASD of some TISO modes (e.g. $i = 1, \ldots, 24$) resembles those of the SISO modes ($i = 65, \ldots, 96$), the ASD of other TISO modes rapidly decreases for increasing frequency (e.g. $i = 40, \ldots, 48$).

To connect Fig. 4.7b and 4.8a, consider computing the acute angles between x_i, the columns of X, and U_j, the standard left singular vectors of R, for $i, j = 1, \ldots, n_y$. An angle equal to 90° means that standard mode j does *not* contribute to generalised mode i, whereas 0° means mode j is parallel to generalised mode i, but its particular weight depends on X. The angles are computed as $\operatorname{acos}(|x_i^\text{T} U_j|/\|x_i\|_2)$ and shown in Fig. 4.9a, where the horizontal axis refers to the ith standard mode and the vertical axis to the jth generalised mode. Fig. 4.9a shows that most generalised modes form an angle of less than 30° with the first mode, i.e. the vectors x_i are arranged as a cone of (linearly independent) vectors that is "centred" around U_1. In fact, the first standard left singular vectors of R_s and R_f form an angle of 0.5°



(a) Angle as $\operatorname{acos}(|x_i^{\mathrm{T}} U_j| / \|x_i\|_2)$: min. = 13°, max. = 90°.

(b) Angle as $\operatorname{acos}(|x_{\mu,i}^{\mathrm{T}} U_j| / \|x_{\mu,i}\|_2)$ min. = 13°, max. = 90°.

Figure 4.9: Acute angles between columns of X and U (a) and X_μ and U (b). The horizontal line separates TISO from SISO modes.

($\|R_{\mathrm{s}}\|_2 = 103$ and $\|R_{\mathrm{f}}\|_2 = 112$), and it turns out that centring the generalised singular vectors x_i around U_1 yields stationary points for the function $f(y)$ from Lemma 4.4. Fig. 4.9a also shows that the higher-order standard modes $i = 48, \ldots, 96$ are almost orthogonal to the generalised modes; disturbances aligned to these standard modes require larger gains from all generalised modes when multiplied by X^{-1}, which is readily explained through Lemma 4.3.

4.4.4 Results from the Storage Ring

To validate the controller design for Diamond-II, two versions of the two-array controller – one with $\lambda_{\mathrm{s}\setminus\mathrm{f}} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ and one with $\lambda_{\mathrm{s}\setminus\mathrm{f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$ – have been implemented on a customised IT setup (Chapter 7) and tested on the Diamond synchrotron. The control systems are implemented in discrete time using the structure detailed in Appendix 4.B and are capable of producing corrector setpoints at a rate of > 10 kHz.

The two-array controllers are tuned to a TISO bandwidth $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$ and are compared against a single-array controller with $\lambda_{\mathrm{m}} = \lambda_{\mathrm{s}\cap\mathrm{f}}$ that controls

the same $n_y = 96$ BPMs as the two-array controllers. The single-array controller uses $n_u = n_\mathrm{s} = 96$ correctors that are also used by the two-array controllers as slow correctors, and the two-array controllers use an additional array of $n_\mathrm{f} = 64$ fast correctors. The following results were obtained using a nominal beam current of 300 mA and with all other feedbacks from Table 1.1 disabled.

Outputs

Fig. 4.10 shows the output ASD measured in the first cell of the Diamond storage ring for disabled feedback (——), for the single-array controller (-·-·-) and for the two-array controllers with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi\times 50\,\mathrm{rad\,s^{-1}}$ (- - -) and $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi\times 10\,\mathrm{rad\,s^{-1}}$ (·····). The left-hand side of Fig. 4.10 shows the horizontal direction, the right-hand side the vertical direction, and the first to fourth rows corresponds to BPMs 1, 3, 5 and 7.

To interpret the performance of the single-array controller, consider the Bode magnitude diagram from Fig. 4.5, which shows the output sensitivity of the TISO and SISO systems of the two-array controller. Because the TISO systems are tuned to match the performance of the single-array controller, the TISO output sensitivity from Fig. 4.5 (-·•·-) corresponds to the expected single-array sensitivity *before* including the output compensator Γ. According to Fig. 4.5, an attenuation of 20 dB = 0.1 and 40 dB = 0.01 is expected at 10 Hz and 1 Hz for disturbances that are aligned to low-order modes of R. In Fig. 4.10, the same attenuation can be seen for the horizontal direction of BPM 1, where the disturbance ASD is attenuated by 20 dB and 40 dB at 10 Hz and 1 Hz. However, the attenuation is worse for other BPMs, in particular those for which the ASD is small for disabled feedback.

As expected from the controller design, the two-array controllers perform worse than the single-array controller, because fewer correctors cover the 176 Hz bandwidth. For frequencies between above 20 Hz, the performance of the two-array controllers is comparable to the performance of the single-array controller, which suggests that the disturbances are aligned to directions that correspond to the maximum attenuation in Fig. 4.6a and 4.6b. Indeed, the disturbance peaks between 20 Hz to 100 Hz are associated with girder eigenfrequencies and harmonics [9], which are

proportional to the term $R\,d_u(s)$ in (4.59) and therefore particularly pronounced in the direction of the low-order modes. For frequencies below 20 Hz, the two-array controllers perform worse than the single-array controller, but according to Fig. 4.6a and 4.6b, remain within the theoretical expectations.

Another performance measure is given by IBM in Fig. 4.11, which is computed from the ASD (4.60) as

$$\mathrm{IBM}_{(i)}(\omega_p) = \sqrt{\Delta f \sum_{k=1}^{p} (D_{(i)}(\omega_k))^2}, \tag{4.62}$$

where $\Delta f := f_s/N = 0.01\,\mathrm{Hz}$. The IBM has the effect of smoothing out the ASD, and compared to Fig. 4.10, the performance difference between the single-array and the two-array controllers largely disappeared. For high frequencies, the two-array controller with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ performs slightly worse ($\approx 100\,\mathrm{nm}$) than the one with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$, which is due to an increase in IBM between 50 Hz and 60 Hz.

Inputs

The main reason for augmenting the single-array system (1.3) with an additional array of actuators is to split the control effort onto two different kinds of corrector magnets: slow but strong magnets that cover low frequencies where the magnitude of the disturbance spectrum is large, and fast but weaker magnets that cover high frequencies where the magnitude of the disturbance spectrum is smaller.

Fig. 4.12 shows the ASD of the inputs ($\mathrm{A}/\sqrt{\mathrm{Hz}}$) for the experiments from Fig. 4.10. The first row of Fig. 4.12 corresponds to the two-array controller with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$, the second row to the two-array controller with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$ and the third row to the single-array controller. For the two-array controllers, the first column of Fig. 4.12 corresponds to the slow actuators and the second column to the fast actuators.

As expected from the mid-ranging approach, the ASD of the slow correctors is large at low frequencies and rapidly decreases between 10 Hz and 100 Hz. Comparing Fig. 4.12a with the theoretical transfer functions from $d(s)$ to $u_{(.)}(s)$ from Fig. 4.6c,

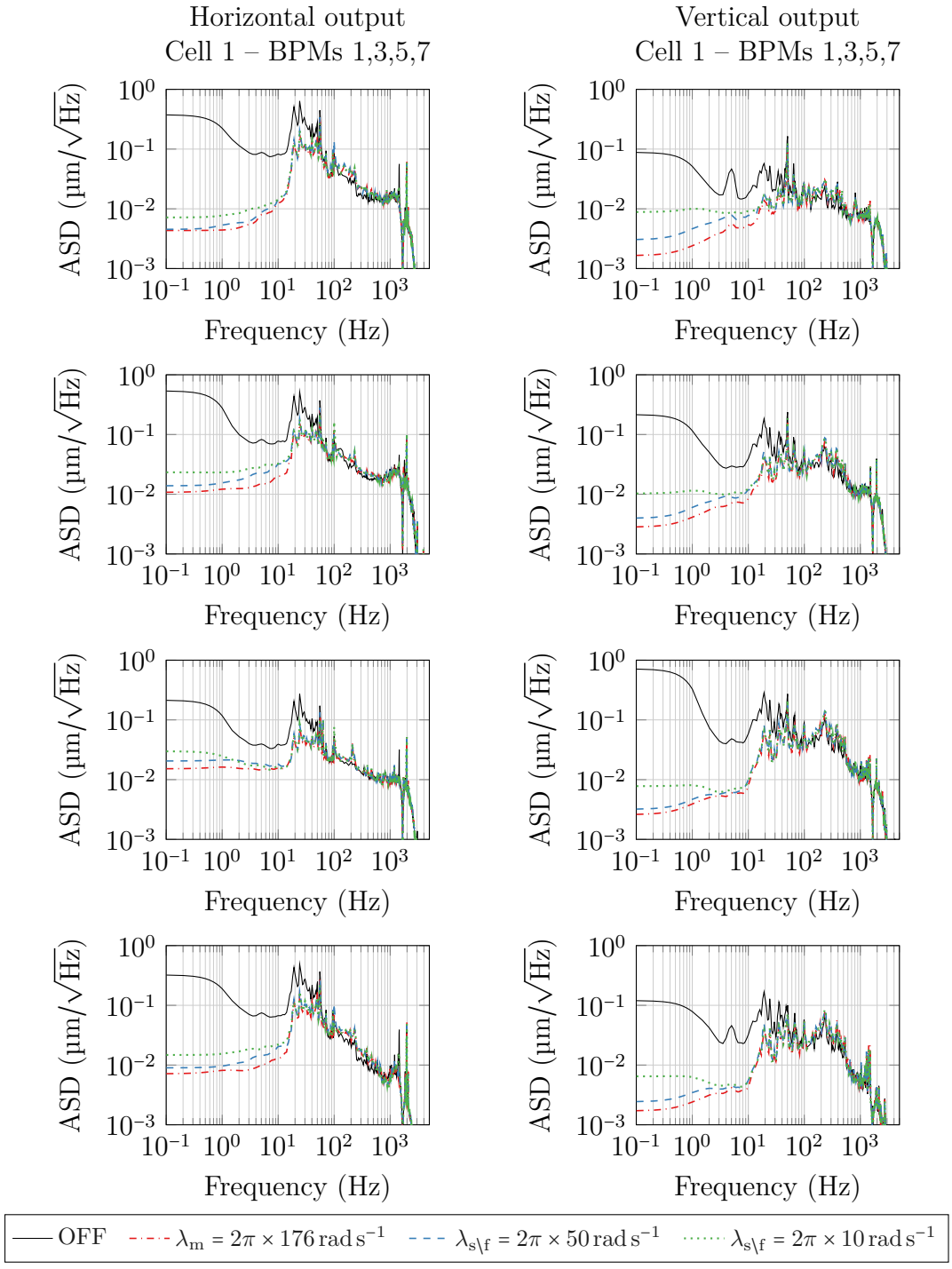


Figure 4.10: Measured output ASD in the first cell of the Diamond storage ring for disabled feedback (OFF), single-array controller (λ_m) from Section 4.4.1 and two differently tuned two-array controllers ($\lambda_{s\backslash f}$) from Section 4.4.2. The ASDs are computed using Welch's method and using 100 s of data sampled at 10 kHz.

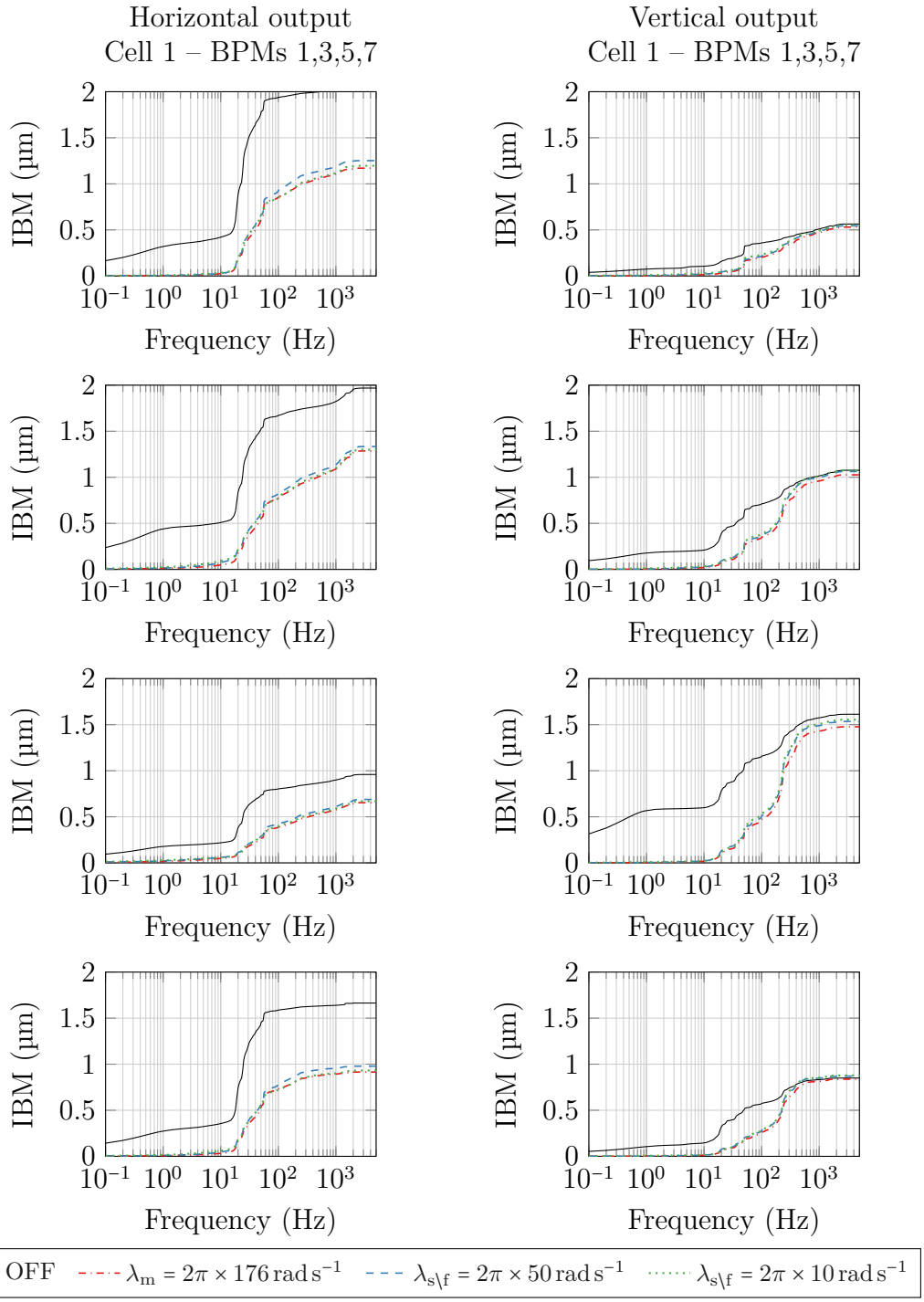


Figure 4.11: Measured IBM in the first cell of the Diamond storage ring for disabled feedback (OFF), single-array controller (λ_m) from Section 4.4.1 and two differently tuned two-array controllers ($\lambda_{s\backslash f}$) from Section 4.4.2.

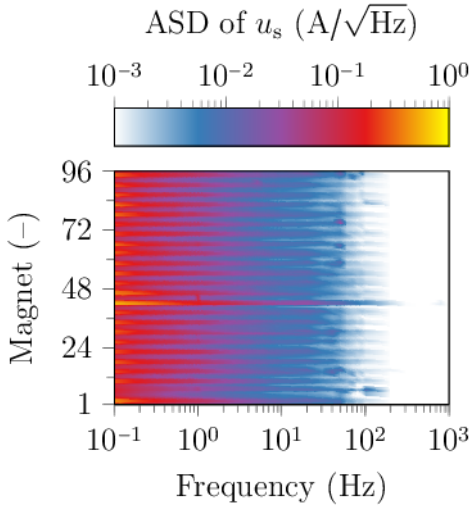
the input gain decreases from roughly $0.1\,\mathrm{A}/\sqrt{\mathrm{Hz}}$ (red) at low frequencies to roughly $0.01\,\mathrm{A}/\sqrt{\mathrm{Hz}}$ in the 10 Hz to 100 Hz range, which matches the theoretical prediction. Comparing the first row of Fig. 4.12 with the second row of Fig. 4.12, it can be seen how lowering the SISO bandwidth from $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ to $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$ increases the control effort of the fast actuator array. Comparing with the theoretical magnitude of $S_{u,(\cdot)}(s)$ from Fig. 4.6c and 4.6d, it can be seen that at 10 Hz, the gain of the two-array controller with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 50\,\mathrm{rad\,s^{-1}}$ is 10 dB ≈ 0.3 lower than the gain of the two-array controller with $\lambda_{\mathrm{s}\cap\mathrm{f}} = 2\pi \times 10\,\mathrm{rad\,s^{-1}}$, which is reflected in Fig. 4.12b and 4.12d.

The slow array of the two-array controllers can also be compared to the single-array controller (Fig. 4.12e), but given that the disturbance spectrum is concentrated in frequencies lower than 10 Hz, the difference is not very pronounced. For the single-array controller, it can be seen that a strong control effort is sustained up to 10 Hz, whereas the control effort of slow array of the two-array controllers decreases at lower frequencies. For frequencies above 100 Hz, the difference between the slow arrays of the two-array controllers and the single-array controllers is evident.

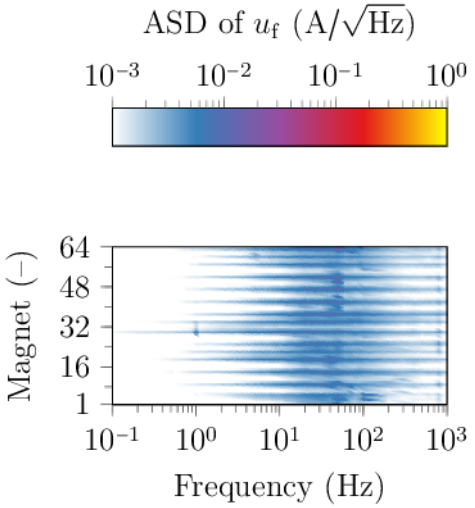
4.5 Controller Design for Diamond-II

As part of the FOFB design for Diamond-II, the next-generation upgrade of Diamond Light Source, the GSVD-based control approach of this chapter was adopted in the technical design report [2, Ch. 2.11.7]. Using Diamond-II ORMs and preliminary models of the slow and fast actuator dynamics [86], a GSVD-based controller was designed for Diamond-II and combined with estimates of the disturbance spectra [101] in simulations to demonstrate that the Diamond-II performance criteria are met. This also provides additional specifications for the design of the fast corrector magnets. This section summarises the application of the two-array controller to the preliminary Diamond-II model with $n_y = 252$ BPMs, $n_{\mathrm{s}} = 252$ slow and $n_{\mathrm{f}} = 144$ fast corrector magnets.

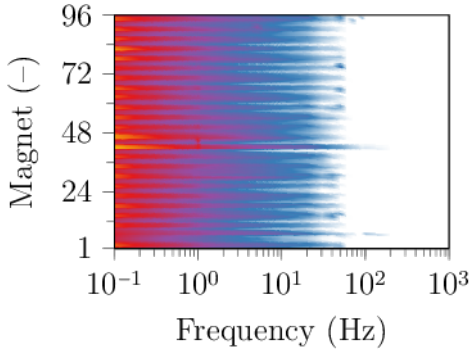
The actuator dynamics are obtained from a low-order approximation of a composite transfer function that considers several subsystems [86, App. B]. The



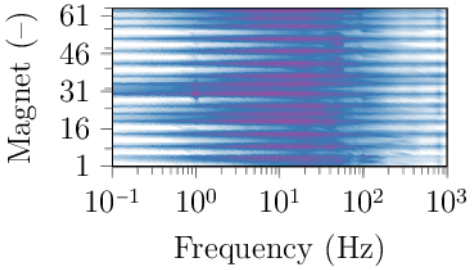
(a) Min.= $10^{-4.1}$, max.= $10^{-0.2}$ A/$\sqrt{\text{Hz}}$.



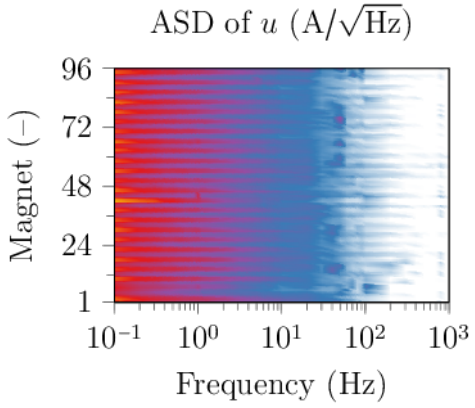
(b) Min.= $10^{-4.0}$, max.= $10^{-1.7}$ A/$\sqrt{\text{Hz}}$.



(c) Min.= $10^{-4.7}$, max.= $10^{-0.2}$ A/$\sqrt{\text{Hz}}$.

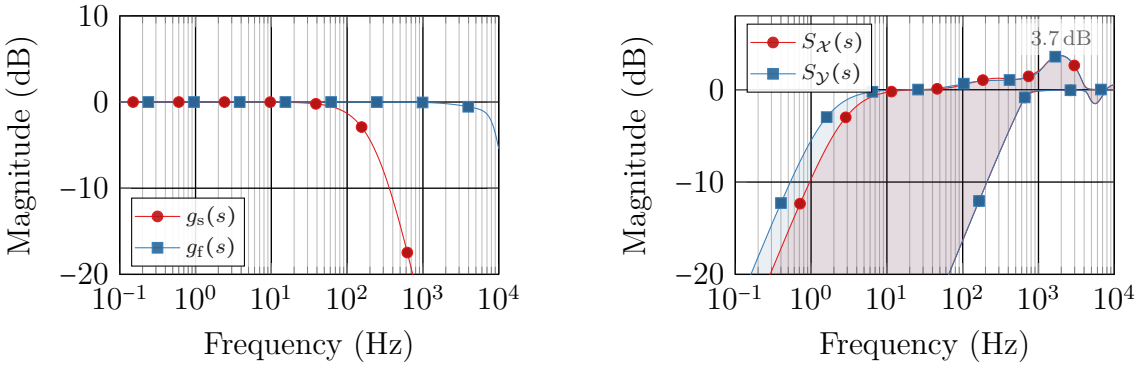


(d) Min.= $10^{-3.4}$, max.= $10^{-1.3}$ A/$\sqrt{\text{Hz}}$.



(e) Min.= $10^{-3.6}$, max.= $10^{-0.3}$ A/$\sqrt{\text{Hz}}$.

Figure 4.12: Measured ASD of inputs. The first row shows a two-array controller with $\lambda_{s\setminus f} = 2\pi \times 50\,\text{rad}\,\text{s}^{-1}$ and $\lambda_{s\cap f} = 2\pi \times 176\,\text{rad}\,\text{s}^{-1}$, the second row one with $\lambda_{s\setminus f} = 2\pi \times 10\,\text{Hz}$ and $\lambda_{s\cap f} = 2\pi \times 176\,\text{rad}\,\text{s}^{-1}$, and the third row a single-array controller with $\lambda_{s\cap f} = 2\pi \times 176\,\text{rad}\,\text{s}^{-1}$.



(a) Acuator dynamics. **(b)** Output sensitivity.

Figure 4.13: Preliminary actuator models for Diamond-II (a) and minimum and maximum output sensitivity gains for the horizontal and vertical direction (b).

resulting slow and fast dynamics are shown in Fig. 4.13a and given by

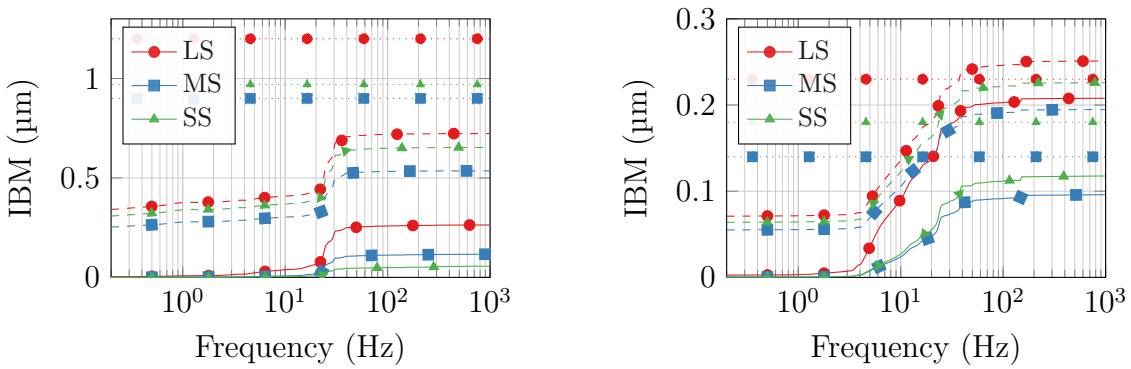
$$g_{\mathrm{s}}(s) = \frac{\omega_{n,\mathrm{s}}^2}{s^2 + 2\zeta_{\mathrm{s}}\omega_{n,\mathrm{s}}s + \omega_{n,\mathrm{s}}^2}\mathrm{e}^{-\tau_d s}, \qquad g_{\mathrm{f}}(s) = \frac{\omega_{n,\mathrm{f}}^2}{s^2 + 2\zeta_{\mathrm{f}}\omega_{n,\mathrm{f}}s + \omega_{n,\mathrm{f}}^2}\frac{a_{\mathrm{f}}}{s + a_{\mathrm{f}}}\mathrm{e}^{-\tau_d s}, \tag{4.63}$$

where $\omega_{n,\mathrm{s}} := 1.5\,\mathrm{krad\,s^{-1}}$, $\omega_{n,\mathrm{f}} := 56.6\,\mathrm{krad\,s^{-1}}$, $\zeta_{\mathrm{s}} := 0.99$, $\zeta_{\mathrm{f}} := 1$, $a_{\mathrm{f}} := 37.4\,\mathrm{krad\,s^{-1}}$ and $\tau_d := 100\,\mu\mathrm{s}$. In contrast to the existing actuator model (1.2), the slow and fast transfer functions (4.63) are of order 2 and 3. The TISO and SISO complementary sensitivity functions are therefore extended with roll-off terms as

$$T_{\mathrm{s}\cap\mathrm{f}}(s) = \prod_{k=1}^{3}\frac{a_{\mathrm{s}\cap\mathrm{f},k}}{s + a_{\mathrm{s}\cap\mathrm{f},k}}\mathrm{e}^{-s\tau_d}, \qquad T_{\mathrm{s}\setminus\mathrm{f}}(s) = \prod_{k=1}^{3}\frac{a_{\mathrm{s}\setminus\mathrm{f},k}}{s + a_{\mathrm{s}\setminus\mathrm{f},k}}\mathrm{e}^{-s\tau_d}, \tag{4.64}$$

where $a_{\mathrm{s}\cap\mathrm{f},1} = 1/\tau_d = 10\,\mathrm{krad\,s^{-1}}$ and $a_{\mathrm{s}\setminus\mathrm{f},1} = 0.63\,\mathrm{krad\,s^{-1}}$ and the roll-off parameters are chosen as $a_{\mathrm{s}\cap\mathrm{f},2} = 3\times a_{\mathrm{s}\cap\mathrm{f},1}$, $a_{\mathrm{s}\cap\mathrm{f},3} = 10\times a_{\mathrm{s}\cap\mathrm{f},1}$, $a_{\mathrm{s}\setminus\mathrm{f},2} = 5\times a_{\mathrm{s}\setminus\mathrm{f},1}$, and $a_{\mathrm{s}\setminus\mathrm{f},3} = 20\times a_{\mathrm{s}\cap\mathrm{f},1}$. Note (4.64) could also be realised using a Butterworth filter [98, p. 109].

Based on Section 4.2, the compensators are designed for the Diamond-II orbit response matrices ($\kappa(R_{\mathcal{X}}) = 2.4\times10^3$) and $\kappa(R_{\mathcal{Y}}) = 4.4\times10^3$)) with $\mu = 1$, which results in the sensitivity gains from Fig. 4.13b. With this choice of parameters, the output sensitivity has a bandwidth of 600 Hz and a peak of 3.7 dB at 2 kHz. Note that the delay τ_d and the additional phase lag of the actuator models (4.63) represent worse-case estimates, which result in worst-case values for the bandwidth and sensitivity peak in Fig. 4.13b.



(a) Horizontal direction. **(b)** Vertical direction.

Figure 4.14: Target values (dotted) for Diamond-II, disturbance predictions (dashed) and resulting IBMs (continuous) for long straights (LS), mid straights (MS) and standard straights (SS).

Using PSD estimates $D_{\mathcal{X},(i)}(\omega)$ and $D_{\mathcal{Y},(i)}(\omega)$ of the Diamond-II disturbances [101], an upper bound on the expected output PSD is computed as

$$Y_{\times,(j)}(\omega) = \left(\sum_{i=1}^{n_y} |S_{\times,(j,i)}(\omega)| \sqrt{D_{\times,(i)}(\omega)} \right)^2, \tag{4.65}$$

where $\times = \{\mathrm{s}, \mathrm{f}\}$ and $S_{\times,(j,i)}$ refers to element on row j and column i of $S_\times(\omega)$. The resulting IBMs are shown in Fig. 4.14 for enabled and disabled (dashed) FOFB and BPMs at source points of radiation on long straights (LS), mid straights (MS) and short straights (SS). As evaluated in [2, Ch. 2.11.7], the IBMs from Fig. 4.14 (continuous) are below the target values (dotted), and the Diamond-II performance specifications are therefore satisfied.

4.6 Conclusion

In this chapter, the generalised modal decomposition was proposed for the control of two-array CD systems. The generalised modal decomposition is based on the GSVD that simultaneously factors the ORMs $R_\mathrm{s} \in \mathbb{R}^{n_y \times n_\mathrm{s}}$ and $R_\mathrm{f} \in \mathbb{R}^{n_y \times n_\mathrm{f}}$ of each actuator array. In generalised modal space, the two-array system is decoupled into a set of TISO systems and a set of SISO systems.

Analogous to the single-array controller, the two-array controller was designed in generalised modal space using the IMC structure. For systems with $n_\mathrm{f} < n_\mathrm{s}$,

an input compensator was added to the IMC structure to account for the non-normal transformation into generalised modal space and remove the performance difference between original and generalised modal space. It was shown that the generalised modal decomposition is closely related to the modal decomposition of a hypothetical system with $R = \begin{bmatrix} R_\mathrm{s} & R_\mathrm{f} \end{bmatrix}$, and therefore allows ill-conditioned systems to be treated with regularisation techniques that proved to be efficient for single-array systems. Analogous to the single-array case, the IMC structure was augmented with an output compensator that damps the control action in direction of the small-magnitude singular values of R.

In view of the Diamond-II upgrade that will introduce a two-array system, the proposed algorithm was simulated on preliminary Diamond-II data and also implemented and tested on the existing Diamond storage ring. For the implementation, to mimic the Diamond-II situation, the correctors were divided into a slow and a fast array that were controlling a subset of the BPMs, and the controller dynamics were designed using mid-ranging control. The two-array controller was compared against a single-array controller, and the results showed that the single-array and the two-array controllers perform similarly well. For the two-array controller, the slow array covered the low frequencies, while the fast array attenuated higher frequencies as intended for the Diamond-II upgrade.

Even though the real-world results proved the feasibility and applicability of the proposed control algorithm, several research questions remain. It was shown that due to the output compensator, certain disturbance directions are amplified at frequencies at which the control action is transferred from one actuator array to the other. This amplification does not occur without output compensator, and future research could focus on modifying the output compensator to avoid disturbance amplification in this particular frequency range.

For designing the controller dynamics, a mid-ranging approach was used. As desired for the Diamond-II upgrade, the mid-ranging approach yields integrating behaviour for the slow actuator array, while the fast actuator array does not contribute to the steady-state control action. However, the mid-ranging approach

requires inverting the actuator dynamics $g_{\mathrm{s}}(s)$ and $g_{\mathrm{f}}(s)$, but for Diamond-II the dynamics of the fast actuator array may be such that $g_{\mathrm{f}}(0) = 0$. This means that using a mid-ranging approach would result in undesirable integrating behaviour for the fast actuator array. To avoid this problem, one solution would be to invert only parts of $g_{\mathrm{f}}(s)$ and quantify the resulting performance loss. Alternatively, one could combine the generalised modal decomposition with a $\mathcal{H}_2$ or $\mathcal{H}_\infty$ controller design [130, Ch. 9.3], which would benefit from the sparsity of the system in generalised modal space.

The performance of the algorithms was compared using amplitude spectral density and integrated beam motion figures. While these figures are sufficient to evaluate the performance of a single algorithm, they only allow a partial comparison of different algorithms as the output is subjected to different disturbances when testing the algorithms in practice. As an alternative to these figures, the algorithms could be compared using the output sensitivity, which, in theory, is independent of the actual disturbance affecting the output during experiments. However, the input-output signals that are obtained from the experiments are closed-loop measurements and therefore noise-correlated, which prohibits from applying techniques from system identification to estimate the output sensitivity [154]. Future research could focus on introducing a beam position reference signal with the aim of identifying the complementary sensitivity. The reference signal would need to cover the whole frequency range during which the control action is significant, as well as the high-dimensional spatial output space. In addition, reference directions that are aligned with higher-order modes would need to be treated differently for ill-conditioned systems.

Appendix

4.A Standard Feedback Controller

By inspecting Fig. 4.1, the transfer function from $-y(s)$ to $u(s)$, i.e. the standard feedback controller $C : \mathbb{C}^{n_y} \mapsto \mathbb{C}^{n_s+n_f}$, is obtained for $\Upsilon = \begin{bmatrix} I & I \end{bmatrix}^{\mathrm{T}}$ as

$$C(s) = (I - Q(s)P(s))^{-1}Q(s)\Gamma = Q(s)(I - P(s)Q(s))^{-1}\Gamma,$$

where the push-through rule [130, Ch. 3.2] has been used. From the closed-loop dynamics (4.18), the term $(I - Q(s)P(s))^{-1}$ is

$$(I - P(s)Q(s))^{-1} = X\begin{bmatrix} I\frac{1}{S_{s\cap f}(s)} & 0 \\ 0 & I\frac{1}{S_{s\setminus f}(s)} \end{bmatrix}X^{-1},$$

so that using $Q(s)$ (4.14), the standard feedback controller $C(s)$ is obtained as

$$\begin{aligned} C(s) &= \begin{bmatrix} Q_s(s) \\ Q_f(s) \end{bmatrix}X\begin{bmatrix} I\frac{1}{S_{s\cap f}(s)} & 0 \\ 0 & I\frac{1}{S_{s\setminus f}(s)} \end{bmatrix}X^{-1}, \\ &= \begin{bmatrix} U_s & 0 \\ 0 & U_f \end{bmatrix}\begin{bmatrix} \Sigma_s^{-1}q_s(s) & 0 \\ 0 & Iq_s(s) \\ \Sigma_f^{-1}q_s(s) & 0 \end{bmatrix}\begin{bmatrix} I\frac{1}{S_{s\cap f}(s)} & 0 \\ 0 & I\frac{1}{S_{s\setminus f}(s)} \end{bmatrix}X^{-1}, \\ &= \begin{bmatrix} U_s & 0 \\ 0 & U_f \end{bmatrix}\begin{bmatrix} \frac{1-S_{s\setminus f}(s)}{g_s(s)S_{s\cap f}(s)}\Sigma_s^{-1} & 0 \\ 0 & \frac{1-S_{s\setminus f}(s)}{g_s(s)S_{s\setminus f}(s)}I \\ \frac{S_{s\setminus f}(s)-S_{s\cap f}(s)}{g_f(s)S_{s\cap f}(s)}\Sigma_f^{-1} & 0 \end{bmatrix}X^{-1}\Gamma. \end{aligned}$$

The open-loop transfer function $L(s) = P(s)C(s)$ is therefore

$$\begin{aligned} L(s) &= X\begin{bmatrix} \Sigma_s g_s(s) & 0 & \Sigma_f g_f(s) \\ 0 & Ig_s(s) & 0 \end{bmatrix}\begin{bmatrix} U_s & 0 \\ 0 & U_f \end{bmatrix}^{\mathrm{T}}C(s) \\ &= X\begin{bmatrix} I\frac{1-S_{s\cap f}(s)}{S_{s\cap f}(s)} & 0 \\ 0 & \frac{1-S_{s\setminus f}(s)}{S_{s\setminus f}(s)} \end{bmatrix}X^{-1}\Gamma. \end{aligned}$$

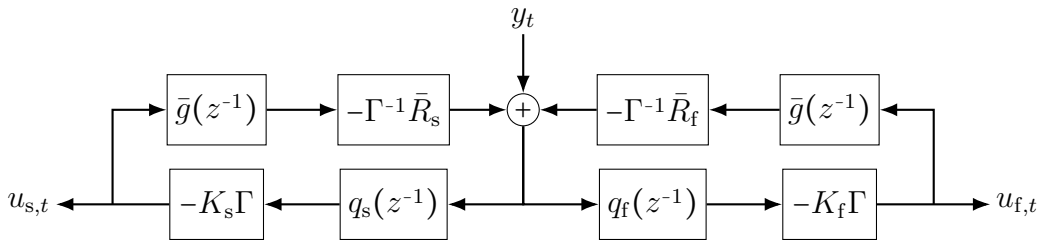


Figure 4.B.1: Rearranged diagram for the implementation of the two-array controller.

4.B Implementation

For the implementation of the two-array controller, the IMC structure from Fig. 4.1 is mapped to discrete time and rearranged into Fig. 4.B.1, where $K_\mathrm{s} \coloneqq \bar{R}_\mathrm{s}^\dagger \in \mathbb{R}^{n_\mathrm{s}\times n_y}$, $K_\mathrm{f} \coloneqq \bar{R}_\mathrm{f}^\dagger \Upsilon_\mathrm{f} \in \mathbb{R}^{n_\mathrm{f}\times n_y}$ and $\bar{g}(z^{-1})$ is the zero-order hold discretisation of (1.2). At time tT_s, $T_s = 100\,\mu\mathrm{s}$, the control inputs $u_{\mathrm{s},t}$ and $u_{\mathrm{f},t}$ are computed from

$$u_{\mathrm{s},t} = -q_\mathrm{s}(z^{-1})K_\mathrm{s}\Gamma^{-1}\Delta y_t, \qquad u_{\mathrm{f},t} = -q_\mathrm{f}(z^{-1})K_\mathrm{f}\Gamma^{-1}\Delta y_t, \tag{4.66}$$

where $q_\mathrm{s}(z^{-1})$ and $q_\mathrm{f}(z^{-1})$ are the zero-order hold discretisations of (4.57) and (4.58), respectively, and $\Delta y_t \coloneqq y_t - \bar{y}_{\mathrm{s},t} - \bar{y}_{\mathrm{f},t}$ with $y_t \in \mathbb{R}^{n_y}$ being the BPM measurements. The signals $\bar{y}_{\mathrm{s},t+1}$ and $\bar{y}_{\mathrm{f},t+1}$ are computed as

$$\bar{y}_{\mathrm{s},t+1} = -\Gamma^{-1}\bar{R}_\mathrm{s}g_\mathrm{s}(z^{-1})u_{\mathrm{s},t}, \qquad \bar{y}_{\mathrm{f},t+1} = -\Gamma^{-1}\bar{R}_\mathrm{f}g_\mathrm{f}(z^{-1})u_{\mathrm{f},t}. \tag{4.67}$$

At time $(t+1)T_s$, the signals $\bar{y}_{\mathrm{s},t+1}$ and $\bar{y}_{\mathrm{f},t+1}$ are used to compute Δy_{t+1}. The computational complexity of the control system is dominated by the matrix-vector multiplications, which require 18 432 floating-point operations for matrices of size $n_\mathrm{s}\times n_y$ and 12 288 for matrices of size $n_\mathrm{s}\times n_y$ ($n_y = n_\mathrm{s} = 96$ and $n_\mathrm{f} = 64$). Neglecting the filter computations, which require roughly 2000 floating-point operations, computing the control inputs using (4.66)-(4.67) requires 61 440 floating-point operations per plane for $n_y = n_\mathrm{s} = 96$ and $n_\mathrm{f} = 64$ (0.6 GFLOPS for $f_s = 10\,\mathrm{kHz}$).

5

The Higher-Order GSVD for Rank-Deficient Matrices

At some point between the conceptual design phase and the technical design phase of Diamond-II, it was suggested that the FOFB could be fitted with *three* different types of corrector magnets, so that the system dynamics would have been given by a three-array cross-directional system:

$$y(s) = R_{\mathrm{s}} g_{\mathrm{s}}(s) u_{\mathrm{s}}(s) + R_{\mathrm{f},1} g_{\mathrm{f},1}(s) u_{\mathrm{f},1}(s) + R_{\mathrm{f},2} g_{\mathrm{f},2}(s) u_{\mathrm{f},2}(s) + d(s). \tag{5.1}$$

Analogous to the single-array and two-array case from Chapter 4, decoupling the dynamics (5.1) into sets of SISO, TISO, and three-input single-output systems would simplify the controller design in many ways and allow concepts from single-array and two-array control to be reused.

The *higher-order GSVD* (HO-GSVD) [119] is an extension of the GSVD to $N \geq 2$ matrices. Given N matrices $A_1, \ldots, A_N$, the HO-GSVD decomposes each A_i as

$$A_i = U_i \Sigma_i V^{\mathrm{T}}, \qquad i = 1, \ldots, N, \tag{5.2}$$

where $U_i \in \mathbb{R}^{m_i \times n}$, $\Sigma_i \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{n \times n}$ with $\det(V) \neq 0$ being shared among all factorisations. The matrix V is obtained from the eigensystem $S_\pi V = V \Sigma$, where

This chapter is based on [85] I. Kempf, P. J. Goulart, and S. R. Duncan, "A higher-order generalized singular value decomposition for rank deficient matrices," *SIAM J. Matrix Anal. Appl.*, 2023, to appear.

$\Sigma \coloneqq \mathrm{diag}(\varsigma_1, \ldots, \varsigma_n)$ and S_π is the *arithmetic mean of all pairwise quotients* $D_{i,\pi} D_{j,\pi}^{-1}$,

$$S_\pi \coloneqq \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(D_{i,\pi} D_{j,\pi}^{-1} + D_{j,\pi} D_{i,\pi}^{-1} \right), \tag{5.3}$$

with $D_{i,\pi}$ defined as

$$D_{i,\pi} \coloneqq A_i^{\mathrm{T}} A_i + \pi A^{\mathrm{T}} A, \qquad \pi \geq 0, \tag{5.4}$$

where $A \coloneqq [A_1^{\mathrm{T}}, \ldots, A_N^{\mathrm{T}}]^{\mathrm{T}}$. The case $\pi = 0$ corresponds to the standard HO-GSVD framework [119].

One shortcoming of the original HO-GSVD framework [119] is that the arithmetic mean (5.3) is only well defined for matrices A_i that have full column rank. If $\mathrm{rank}(A_i) < n$ or $m_i < n$ for some i, then the inverse $(A_i^{\mathrm{T}} A_i)^{-1}$ does not exist and so S_0 in (5.3) is not well defined. In addition, computing (5.3) may be inaccurate when one or more of the A_i have small singular values. Under the assumption that A has full column rank, introducing the term $\pi A^{\mathrm{T}} A$ with $\pi > 0$ will allow the HO-GSVD to accommodate rank-deficient matrices A_i, such as it would be the case for the (transposed) response matrices of the three-array cross-directional system (5.1).

Using the factorisation (5.2), the matrices A_i can be rewritten as

$$A_i = \underbrace{\sum_{k \in \mathcal{I}_N} \sigma_{i,k} u_{i,k} v_k^{\mathrm{T}}}_{\text{common}} + \underbrace{\sum_{k \in \mathcal{I}_1} \sigma_{i,k} u_{i,k} v_k^{\mathrm{T}}}_{\text{isolated}} + \sum_{k \in \mathcal{I}_\perp} \sigma_{i,k} u_{i,k} v_k^{\mathrm{T}}, \tag{5.5}$$

where $\mathcal{I}_N \cup \mathcal{I}_1 \cup \mathcal{I}_\perp = \{1, \ldots, n\}$ and $\mathcal{I}_N$, $\mathcal{I}_1$ and $\mathcal{I}_\perp$ are mutually disjoint. The columns $u_{i,k}$ of the matrices U_i are referred to as *left basis vectors*, and the diagonal matrices Σ_i contain the generalised singular values $\sigma_{i,k}$. The *right basis vectors* v_k are shared across all A_i. It can be shown that the GSVD is a special case of the HO-GSVD with $N = 2$ and that the standard SVD of A_j can be obtained from setting $A_i = I$ for $i \neq j$ and $N \geq 2$ [119]. For the case $\pi = 0$, it was shown in [119] that the subspace associated with the unit eigenvalues of S_π forms the *common HO-GSVD subspace* (see Section 5.2 and Def. 5.8), which is preserved for $\pi > 0$. This subspace is spanned by the right basis vectors v_k, $k \in \mathcal{I}_N$, for which $\sigma_{i,k} = \sigma_{j,k}$, and the associated left generalised singular vectors $u_{i,k}$ are orthogonal to $u_{i,j}$, $j \neq k$.

The HO-GSVD is a technique that is of particular use in multimodal data fusion [93], which aims to identify common features across multiple data sets that describe related phenomena. Many tensor or multi-matrix decompositions are obtained from extending single-matrix factorisations to multiple matrices, such as the parallel factor analysis (PARAFAC [59] or PARAFAC2 [60]), multilinear SVDs [34], multilinear principal component analysis [97], or the higher-order eigenvalue decomposition [7]. The different extensions preserve some but not all of the single-matrix factorisation properties [118], such as exactness, orthogonality, or rank conditions of the factor matrices. Some tensor decompositions require that the matrices A_i share the same dimensions, e.g., a third-order tensor $\mathcal{A} = A_1 \times A_2 \times \dots A_N$ requires that all matrices A_i have dimensions $m \times n$, which imposes constraints on the data acquisition. In contrast, the HO-GSVD is an exact matrix factorisation so that $A_i = U_i \Sigma_i V^{\mathrm{T}}$ for $i = 1, \dots, N$, and it can accommodate $A_i \in \mathbb{R}^{m_i \times n}$ with different row dimensions m_i, although no constraints, such as orthogonality, can be imposed on the factor matrices.

Provided that the matrix A of stacked A_i has full rank, introducing the term $\pi A^{\mathrm{T}} A$ in (5.4) with parameter $\pi > 0$ has the effect of shifting the eigenvalues of each $D_{i,\pi}$, so that the terms $D_{i,\pi}$ are guaranteed to be invertible and the HO-GSVD can be computed for A_i with arbitrary rank. When all A_i have full column rank, it is shown that S_π with $\pi > 0$ and S_0 both capture the common subspaces of $A_1, \dots, A_N$. The notion of an *isolated HO-GSVD subspace* is introduced that accounts for the fact that a rank deficient A_i can have a non-empty (right) nullspace. The isolated HO-GSVD subspace is spanned by the right basis vectors v_k, $k \in \mathcal{I}_1$, for which $\sigma_{i,k} > 0$ and $\sigma_{j,k} = 0$, $j \neq i$. The associated left basis vectors $u_{i,k}$ are orthogonal to $u_{i,l}$, $l \neq k$, $i = 1, \dots, N$.

The GSVD is closely related to the (thin) *cosine-sine decomposition* (CSD) [55, Ch. 2.5.4]. In essence, the CSD states that the SVDs of $Q_1 \in \mathbb{R}^{m_1 \times n}$ and $Q_2 \in \mathbb{R}^{m_2 \times n}$ satisfying $Q_1^{\mathrm{T}} Q_1 + Q_2^{\mathrm{T}} Q_2 = I$ share the same matrix of standard right singular vectors [156]. The GSVD can be obtained from applying a CSD to the matrices Q_1

and Q_2 that are obtained from the thin QR factorisation of the stacked matrices $[A_1^{\mathrm{T}}, A_2^{\mathrm{T}}]^{\mathrm{T}} = QR$, where Q is conformably partitioned such that $A_i = Q_i R$.

Analogous to the GSVD and the CSD, the HO-GSVD is closely related to the *higher-order CSD* (HO-CSD) [157]. The HO-GSVD of N matrices A_i can be obtained from the HO-CSD of $Q_1, \ldots, Q_N$ that are obtained from the thin QR factorisation of the stacked matrices $[A_1^{\mathrm{T}}, \ldots, A_N^{\mathrm{T}}]^{\mathrm{T}}$. As in the case of the HO-GSVD, the computation of the HO-CSD proposed in [157] is limited to the case that all Q_i have full rank. In this chapter, it is also proposed to compute the HO-CSD in a different way, which allows for the factorisation of rank-deficient Q_i satisfying $Q_1^{\mathrm{T}}Q_1 + \cdots + Q_N^{\mathrm{T}}Q_N = I$.

This chapter is organised as follows. Section 5.1 presents the HO-CSD and the HO-GSVD, which are applicable to rank-deficient matrices. In Section 5.2, the notion of common HO-CSD and HO-GSVD subspaces is extended to rank-deficient matrices. The effect of the parameter π is investigated in Section 5.3, followed by relating the rank-deficient HO-GSVD to existing methods in Section 5.4. In Section 5.5, an algorithm for computing the HO-GSVD and the isolated subspace is proposed. The chapter is concluded with two example applications of the HO-GSVD in Section 5.6.

5.1 Main Results

Given N matrices $A_i \in \mathbb{R}^{m_i \times n}$, let A denote the matrix of stacked A_i and $QR = A$ its thin QR factorisation,

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_N \end{bmatrix} = QR = \begin{bmatrix} Q_1 \\ \vdots \\ Q_N \end{bmatrix} R, \qquad Q_i \in \mathbb{R}^{m_i \times n}, \qquad R \in \mathbb{R}^{n \times n}, \tag{5.6}$$

where it holds that

$$Q^{\mathrm{T}}Q = \sum_{i=1}^{N} Q_i^{\mathrm{T}} Q_i = I, \qquad \|Q_i\|_2 \leq 1 \ \forall i = 1, \ldots, N. \tag{5.7}$$

The matrices $A_i = Q_i R$ can individually have arbitrary rank, but throughout the chapter it is assumed that

$$\operatorname{rank}(A) = \operatorname{rank}\begin{pmatrix} A_1 \\ \vdots \\ A_N \end{pmatrix} = n, \tag{5.8}$$

so that $\det(R) \neq 0$ and $M := \sum_{i=1}^{n} m_i \geq n$. If (5.8) does not hold, the matrix A can be padded using an additional matrix A_{N+1} (see Remark 5.16). The quotient terms $D_{i,\pi}$ (5.9) of the arithmetic mean S_π (5.3) can be rewritten as

$$D_{i,\pi} = A_i^{\mathrm{T}} A_i + \pi A^{\mathrm{T}} A = R^{\mathrm{T}}\big(Q_i^{\mathrm{T}} Q_i + \pi I\big)R, \tag{5.9}$$

with parameter $\pi > 0$. Since $A_i^{\mathrm{T}} A_i \succeq 0$ and $\pi A^{\mathrm{T}} A \succ 0$, the terms $D_{i,\pi}$ are guaranteed to be invertible.

Most of the following developments are based on the HO-CSD. Define T_π as

$$T_\pi := \frac{1}{N}\sum_{i=1}^{N}\big(Q_i^{\mathrm{T}} Q_i + \pi I\big)^{-1}, \tag{5.10}$$

where it is assumed that (5.7) holds. The eigensystem of T_π leads to the HO-CSD of the matrices Q_i. It can be shown (Appendix 5.A) that S_π and T_π are related by:

$$R^{-\mathrm{T}} S_\pi R^{\mathrm{T}} = \frac{1}{N-1}\big((1+\pi N)T_\pi - I\big). \tag{5.11}$$

Theorem 5.1. *Let T_π be defined by* (5.10) *and suppose that* (5.7) *holds. There exists an orthogonal $Z \in \mathbb{R}^{n\times n}$ such that*

$$Z^T T_\pi Z = \operatorname{diag}(\tau_1, \ldots, \tau_n), \tag{5.12}$$

where the columns of Z are eigenvectors of T_π and the eigenvalues τ_i of T_π satisfy

$$\tau_i \in [\tau_{min}, \tau_{max}] := \left[\big(N^{-1} + \pi\big)^{-1},\ \frac{N-1}{N}\pi^{-1} + \frac{1}{N}(1+\pi)^{-1}\right].$$

For the proof of Theorem 5.1, the following lemma is used.

Lemma 5.2. *Let $P = P^T \in \mathbb{R}^{n\times n}$ with $0 \preceq P \preceq I$. For all $t \in \mathbb{R}^n$ with $\|t\|_2 = 1$ and $\pi \geq 0$, it holds that $t^T\big(\pi(1+\pi)(\pi I + P)^{-1}\big)t \leq t^T((1+\pi)I - P)t$. Moreover, equality holds iff P has $p \geq 1$ eigenvalues $\lambda_1, \ldots, \lambda_p \in \{0, 1\}$ associated with eigenvectors $v_1, \ldots, v_p$, and $t \in \operatorname{span}(v_1, \ldots, v_p)$.*

Proof. The inequality $t^{\mathrm{T}}\big(\pi(1+\pi)(\pi I + P)^{-1}\big)t \leq t^{\mathrm{T}}((1+\pi)I - P)t$ holds iff

$$(1+\pi)I - P - \pi(1+\pi)(\pi I + P)^{-1} \succeq 0. \tag{5.13}$$

Set $P = V\Lambda V^{\mathrm{T}}$ with $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, $\lambda_i \in [0,1]$, so that (5.13) amounts to

$$f_i(\lambda_i) \coloneqq 1 + \pi - \lambda_i - \frac{\pi(1+\pi)}{\pi + \lambda_i} \geq 0, \qquad i = 1, \ldots, n.$$

Since $f_i''(\lambda_i) = -2\pi(1+\pi)/(\pi+\lambda_i)^3 < 0$ for $\lambda_i \in [0,1]$, the function $f_i(\lambda_i)$ is concave on $\lambda_i \in [0,1]$ and hence $f_i(\lambda_i) \geq \min\{f_i(0), f_i(1)\} = \min\{0,0\} = 0 \ \forall i = 1, \ldots, n$. Equality therefore holds iff $\lambda_i \in \{0,1\}$.

For the second part of the claim, set $t = Va$ with $\|a\|_2 = 1$, and pre- and post-multiply (5.13) with t^{T} and t, respectively, to obtain

$$\sum_{i=1}^{n} f_i(\lambda_i) a_i^2 \geq 0. \tag{5.14}$$

Suppose that $t \in \operatorname{span}(v_1, \ldots, v_p)$, then $\sum_{i=1}^{n} f_i(\lambda_i)a_i^2 = \sum_{i=1}^{p} f_i(\lambda_i)a_i^2 = 0$. For the converse, suppose that $t \notin \operatorname{span}(v_1, \ldots, v_p)$ and that equality holds in (5.14). Then there exists $j \in \{p+1, \ldots, n\}$ with $a_j^2 > 0$ and $f_j(\lambda_j) > 0$, which is a contradiction. □

Proof of Theorem 5.1. The existence of a matrix $Z \in \mathbb{R}^{n\times n}$, $Z^{\mathrm{T}}Z = I$, that diagonalizes T_π is a consequence of the symmetry in (5.10). For the lower bound, substitute $u = (Q_i^{\mathrm{T}}Q_i + \pi I)^{\frac{1}{2}}t$ and $v = (Q_i^{\mathrm{T}}Q_i + \pi I)^{-\frac{1}{2}}t$ with $\|t\|_2 = 1$ in the Cauchy-Schwarz inequality $(u^{\mathrm{T}}v)^2 \leq \|u\|_2^2\|v\|_2^2$ to obtain

$$t^{\mathrm{T}}\left(Q_i^{\mathrm{T}}Q_i + \pi I\right)^{-1} t \geq \left(t^{\mathrm{T}}(Q_i^{\mathrm{T}}Q_i + \pi I)t\right)^{-1}. \tag{5.15}$$

Using (5.15) and the harmonic-mean arithmetic-mean (HM-AM) inequality [57, Thm. 16], a lower bound on $t^{\mathrm{T}}T_\pi t$ can be established as

$$t^{\mathrm{T}}T_\pi t = \frac{1}{N}\sum_{i=1}^{N} t^{\mathrm{T}}\left(Q_i^{\mathrm{T}}Q_i + \pi I\right)^{-1} t \geq \frac{1}{N}\sum_{i=1}^{N} \frac{1}{t^{\mathrm{T}}(Q_i^{\mathrm{T}}Q_i + \pi I)t} \tag{5.16a}$$

$$\geq \frac{N}{\pi N + \sum_{i=1}^{N} t^{\mathrm{T}}(Q_i^{\mathrm{T}}Q_i)t} = \tau_{\min}. \tag{5.16b}$$

For the upper bound, apply Lemma 5.2 with $P = Q_i^{\mathrm{T}}Q_i$ to each summand of T_π:

$$t^{\mathrm{T}}T_\pi t \leq \frac{1}{N}\sum_{i=1}^{N} t^{\mathrm{T}}\left(\frac{1}{\pi}I - \frac{1}{\pi(1+\pi)}Q_i^{\mathrm{T}}Q_i\right)t = \frac{1}{\pi} - \frac{1}{N\pi(1+\pi)} = \tau_{\max}. \tag{5.17}$$

□

Theorem 5.3. *Let S_π be defined by* (5.3) *and suppose that* (5.8) *holds. There exists an invertible $V \in \mathbb{R}^{n\times n}$ such that*

$$V^{-1} S_\pi V = \operatorname{diag}(\varsigma_1, \ldots, \varsigma_n), \tag{5.18}$$

where the columns of V are eigenvectors of S_π and the eigenvalues ς_i satisfy

$$\varsigma_i \in [\varsigma_{min}, \varsigma_{max}] := \left[1,\ 1 + \frac{1}{\pi N(1+\pi)}\right].$$

Proof. Pre- and post-multiplying (5.11) with Z^{T} and Z from Theorem 5.1 yields

$$Z^{\mathrm{T}} R^{-\mathrm{T}} S_\pi R^{\mathrm{T}} Z = \frac{1}{N-1}\big((1+\pi N) Z^{\mathrm{T}} T_\pi Z - I\big).$$

Since $Z^{\mathrm{T}} T_\pi Z = \operatorname{diag}(\tau_1, \ldots, \tau_n)$, the matrix $Z^{\mathrm{T}} R^{-\mathrm{T}} S_\pi R^{\mathrm{T}} Z$ is diagonal. Set $V := R^{\mathrm{T}} Z$, which is invertible because $\det(R) \neq 0$ and $Z^{\mathrm{T}} Z = I$, then the columns of V are eigenvectors of S_π associated with eigenvalues $\varsigma_i = ((1+\pi N)\tau_i - 1)/(N-1)$. The bounds on ς_i are obtained from the bounds on τ_i. □

The significance of Theorems 5.1 and 5.3 is that the diagonalizable matrices T_π and S_π have eigenvalues that are both bounded away from zero and contained in finite intervals, in contrast to the original formulation [119] that requires a full rank condition and corresponds to $\pi = 0$. More precisely, the range of eigenvalues of S_π is contracted from $[1, \infty)$ for the original formulation to $[1, 1 + 1/(\pi N(1+\pi))]$ in the rank-deficient case, which bounds the spectral condition number as $\kappa(S_\pi) := \|S_\pi\|_2 \|S_\pi^{-1}\|_2 \leq 1 + 1/(\pi N(1+\pi))$.

Before examining the eigenvalues of S_π and T_π further, the version of the HO-CSD and HO-GSVD for rank-deficient matrices is stated. The HO-CSD and HO-GSVD have already been described in [157] and [119], respectively, but the modified $D_{i,\pi}$ from (5.9) allows one to omit the requirements that A_i and Q_i be full rank.

Definition 5.4 (HO-CSD)**.** Given $Q_1, \ldots, Q_N$ satisfying (5.7) and $N \geq 2$, the HO-CSD of $Q_i \in \mathbb{R}^{m_i \times n}$ is given by $Q_i = U_i \Sigma_i Z^{\mathrm{T}}$, $i = 1, \ldots, N$, with Z defined as in (5.12). The matrices $\Sigma_i \in \mathbb{R}^{n\times n}$ with $\Sigma_i = \operatorname{diag}(\sigma_{i,1}, \ldots, \sigma_{i,n}) \geq 0$ are obtained from

$$B_i := Q_i Z, \qquad B_i = [b_{i,1}, \ldots, b_{i,n}], \qquad \sigma_{i,k} = \|b_{i,k}\|_2,$$

and $U_i \in \mathbb{R}^{m_i \times n}$ with $U_i = [u_{i,1}, \ldots, u_{i,n}]$ from

$$u_{i,k} = \begin{cases} b_{i,k}/\sigma_{i,k} & \text{if } \sigma_{i,k} > 0 \\ u \in \mathbb{R}^{m_i} \text{ with } \|u\|_2 = 1 & \text{if } \sigma_{i,k} = 0. \end{cases}$$

The left basis vectors $u_{i,k}$ have unit 2-norm and are, under certain circumstances, mutually orthogonal, in which case they coincide with certain left generalised singular vectors of all pair-wise standard GSVD factorisations (see Section 5.2). Because Q_i with $\operatorname{rank}(Q_i) < n$ are allowed, it is possible that $Q_i z_k = 0$ for some eigenvector z_k of T_π, consequently making the corresponding generalised singular value $\sigma_{i,k} = 0$. In these cases, the column $u_{i,k}$ can be chosen freely or the corresponding row of Σ_i can be dropped. Alternatively, they can be chosen to be orthogonal to all other columns, such as stated in the following lemma.

Lemma 5.5. *Suppose that $r_i := \operatorname{rank}(Q_i) < \min(m_i, n)$, and let the generalized singular values be ordered such that $\sigma_{i,k} = 0$ for $k \leq K$, and $\sigma_{i,j} > 0$ for $j > K$. There exist $m_i - r_i \leq K$ mutually orthogonal vectors $u_{i,1}, \ldots, u_{i,(m_i - r_i)}$ such that $u_{i,k}^T u_{i,j} = 0$ $\forall k \leq m_i - r_i$, $j > K$.*

Proof. Note that $\operatorname{span}(u_{i,K+1}, \ldots, u_{i,n}) = \operatorname{range}(Q_i)$. Since $r_i < \min(m_i, n)$, there exist $m_i - r_i$ vectors $u_{i,1}, \ldots, u_{i,(m_i - r_i)}$ satisfying $\operatorname{span}\big(u_{i,1}, \ldots, u_{i,(m_i - r_i)}\big) = \ker(Q_i^{\mathrm{T}})$ and $u_{i,k}^{\mathrm{T}} u_{i,j} = 0$ $\forall k \leq m_i - r_i$, $j > K$, e.g. the last $m_i - r_i$ columns of the matrix of standard left singular vectors of Q_i. □

Definition 5.6 (HO-GSVD)**.** Given $A_1, \ldots, A_N$ satisfying (5.8) and $N \geq 2$, the HO-GSVD of $A_i \in \mathbb{R}^{m_i \times n}$ is given by $A_i = U_i \Sigma_i V^{\mathrm{T}}$, with V defined as in (5.18). The matrices $\Sigma_i \in \mathbb{R}^{n \times n}$ with $\Sigma_i = \operatorname{diag}(\sigma_{i,1}, \ldots, \sigma_{i,n}) \geq 0$ are obtained from

$$B_i := A_i V^{-\mathrm{T}}, \qquad B_i = [b_{i,1}, \ldots, b_{i,n}], \qquad \sigma_{i,k} = \|b_{i,k}\|_2,$$

and $U_i \in \mathbb{R}^{m_i \times n}$ with $U_i = [u_{i,1}, \ldots, u_{i,n}]$ from

$$u_{i,k} = \begin{cases} b_{i,k}/\sigma_{i,k} & \text{if } \sigma_{i,k} > 0 \\ u \in \mathbb{R}^{m_i} \text{ with } \|u\|_2 = 1 & \text{if } \sigma_{i,k} = 0. \end{cases}$$

According to Theorem 5.3, Definitions 5.4 and 5.6 are equivalent in the sense that the HO-GSVD can be obtained from setting $V = R^{\mathrm{T}}Z$:

$$B_i = A_i V^{-\mathrm{T}} = Q_i R R^{-1} Z = Q_i Z, \tag{5.19}$$

where the rightmost term corresponds to B_i as found in Def. 5.4. The matrix of left basis vectors U_i and the generalised singular values therefore depend only on the column space Q. However, when the HO-GSVD and the HO-CSD are computed separately, and T_π and S_π have eigenvalues with geometric multiplicity greater than 1, it is possible that $V \neq R^{\mathrm{T}}Z$.

Remark 5.7. For rank-deficient A_i, the reader may wonder why the standard formulation of S_π and T_π with $\pi = 0$ are not adapted by substituting the pseudoinverse for the inverse in (5.3) and (5.10). The reason is that, in general, $A_i^\dagger = (Q_i R)^\dagger \neq R^\dagger Q_i^\dagger$ and using the pseudoinverse, the relationship (5.11) does not hold. However, relationship (5.11) is fundamental in determining the minimum and maximum eigenvalue of S_π that will play an important role in subsequent sections, which is why pseudoinverses are not considered further.

5.2 Common and Isolated Subspaces

The HO-CSD and HO-GSVD identify directions, corresponding to columns of Z and V, that, in the sense of (5.5), contribute equally to the factorisations of Q_i and A_i, respectively. The directions are the right basis vectors $v_{i,k}$ associated with generalised singular values that are identical for each Q_i and A_i, i.e. $\sigma_{i,k} = \sigma_{j,k}$. These vectors form subspaces [119], [157], which are referred to as the common HO-CSD and HO-GSVD subspaces, and are defined in the following:

Definition 5.8. The common HO-CSD subspace is defined as

$$\mathcal{T}_N\{Q_1, \ldots, Q_N\} := \{z \in \mathbb{R}^n \mid T_\pi z = \tau_{\min} z\},$$

and the common HO-GSVD subspace is defined as

$$\mathcal{S}_N\{A_1, \ldots, A_N\} := \{v \in \mathbb{R}^n \mid S_\pi v = \varsigma_{\min} v\},$$

where $\tau_{\min}$ and $\varsigma_{\min}$ are the lower bounds on the range of eigenvalues defined in Theorems 5.1 and 5.3, and $N \geq 2$.

Note that for a given set of matrices $A_1, \ldots, A_N$, the subspaces $\mathcal{T}_N\{Q_1, \ldots, Q_N\}$ and $\mathcal{S}_N\{A_1, \ldots, A_N\}$ might be empty. By Theorem 5.3, the HO-GSVD and HO-CSD subspaces are related by

$$\mathcal{S}_N\{A_1, \ldots, A_N\} = \left\{R^{\mathrm{T}} z \in \mathbb{R}^n \mid z \in \mathcal{T}_N\{Q_1, \ldots, Q_N\}\right\}, \tag{5.20}$$

so that $v \in \mathcal{S}_N\{A_1, \ldots, A_N\}$ iff $z = R^{-\mathrm{T}} v \in \mathcal{T}_N\{Q_1, \ldots, Q_N\}$. The definition of the common subspace is complemented in the following theorem.

Theorem 5.9. *The following statements are equivalent:*

5.9a $\mathcal{T}_N\{Q_1, \ldots, Q_N\} \neq \varnothing$.

5.9b There exists $\hat{z} \in \mathbb{R}^n$ *that is a standard right singular vector for each* Q_i *and associated with a standard singular value* $\hat{\sigma} = 1/\sqrt{N}$ *for each* Q_i.

5.9c For each Q_i, *there is a left basis vector* $u_{i,k}$ *satisfying* $u_{i,k}^T u_{i,p} = 0 \ \forall p \neq k$ *and the corresponding generalised singular values is* $\sigma_{i,k} = 1/\sqrt{N}$ *for each* Q_i.

Proof. The biconditional relationship 5.9a $\Leftrightarrow$ 5.9b is a consequence of Theorem 5.1. Equality holds in (5.15) iff t is an eigenvector of $Q_i^{\mathrm{T}} Q_i$ [57, Thm. 7] or consequently in (5.16a) iff t is an eigenvector of each $Q_i^{\mathrm{T}} Q_i$ for $i = 1, \ldots, N$. Equality holds in (5.16b) iff $t^{\mathrm{T}}(Q_i^{\mathrm{T}} Q_i + \pi I)t = t^{\mathrm{T}}(Q_j^{\mathrm{T}} Q_j + \pi I)t$ for $i, j = 1, \ldots, N$. It follows that $T_\pi t = \tau_{\min} t$ iff t is a standard right singular vector for each Q_i and from (5.7) that $1 = N\hat{\sigma}^2$, where $\hat{\sigma} = 1/\sqrt{N}$ is the corresponding standard singular value. To show 5.9b $\Rightarrow$ 5.9c, let $\hat{u}_{i,k}$ be the corresponding standard left singular vector, then $Q_i z_k = \hat{\sigma}\hat{u}_{i,k}$ and from the HO-CSD, $Q_i z_k = \sigma_{i,k} u_{i,k}$, so the generalised singular values satisfy $\sigma_{i,k} = \hat{\sigma}$ since $\|\hat{u}_{i,k}\|_2 = \|u_{i,k}\|_2 = 1$. To show that $u_{i,k}^{\mathrm{T}} u_{i,p} = 0 \ \forall p \neq k$, consider the following equations for $\sigma_{i,p} \neq 0$:

$$u_{i,k}^{\mathrm{T}} u_{i,p} = \frac{b_{i,k}^{\mathrm{T}} b_{i,p}}{\sigma_{i,k}\sigma_{i,p}} = \frac{z_k^{\mathrm{T}} Q_i^{\mathrm{T}} Q_i z_p}{\sigma_{i,k}\sigma_{i,p}} = \frac{\sigma_{i,k}}{\sigma_{i,p}} z_k^{\mathrm{T}} z_p = 0,$$

where $b_{i,k}$ denotes column k of the matrix B_i from Def. 5.4.

To show 5.9c $\Rightarrow$ 5.9b, suppose that 5.9c holds and let z_k be the corresponding right generalised singular vector. Then, $Q_i^{\mathrm{T}}Q_i z_k = Z\Sigma_i U_i U_i^{\mathrm{T}}\Sigma_i Z z_k = \sigma_{i,k}^2 z_k$ since $u_{i,k}^{\mathrm{T}}u_{i,p} = 0\ \forall p \neq k$, hence z_k is a shared standard right singular vector associated with a standard singular value $\sigma_{i,k}$. □

Note that statement 5.9c implies that the corresponding left basis vector $u_{i,k}$ is an eigenvector for $Q_i Q_i^{\mathrm{T}}$ for each i and therefore also a *standard* left singular vector for each Q_i.

The common HO-GSVD and HO-CSD subspaces are related by (5.20), and Theorem 5.9 can be adapted for the common HO-GSVD subspace as follows.

Corollary 5.10. *The following statements are equivalent:*

5.10a $\mathcal{S}_N\{A_1, \ldots, A_N\} \neq \varnothing$.

5.10b For each A_i, there is a left basis vector $u_{i,k}$ satisfying $u_{i,k}^T u_{i,p} = 0\ \forall p \neq k$ and the corresponding generalised singular value is $\sigma_{i,k} = 1/\sqrt{N}$ for each A_i.

5.10c There exists $v \in \mathbb{R}^n$ that is an eigenvector for each pairwise quotient $D_{i,\pi}D_{j,\pi}^{-1}$ associated with an eigenvalue $\lambda_{i,j} = 1$.

Proof. The biconditional relationship 5.10a $\Leftrightarrow$ 5.10b immediately follows from (5.20) and Theorem 5.9. To show 5.10b $\Rightarrow$ 5.10c, substitute the HO-GSVD in (5.4) to obtain

$$D_{i,\pi} = V \underbrace{\Sigma_i U_i^{\mathrm{T}} U_i \Sigma_i}_{=:W_i} V^{\mathrm{T}} + \pi A^{\mathrm{T}} A = V\left(W_i + \pi \sum_{p=1}^{N} W_p\right)V^{\mathrm{T}}, \tag{5.21}$$

so that $D_{i,\pi}D_{j,\pi}^{-1} = V\big(W_i + \pi\sum_{p=1}^N W_p\big)\big(W_j + \pi\sum_{p=1}^N W_p\big)^{-1}V^{-1}$. Because of 5.10b, each W_i has the block-diagonal form $W_i = \mathrm{diag}(\underline{W_i}, \sigma_{i,k}^2 + \pi\sum_{p=1}^N \sigma_{p,k}^2, \overline{W_i})$, where the scalar entry is on the kth row of W_i and $\underline{W_i}$ and $\overline{W_i}$ are principal submatrices of

W_i. Again from 5.10b, $\sigma_{i,k} = \sigma_{j,k}$, so that $D_{i,\pi}D_{j,\pi}^{-1}v = v$. To complete the proof, it is shown that 5.10c $\Rightarrow$ 5.10a by right-multiplying S_π from (5.3) with v from 5.10c:

$$\begin{aligned} S_\pi v &= \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} (\lambda_{i,j}v + \lambda_{j,i}v) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} v \\ &= \frac{2}{N(N-1)} \sum_{i=1}^{N} (N-i)v = \frac{2}{N(N-1)} \left(N^2 - \frac{N^2+N}{2} \right) v = \varsigma_{\min} v. \end{aligned}$$

□

The "common features" of A_i in (5.5) can therefore be identified by the right basis vectors associated with eigenvalues of S_π that equal $\varsigma_{\min}$. In general, $R^{-\mathrm{T}}z$ is not an eigenvector for $A_i^{\mathrm{T}}A_i = RQ_i^{\mathrm{T}}Q_iR^{\mathrm{T}}$, so that statement 5.9b cannot be adapted to the HO-GSVD subspace, and the corresponding right basis vectors associated with the common subspace are not orthogonal in general. However, the right basis vectors spanning $\mathcal{S}_N\{A_1, \ldots, A_N\}$ are eigenvectors of all pairwise quotients $D_{i,\pi}D_{j,\pi}^{-1}$, which is exploited in [157] to compute the common HO-GSVD subspace using the standard pairwise GSVD. In addition, one can reformulate Statement 5.10c to show that there exists a vector $\tilde{v} = D_{j,\pi}^{-1}v = D_{i,\pi}^{-1}v$, $v \in \mathcal{S}_N\{A_1, \ldots, A_N\}$, that solves the *higher-order generalised singular value problem* $A_i^{\mathrm{T}}A_i\tilde{v} = \mu A_j^{\mathrm{T}}A_j\tilde{v}$ with $\mu = 1$.

In contrast to the common subspace, the isolated part of (5.5) that is unique to a single A_i is identified by the right basis vectors associated with eigenvalues of S_π (T_π) that equal $\varsigma_{\max}$ ($\tau_{\max}$).

Definition 5.11. The isolated HO-CSD subspace is defined as

$$\mathcal{T}_1\{Q_1, \ldots, Q_N\} := \{z \in \mathbb{R}^n \mid T_\pi z = \tau_{\max} z\},$$

and the isolated HO-GSVD subspace is defined as

$$\mathcal{S}_1\{A_1, \ldots, A_N\} := \{v \in \mathbb{R}^n \mid S_\pi v = \varsigma_{\max} v\},$$

where $\tau_{\max}$ and $\varsigma_{\max}$ are upper bounds on the range of eigenvalues defined in Theorems 5.1 and 5.3, and $N \geq 2$.

Theorem 5.12. *The following statements are equivalent:*

5.12a $\mathcal{T}_1\{Q_1, \ldots, Q_N\} \neq \varnothing$.

5.12b There exists $\hat{z} \in \mathbb{R}^n$ *that is a standard right singular vector for each* Q_i *and associated with a standard singular value* $\hat{\sigma}_{j,k} = 1$ *for one* Q_j *and* $\hat{\sigma}_{i,k} = 0$ *for all other* Q_i, $i \neq j$.

5.12c There is a right basis vector $z \in \mathbb{R}^n$ *associated with a generalized singular value* $\sigma_{j,k} = 1$ *for some* Q_j, *and* $\sigma_{i,k} = 0$ *for all other* Q_i, $i \neq j$.

Proof. The biconditional relationship 5.12a $\Leftrightarrow$ 5.12b is a consequence of the proof of Theorem 5.1. According to Lemma 5.2, equality is attained in (5.17) iff for each summand, $t \in \operatorname{span}\left(v_1^i, \ldots, v_p^i\right)$, where v_k^i are eigenvectors of $Q_i^{\mathrm{T}} Q_i$ associated with eigenvalues that are equal to either 0 or 1. It remains to consider (5.7). The relationship 5.12b $\Leftrightarrow$ 5.12c follows from Definition 5.4 and (5.7). □

Corollary 5.13. *If* $\sigma_{j,k} = 1$, *then the corresponding left basis vector* $u_{j,k}$ *satisfies* $u_{j,k}^T u_{j,p} = 0 \ \forall p \neq k$ *with* $\sigma_{j,p} \neq 0$. *If* $\operatorname{rank}(Q_i) < \min(m_i, n)$, *the left basis vectors* $u_{i,k}$ *associated with* $\sigma_{i,k} = 0$ *can be chosen such that* $u_{i,k}^T u_{i,p} = 0 \ \forall p \neq k$ *with* $\sigma_{i,p} \neq 0$.

Proof. According to Theorem 5.12, the right basis vector z_k associated with $\sigma_{j,k} = 1$ is also a standard right singular vector of Q_j, and the proof of $u_{j,k}^{\mathrm{T}} u_{j,p} = 0 \ \forall p \neq k$ with $\sigma_{j,p} \neq 0$ follows the proof of Theorem 5.12. For $\operatorname{rank}(Q_i) < \min(m_i, n)$, the left basis vectors $u_{i,k}$ associated with $\sigma_{i,k} = 0$ can be chosen according to Lemma 5.5. □

Note that for $m_i \geq n$, the left basis vectors $u_{i,k}$ associated with zero or non-zero generalized singular values can *always* chosen to be orthogonal to the remaining left basis vectors (Lemma 5.5).

By Theorem 5.3, the isolated HO-GSVD and HO-CSD subspaces are related by

$$\mathcal{S}_1\{A_1, \ldots, A_N\} = \left\{R^{\mathrm{T}} z \in \mathbb{R}^n \mid z \in \mathcal{T}_1\{Q_1, \ldots, Q_N\}\right\}, \tag{5.22}$$

and Theorem 5.12 is reformulated for the HO-GSVD as follows.

Corollary 5.14. *The following statements are equivalent:*

5.14a $\mathcal{S}_1\{A_1, \ldots, A_N\} \neq \varnothing$.

5.14b For each A_i, there is a left basis vector $u_{i,k}$ satisfying $u_{i,k}^T u_{i,p} = 0 \ \forall p \neq k$, and the corresponding generalised singular value is $\sigma_{j,k} = 1$ for one A_j and $\sigma_{i,k} = 0$ for all other A_i, $i \neq j$.

5.14c There exist $v \in \mathbb{R}^n$ and $i \in \{1, \ldots, N\}$ such that v is an eigenvector for each pairwise quotient $D_{p,\pi} D_{j,\pi}^{-1}$ associated with eigenvalues $\lambda_{i,j} = \frac{1+\pi}{\pi}$, $\lambda_{j,i} = \frac{\pi}{1+\pi}$ and $\lambda_{p,j} = \lambda_{j,p} = 1$ for $j = \{1, \ldots, N\}$, $p = \{1, \ldots, N\}$ and $j \neq p \neq i$.

Proof. The proof follows the proof of Corollary 5.10. To show 5.14a $\Leftrightarrow$ 5.14b, use (5.22) and apply Theorem 5.12. To show 5.14b $\Rightarrow$ 5.14c, use (5.21) while considering Corollary 5.13. Finally, to show 5.14c $\Rightarrow$ 5.14a, compute $S_\pi v$ and assume without loss of generality that $i = 1$:

$$\begin{aligned} S_\pi v &= \frac{1}{N(N-1)} \sum_{j=2}^{N} (\lambda_{1,j} + \lambda_{j,1}) v + \frac{1}{N(N-1)} \sum_{p=2}^{N} \sum_{j=p+1}^{N} (\lambda_{p,j} + \lambda_{j,p}) v \\ &= \frac{1}{N(N-1)} \sum_{j=2}^{N} \left(\frac{1+\pi}{\pi} + \frac{\pi}{1+\pi} \right) v + \frac{1}{N(N-1)} \sum_{p=2}^{N} \sum_{j=p+1}^{N} 2v \\ &= \frac{1}{N} \left(\frac{1+\pi}{\pi} + \frac{\pi}{1+\pi} + N - 2 \right) v = \varsigma_{\max} v. \end{aligned}$$

□

Note that Corollary 5.13 also applies to the left basis vectors associated with the isolated subspace of the HO-GSVD.

Statements 5.9c and 5.12c of Theorems 5.9 and 5.12 show that, in certain cases, the orthogonality of the left factor matrix, which always holds for the standard SVD and GSVD, is preserved for higher-order datasets (see also Section 5.4). If the generalised singular values $\sigma_{i,k}$, the left basis vectors $u_{i,k}$ and the right basis vectors v_k are grouped according to whether they are associated with the common subspace ($k \in \mathcal{I}_N$), the isolated subspace ($k \in \mathcal{I}_1$) or neither of the subspaces ($k \in \mathcal{I}_\perp$), Def. 5.6 can be refined as

$$A_i = \begin{bmatrix} U_{i,\mathcal{I}_1} & U_{i,\mathcal{I}_\perp} & U_{i,\mathcal{I}_N} \end{bmatrix} \begin{bmatrix} \Sigma_{i,\mathcal{I}_1} & & \\ & \Sigma_{i,\mathcal{I}_\perp} & \\ & & I/\sqrt{N} \end{bmatrix} \begin{bmatrix} V_{\mathcal{I}_1} & V_{\mathcal{I}_\perp} & V_{\mathcal{I}_N} \end{bmatrix}^{\mathrm{T}}, \tag{5.23}$$

where $\Sigma_{i,\mathcal{I}_1}$ contains the generalised singular values associated with $\mathcal{S}_1\{A_1,\ldots,A_N\}$ and $\Sigma_{i,\mathcal{I}_\perp} \succ 0$. In the notation of (5.23) and for $\operatorname{rank}(A_i) < \min(m_i, n)$, the three blocks of left basis vectors are mutually orthogonal, e.g. $(U_{i,\mathcal{I}_N})^{\mathrm{T}} U_{i,\mathcal{I}_1} = 0$, which follows from statements 5.10b and 5.14b of Corollaries 5.10 and 5.14. Note that for the HO-GSVD, the right basis vectors are *not* orthogonal in general.

As can also be concluded from Theorems 5.9 and 5.12, the parameter π does not alter the common and isolated subspaces, which shows that the standard HO-GSVD formulation and the present one are equivalent.

Corollary 5.15. *The common and isolated HO-GSVD and HO-CSD subspaces are independent of the value of π.*

Proof. For the HO-CSD, the claim follows from statements 5.9b and 5.12b of Theorems 5.9 and 5.12, which are independent of the value of π. As a consequence of (5.20) and (5.22), the claim is also true for the HO-GSVD. □

Note that Corollary 5.15 ignores potential numerical inaccuracies, which are treated in Section 5.5. Numerical inaccuracies can also cause rank deficiencies of the stacked matrix A, and the following Remark 5.16 explains how the HO-GSVD can be applied even when A does not satisfy (5.8).

Remark 5.16. Suppose that assumption (5.8) does *not* hold and that $\operatorname{rank}(A) = r < n$. Then, S_π is undefined and (5.11) invalid. Let $\operatorname{span}(v_1,\ldots,v_{n-r}) = \ker(A)$ be an orthogonal basis and set $A_{N+1} \coloneqq [v_1, \ldots, v_{n-r}]^{\mathrm{T}}$. The HO-GSVD can be applied to the augmented dataset $A_1,\ldots,A_{N+1}$, and at least $n-r$ directions of the resulting isolated HO-GSVD subspace are associated with $\ker(A)$.

5.3 The Parameter π

The eigenvectors of T_π that are in the common or isolated HO-CSD subspaces are not affected by the choice of π, but other (normalized) eigenvectors can be modified as π varies. Here, the focus lies on the limits of these eigenvectors as $\pi \to 0$ and

$\pi \to \infty$. Since from (5.10) it holds that $\lim_{\pi\to\infty} S_\pi = I$ and $\lim_{\pi\to\infty} T_\pi = 0$, some caution is required in determining the limits of the associated eigenvectors.

Semisimple eigenvalues are expected to be associated with the common or isolated subspaces and therefore not considered further. To examine the remaining eigenvectors associated with simple eigenvalues, the following result will be used in both cases:

Theorem 5.17 ([95, Thm. 7 & 8, Ch. 9.3])**.** *Let $M(x)$ be a differentiable square matrix-valued function of the real variable x. Suppose that $M(0)$ has a simple eigenvalue m_0. Then for x small enough, $M(x)$ has an eigenvalue $m(x)$ that depends differentiably on x with $m(0) = m_0$ and an eigenvector $h(x)$ of $M(x)$ pertaining to the eigenvalue $m(x)$ can be chosen such that it depends differentiably on x.*

The case $\pi \to \infty$:

Lemma 5.18 (Eigenvectors of T_π for $\pi \to \infty$)**.** *Consider the matrix $\tilde{T}_\infty$,*

$$\tilde{T}_\infty := \frac{1}{N}\sum_{i=1}^{N}(Q_i^T Q_i)^2, \tag{5.24}$$

and suppose that $\tilde{T}_\infty$ has a simple eigenvalue $\tilde{\tau}_\infty$ associated with an eigenvector $\tilde{z}_\infty$. Then, there exists an eigenvector $z(\pi)$ of T_π that depends differentiably on π and converges to $\tilde{z}_\infty$ as $\pi \to \infty$.

Proof. Use the Neumann series $(I - M)^{-1} = \sum_{k=0}^{\infty} M^k$ with $\|M\| < 1$ [74, Ch. 1.4] to expand each of the summands in (5.10) as $(Q_i^{\mathrm{T}} Q_i + \pi I)^{-1} = \frac{1}{\pi}\sum_{k=0}^{\infty}\left(\frac{-1}{\pi} Q_i^{\mathrm{T}} Q_i\right)^k$, and rewrite T_π as

$$T_\pi = \frac{1}{N\pi}\sum_{i=1}^{N}\sum_{k=0}^{\infty}\left(\frac{-1}{\pi} Q_i^{\mathrm{T}} Q_i\right)^k = \frac{1}{\pi} I - \frac{1}{N\pi^2} I + \sum_{i=1}^{N}\frac{1}{N\pi^3}(Q_i^{\mathrm{T}} Q_i)^2 + \mathcal{O}\left(\frac{1}{\pi^4}\right),$$

where $\left\|\frac{1}{\pi} Q_i^{\mathrm{T}} Q_i\right\| < 1$ for $\pi > 1$. Set $\tilde{T}(\pi) = \pi^3 (T_\pi - \frac{N\pi - 1}{N\pi^2} I)$, which depends differentiably on π for $\pi > 0$ and has the same eigenvectors as T_π. Neglecting

higher-order terms $\mathcal{O}(1/\pi^4)$, the limit $\lim_{\pi\to\infty}\tilde{T}(\pi) = \tilde{T}_\infty$ is obtained, where equality holds element-wise. Finally, defining

$$M(x) = \begin{cases} \tilde{T}_\infty & x = 0, \\ \tilde{T}(1/x), & x > 0, \end{cases}$$

the proof follows from Theorem 5.17. □

According to Lemma 5.18, the eigenvectors of T_π associated with simple eigenvalues can be chosen such that they converge to those of $\tilde{T}_\infty$ for large π. Suppose that some Q_i has a "dominant" standard right singular vector $\bar{v}$, in the sense that $\bar{v}^{\mathrm{T}}Q_i^{\mathrm{T}}Q_i\bar{v} \gg \bar{v}^{\mathrm{T}}Q_j^{\mathrm{T}}Q_j\bar{v}$ for $j \neq i$. In this case, $\tilde{T}_\infty$ can be rewritten as $\tilde{T}_\infty = Q_i^{\mathrm{T}}Q_i/N + \Delta$ with $\|\Delta\|_2 \ll \|Q_i^{\mathrm{T}}Q_i/N\|_2$. According to standard perturbation theory [55, Ch. 7.2.5], T_π will have an eigenvector $v = \bar{v} + \delta v$ with $\|\delta v\|_2 \ll 1$. By using the orthogonality property $\sum_{i=1}^N Q_i^{\mathrm{T}}Q_i = I$, the matrix $\tilde{T}_\infty$ defined in (5.24) can be rewritten as

$$\begin{aligned} \tilde{T}_\infty &= \frac{1}{N}\sum_{i=1}^{N} Q_i^{\mathrm{T}}Q_i(I - \sum_{j\neq i} Q_j^{\mathrm{T}}Q_j) \\ &= \frac{1}{N}I - \frac{1}{N}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\left(Q_i^{\mathrm{T}}Q_iQ_j^{\mathrm{T}}Q_j + Q_j^{\mathrm{T}}Q_jQ_i^{\mathrm{T}}Q_i\right), \end{aligned} \tag{5.25}$$

where the summands on the second line are referred to as *symmetrised products* or *Jordan products* of $Q_i^{\mathrm{T}}Q_i$ and $Q_j^{\mathrm{T}}Q_j$ [95, Ch. 10]. The form (5.25) shows that the eigenvectors of T_π will also converge to those of a "dominant" symmetrised product for large π.

The case $\pi \to 0$: For any rank-deficient Q_i and $\pi = 0$, the corresponding term $Q_i^{\mathrm{T}}Q_i + \pi I$ appearing in the definition of T_π in (5.10) is singular. However, by using the standard SVD $Q_i^{\mathrm{T}}Q_i = V_i\,\mathrm{diag}(\sigma_{i,1}^2,\ldots,\sigma_{i,r}^2,0,\ldots,0)V_i^{\mathrm{T}}$ with $r = \mathrm{rank}(Q_i)$, one can show that

$$\lim_{\pi\to 0}\pi\left(Q_i^{\mathrm{T}}Q_i + \pi I\right)^{-1} = V_i\,\mathrm{diag}(\underbrace{0,\ldots,0}_{r\text{ times}},\underbrace{1,\ldots,1}_{n-r\text{ times}})V_i^{\mathrm{T}},$$

where this limit is zero if Q_i is instead full rank. The following lemma provides useful information in the case where some of the Q_i are rank-deficient.

Lemma 5.19 (Eigenvectors of T_π for $\pi \to 0$)**.** *Suppose that some of the Q_i are rank-deficient. Consider*

$$\tilde{T}_0 := \frac{1}{N}\sum_{i=1}^{N} Q_i^\dagger Q_i, \tag{5.26}$$

where $Q_i^\dagger = \lim_{\pi\to 0} Q_i^T(\pi I + Q_i Q_i^T)^{-1}$ is the Moore-Penrose pseudoinverse of Q_i [55, P5.5.2], and suppose that $\tilde{T}_0$ has a simple eigenvalue $\tilde{\tau}_0$ associated with an eigenvector $\tilde{z}_0$. Then, there exists an eigenvector $z(\pi)$ of T_π that depends differentiably on π and converges to $\tilde{z}_0$ as $\pi \to 0$.

Proof. Use the Woodbury matrix identity [55, Ch. 2.1.4] to rewrite πT_π for $\pi > 0$ as

$$\pi T_\pi = \frac{1}{N}\sum_{i=1}^{N}\Big(I - Q_i^{\mathrm{T}}\big(\pi I + Q_i Q_i^{\mathrm{T}}\big)^{-1} Q_i\Big), \tag{5.27}$$

with $\lim_{\pi\to 0} \pi\,(T_\pi - \frac{1}{\pi} I) = -\tilde{T}_0$ (element-wise), where $\tilde{T}_0$ and $\pi\,(T_\pi - \frac{1}{\pi} I)$ share the same eigenspace [55, Ch.2]. Differentiability of the matrix πT_π with respect to π at 0 is easily shown by substitution of the standard SVD of each Q_i into (5.27). The proof then follows from Theorem 5.17. □

Note that if *all* Q_i have full column rank, then $Q_i^\dagger Q_i = I$ and $\tilde{T}_0$ has no simple eigenvalues. The matrix $Q_i^\dagger Q_i$ is the orthogonal projector onto range(Q_i^{T}) and $Q_i^\dagger Q_i = I$ if Q_i has full rank. It follows that if some Q_j are rank deficient, then the eigendecomposition of T_π can be chosen such that it equals the eigendecomposition of the sum of projectors onto range(Q_j^{T}) (the orthogonal complement of ker(Q_j)), but if all Q_i have full rank, then the eigenvectors of $\lim_{\pi\to 0} T_\pi$ are those of T_0, i.e. (5.10) with $\pi = 0$.

The limits for S_π can be obtained from pre- and post-multiplying $\tilde{T}_0$ or $\tilde{T}_\infty$ with R^{T} and $R^{-\mathrm{T}}$, respectively.

Apart from rotating the eigenvectors, the choice of π also affects the function $f_\pi : \mathbb{R}^n \to \mathbb{R}_{++}$,

$$f_\pi(v) = \frac{1}{N(N-1)}\sum_{i=1}^{N-1}\sum_{j=1}^{N}\left(\frac{v^{\mathrm{T}}(A_i^{\mathrm{T}}A_i + \pi A^{\mathrm{T}}A)v}{v^{\mathrm{T}}(A_j^{\mathrm{T}}A_j + \pi A^{\mathrm{T}}A)v} + \frac{v^{\mathrm{T}}(A_j^{\mathrm{T}}A_j + \pi A^{\mathrm{T}}A)v}{v^{\mathrm{T}}(A_i^{\mathrm{T}}A_i + \pi A^{\mathrm{T}}A)v}\right), \tag{5.28}$$

where $\|v\|_2 = 1$ and $f_\pi(v) \geq 1$. The function $f_\pi(v)$ measures the arithmetic mean of amplifications in a particular direction v and has been shown to be related to the (HO-)GSVD [26], [73], [157]. For $N = 2$, $\pi = 0$ and full-rank A_1 and A_2, the gradient is zero for vectors that lie in the common HO-GSVD subspace [157], which can be extended to the isolated HO-GSVD subspace (Appendix 5.C). The parameter π has the effect of *flattening out* f_π and, in particular, removing the singularities of $f_\pi(v)$ associated with the nullspace of A_i for $\pi > 0$ and $\text{rank}(A) = n$, since in that case $v^{\mathrm{T}}(A_i^{\mathrm{T}}A_i + \pi A^{\mathrm{T}}A)v > 0$ for $v \neq 0$.

5.4 Comparison with Standard HO-GSVD, GSVD and SVD

When one out of two matrices is the identity matrix, the GSVD reduces to the standard SVD [155]. The same has been shown for the full-rank HO-GSVD [119]. When $N-1$ matrices A_i are identity matrices, then the full-rank HO-GSVD reverts to the standard SVD of A_j, $j \neq i$. Here, this fact is demonstrated for the rank-deficient HO-GSVD as given in Def. 5.6.

Theorem 5.20. *Let A_1 be an arbitrary matrix and $A_2 = \cdots = A_N = I$ with $N \geq 2$. The HO-GSVD of $A_1, A_2, \ldots, A_N$ with $\pi > 0$ yields the standard SVD of A_1.*

Proof. Substitute the standard SVD $\hat{U}_1\hat{\Sigma}_1\hat{V}_1^{\mathrm{T}} = A_1$ and $A_j = I$, $j = 2, \ldots, N$, in (5.4), so that

$$\hat{V}_1^{\mathrm{T}} D_1 \hat{V}_1 = (1+\pi)\hat{\Sigma}_1^{\mathrm{T}}\hat{\Sigma}_1 + \pi(N-1)I, \quad \hat{V}_1^{\mathrm{T}} D_j \hat{V}_1 = \pi\hat{\Sigma}_1^{\mathrm{T}}\hat{\Sigma}_1 + (1+\pi(N-1))I.$$

The summands $D_{i,\pi}D_{j,\pi}^{-1} + D_{j,\pi}D_{i,\pi}^{-1}$ in the definition of S_π (5.3) are therefore diagonalized by $\hat{V}_1$, and $V = \hat{V}_1$ is an orthogonal eigenbasis for S_π. According to Def. 5.6, the HO-GSVD $A_1 = U_1\Sigma_1 V^{\mathrm{T}}$ is obtained from $B_1 = A_1 V^{-\mathrm{T}} = A_1\bar{V}_1 = \bar{U}_1\bar{\Sigma}_1$, so that $U_1 = \hat{U}_1$ and $\Sigma_1 = \hat{\Sigma}_1$. □

The HO-GSVD from Def. 5.6 can also be related to the GSVD. For the special case that $N = 2$, $A_1 \in \mathbb{R}^{m_1 \times n}$ with $m_1 \geq n$ and $\text{rank}(A_1) = n$ and an arbitrary

$A_2 \in \mathbb{R}^{m_2 \times n}$, it can be shown that the HO-GSVD yields Σ_i with $\Sigma_1^{\mathrm{T}}\Sigma_1 + \Sigma_2^{\mathrm{T}}\Sigma_2 = I$ and orthogonal U_1 and U_2.

Theorem 5.21. *For $N = 2$ and $\pi > 0$, the HO-CSD from Def. 5.4 yields the standard CSD and the HO-GSVD from Def. 5.6 yields the standard GSVD.*

Proof. Since $(Q_i^{\mathrm{T}}Q_i + \pi I)^{-1}$ and $Q_i^{\mathrm{T}}Q_i$ with $i = 1, 2$ and $Q_1^{\mathrm{T}}Q_1 + Q_2^{\mathrm{T}}Q_2 = I$ share the same eigenspace for any $\pi \in \mathbb{R}_{++}$ [55, Ch. 2], the eigenvectors z_k for T_π can be chosen such that they are right singular vectors for Q_1 and Q_2. Let $b_{i,k}$ denote the columns of $B_i = Q_i Z$, then for $j \neq k$, $b_{i,k}^{\mathrm{T}} b_{i,j} = z_k^{\mathrm{T}} Q_i^{\mathrm{T}} Q_i z_j = \hat{\sigma}_{i,j}^2 \bar{u}_{i,k}^{\mathrm{T}} \hat{u}_{i,j} = 0$, where $\hat{\sigma}_\times$ and $\hat{u}_\times$ denote standard singular values and left singular vectors, respectively. Hence, from $U_i\Sigma_i = B_i$, the columns of U_i are either zero or orthonormal. Substituting $Q_i = U_i\Sigma_i V^{\mathrm{T}}$ in $Q_1^{\mathrm{T}}Q_1 + Q_2^{\mathrm{T}}Q_2 = I$ yields $Z\Sigma_1^{\mathrm{T}}\Sigma_1 Z^{\mathrm{T}} + Z\Sigma_2^{\mathrm{T}}\Sigma_2 Z^{\mathrm{T}} = I$, and from $Z^{\mathrm{T}}Z = I$, follows $\Sigma_1^{\mathrm{T}}\Sigma_1 + \Sigma_2^{\mathrm{T}}\Sigma_2 = I$. The claim on the HO-GSVD follows from Theorem 5.21 with $V = R^{\mathrm{T}}Z$. □

Remark 5.22. Lemma 5.21 shows that for $N = 2$ the three matrices, T_π, $Q_1^{\mathrm{T}}Q_1$ and $Q_2^{\mathrm{T}}Q_2$, share the same eigenspace, but not every eigendecomposition of T_π yields eigenvectors that are parallel to those of $Q_1^{\mathrm{T}}Q_1$ and $Q_2^{\mathrm{T}}Q_2$. For example, suppose that $\dim(\ker(Q_i)) = 1$ and that $q_i \in \ker(Q_i)$, $i = 1, 2$, are linearly independent. From pre- and post-multiplying $Q_1^{\mathrm{T}}Q_1 + Q_2^{\mathrm{T}}Q_2 = I$ with q_1^{T} and q_2, it holds that $q_1^{\mathrm{T}}q_2 = 0$. It follows that $\dim(\mathcal{T}_1) = 2$, so that T_π has a semisimple eigenvalue. When the associated eigenvectors are computed using numerical software, these will not necessarily be parallel to q_1 and q_2, and the HO-CSD will not necessarily yield orthonormal matrices U_i.

The HO-GSVD from Def. 5.6 can also be compared with the full-rank HO-GSVD [119]. For $N = 2$ and full-rank matrices A_i, both HO-GSVDs have been shown to be equivalent to the GSVD. Both HO-GSVDs have also been shown to yield the SVD of A_j when $A_i = I$ for $i \neq j$. For $N > 2$, however, the HO-GSVD from Def. 5.6 and [119] will in general *not* yield identical factorisations $A_i = U_i\Sigma_i V^{\mathrm{T}}$, even when $\mathrm{rank}(A_i) = n$. This can be seen by comparing the eigenspaces of T_π from (5.10) for varying π, where $\pi = 0$ corresponds to the standard HO-CSD [157].

For $N = 2$, the eigenvectors of T_π are independent of the value of π because its eigenvectors are fixed by the orthogonality property $Q_1^{\mathrm{T}}Q_1 + Q_2^{\mathrm{T}}Q_2 = I$, while for $N > 2$ this property is lost. From Theorem 5.3, it follows that the same holds for the HO-GSVD. However, it can be shown that in case the matrices A_i and Q_i have full rank, then the common HO-CSD and HO-GSVD subspaces will be the same for any value of π (Corollary 5.15) and $N > 2$. Moreover, it follows from Theorem 5.17 that in the full-rank case, the eigenvectors of T_π converge to those of the standard HO-GSVD as $\pi \to 0$.

5.5 Computing the HO-GSVD

The early literature on the standard GSVD ($N = 2$) identified numerical issues for the case that A from (5.6) and therefore R are ill-conditioned [113], [133], [155]. This problem was resolved by basing the GSVD computation on the CSD, hereby avoiding computing the inverse of R. To compute the full HO-GSVD (5.2), it is proposed to use Algorithm 5.1, which is based on the HO-CSD. An experimental Matlab implementation is provided in [82].

Algorithm 5.1 HO-GSVD Computation

Input: $A_1, \ldots, A_N$, $\pi > 0$
Output: Factorisations $A_i = U_i \Sigma_i V^{\mathrm{T}}$, $i = 1, \ldots, N$

1: Obtain $Q_i R = A_i$ for $i = 1, \ldots, N$ from (5.6) $\mathcal{O}(2Mn^2)$
2: Form T_π using (5.10) $\mathcal{O}(Mn^2 + Nn^3)$
3: Obtain the eigenvectors $z_1, \ldots, z_n$ of T_π $\mathcal{O}(n^3)$
4: Determine $\mathcal{I}_1$ and align z_k, $k \in \mathcal{I}_1$ $\mathcal{O}(2Mn^2 + n^3)$
5: **for** $i = 1, \ldots, N$ and $k = 1, \ldots, n$ **do**
6: **if** $\sigma_{i,k} = \|Q_i z_k\|_2 > 0$ **then**
7: Set $u_{i,k} = Q_i z_k / \sigma_{i,k}$
8: **else**
9: Apply Lemma 5.5
10: **end if**
11: **end for** $\mathcal{O}(Mn^2)$
12: Set $V = R^{\mathrm{T}}\big[z_1, \ldots, z_n\big]$ $\mathcal{O}(n^3)$

Given a dataset $A_1, \ldots, A_N$ and a parameter $\pi > 0$, Algorithm 5.1 starts by computing the thin QR factorisation (5.6), which enables the use of the HO-CSD

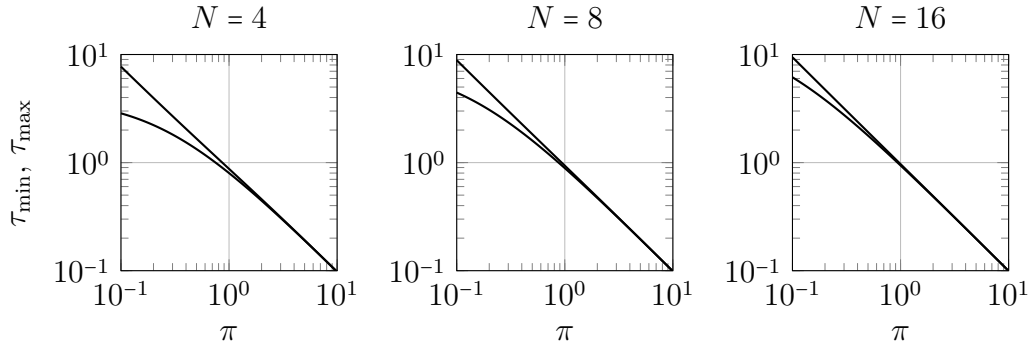


Figure 5.1: Eigenvalues $\tau_{\min}$ and $\tau_{\max}$ of T_π as a function of π for different N.

to avoid computing the inverse of a potentially ill-conditioned R. Next, the terms $(Q_i^{\mathrm{T}}Q_i + \pi I)^{-1}$ are computed to obtain T_π. Forming the products $Q_i^{\mathrm{T}}Q_i$ may lead to a loss of accuracy if Q_i has small singular values, but the condition number $\kappa(Q_i^{\mathrm{T}}Q_i + \pi I)$ can be controlled by choosing π as follows. Let $\kappa_{\max} > 1$ and $\hat{\sigma}_{i,\min}$ and $\hat{\sigma}_{i,\max}$ denote the minimum and maximum standard singular values of Q_i, respectively, then $\kappa(Q_i^{\mathrm{T}}Q_i + \pi I) \leq \kappa_{\max}\ \forall i = 1, \ldots, N$, if π is chosen such that

$$\pi \geq \min_{i \in \{1,\ldots,N\}} \frac{\hat{\sigma}_{i,\max} - \kappa_{\max}\hat{\sigma}_{i,\min}}{\kappa_{\max} - 1}. \tag{5.29}$$

After obtaining the eigenvectors of T_π on line 3, the indices associated with the isolated HO-CSD subspace, $\mathcal{I}_1$, are determined by

$$\mathcal{I}_1 := \left\{ k \in \{1, \ldots, n\} \;\middle|\; \frac{\tau_{\max} - \tau_k}{\tau_{\max} - \tau_{\min}} \leq \epsilon \right\}, \tag{5.30}$$

where τ_k is the corresponding eigenvalue of T_π and the scalar $\epsilon \geq 0$ is introduced to account for finite machine precision. Note the trade-off between (5.29) and (5.30): For increasing π, the difference $\tau_{\max} - \tau_{\min}$ rapidly decreases, such as shown in Fig. 5.1. If the difference $\tau_{\max} - \tau_{\min}$ is too small, numerical inaccuracies can lead to a wrong selection of directions associated with the isolated HO-CSD subspace. The same problem arises when determining the common HO-CSD subspace.

If $\dim(\mathcal{T}_1\{Q_1, \ldots, Q_N\}) \geq 1$, T_π has an eigenvalue that is equal to $\tau_{\max}$ with geometric multiplicity greater than 1. In this case, the corresponding eigenvectors of T_π must be aligned with the shared standard right singular vectors of $Q_1, \ldots, Q_N$. Otherwise, it is not guaranteed that the corresponding generalised singular values

are either 0 or 1. Section 5.5.1 summarises how this can be accomplished by computing a sequence of standard SVDs.

Finally, the matrices Σ_i and U_i are computed in the loop starting on line 5. If $\sigma_{i,k} = 0$, the algorithm substitutes one of the basis vectors spanning $\ker(A^{\mathrm{T}})$. Note that if $\operatorname{rank}(Q_i) = r_i < n$, then $n - r_i$ generalised singular values are zero and $\dim(\ker(A^{\mathrm{T}})) = n - r_i$. On line 12, the shared matrix of right basis vectors V is obtained without the need for computing the inverse of R.

An upper bound on the algorithm complexity is given by summing the shaded numbers on the right-hand side of Algorithm 5.1. The algorithm mainly uses standard routines, such as the QR decomposition or the eigendecomposition, which require roughly $\mathcal{O}(Mn^2)$ floating-point operations. However, forming the matrix T_π requires N matrix inversions of size $n \times n$ or $\mathcal{O}(Nn^3)$ floating-point operations. The accumulation of the N inverses can also lead to a non-trivial loss of accuracy. If the full factorisation (5.2) is not required but only the common or isolated subspace, alternative algorithms exist that compute the common HO-CSD subspace from intersecting the pairwise common HO-GSVD subspaces of Q_i and Q_{i+1} for $i = 1, \dots N-1$ [118], [157]. The pairwise subproblems can be solved by the standard GSVD, which exists as a built-in function in most scientific computing packages, and specialized algorithms exist for large-scale problems [125]. According to Theorem 5.21, the common subspace algorithm from [157] can also be adapted for the isolated HO-CSD subspace.

5.5.1 Computing the Isolated Subspace

It follows from Def. 5.11 and Theorem 5.12 that if

$$\dim(\mathcal{T}_1\{Q_1, \dots, Q_N\}) =: n_{\mathrm{iso}},$$

then T_π has n_{iso} eigenvalues equal to $\tau_{\max}$ and each of the corresponding eigenvectors can be chosen such that it is a standard right singular vector for each Q_i. However, when $n_{\mathrm{iso}} > 1$ the eigendecomposition of T_π will produce an arbitrary set of orthogonal vectors that span $\mathcal{T}_1\{Q_1, \dots, Q_N\}$, but that are not necessarily parallel to the shared

right standard singular vectors. By Def. 5.4, the eigenvectors of T_π spanning $\mathcal{T}_1\{Q_1,\ldots,Q_N\}$ must be aligned with the corresponding standard right singular vectors in order to obtain generalised singular values that are equal to 0 or 1.

Given $Z_{\mathcal{I}_1}$ that has been obtained from (5.30) for some $\epsilon > 0$, one way to align the columns of $Z_{\mathcal{I}_1}$ is to compute the standard SVDs of $Q_i Z_{\mathcal{I}_1}$ for each i, and select those directions associated with standard singular values $\hat{\sigma}_{i,k}$ that satisfy

$$\mathcal{I}_1^i \coloneqq \{k \in \{1,\ldots,n\} \mid \hat{\sigma}_{i,k} \geq 1-\tilde{\epsilon}\}, \qquad i = 1,\ldots,N, \tag{5.31}$$

for some other $\tilde{\epsilon} > 0$. However, in the presence of numerical inaccuracies, it is unclear how to choose $\tilde{\epsilon}$ to obtain exactly n_{iso} directions from (5.31), where n_{iso} is determined from (5.30) for a given $\epsilon \geq 0$. Since from (5.7), (5.10) and Lemma 5.2 it follows that $\|T_\pi z\|_2$ is maximised if z is parallel to the standard right singular vector of Q_i associated with the largest singular value, it appears natural to order the Q_i by magnitude of $\|Q_i Z_{\mathcal{I}_1}\|_2$, and then select the standard right singular vector associated with the largest $\|Q_i Z_{\mathcal{I}_1}\|_2$.

Algorithm 5.2 computes a sequence of ever-thinner standard SVDs to obtain an aligned basis $W_{\mathcal{I}_1}$ from $Z_{\mathcal{I}_1}$, where $W_{\mathcal{I}_1}^{\mathrm{T}} W_{\mathcal{I}_1} = I$ and the columns of $W_{\mathcal{I}_1}$ span the same subspace as those of $Z_{\mathcal{I}_1}$. In the first iteration, the algorithm selects the class i that has the maximum amplification in the subspace spanned by the columns of $Z_{\mathcal{I}_1}$, i.e. by comparing $\|Q_i Z_{\mathcal{I}_1}\|_2$. The corresponding direction $Z_{\mathcal{I}_1}\hat{v}_1$ is assigned to the first column of $W_{\mathcal{I}_1}$. Next, the algorithm selects $n_{\text{iso}} - 1$ remaining directions that are orthogonal to $Z_{\mathcal{I}_1}\hat{v}_1$. Since $Z_{\mathcal{I}_1}$ is orthogonal and at every iteration $\hat{v}_1$ is orthogonal to $\hat{v}_2,\ldots,\hat{v}_{n_{\text{iso}}-k}$, the resulting $W_{\mathcal{I}_1}$ is orthogonal too. Note that the size of X_k decreases at every iteration and that line 6 is not executed at the last iteration. However, given that the first iteration of algorithm Algorithm 5.2 is of the same worst-case complexity as (5.30), which amounts to $\mathcal{O}(2Mn^2 + n^3)$ floating point operations. Both methods – Algorithm 5.2 as well as (5.31) – are implemented in [82], and in Section 5.6, all examples are computed using (5.31).

Algorithm 5.2 Isolated Subspace Computation

Input: $Q_1, \ldots, Q_N$, $Z_{\mathcal{I}_1}$
Output: Aligned basis $W_{\mathcal{I}_1}$
1: Initialize $X_0 \coloneqq Z_{\mathcal{I}_1}$
2: **for** $k = 0, \ldots, n_{\text{iso}} - 1$ **do**
3: Select $p \coloneqq \arg\max_i \|Q_i X_k\|_2$
4: Obtain the standard right singular vectors $\hat{v}_1, \ldots, \hat{v}_{n_{\text{iso}}-k}$ of $Q_p X_k$
5: Assign $X_k \hat{v}_1$ to column $k+1$ of $W_{\mathcal{I}_1}$
6: Update $X_{k+1} \coloneqq X_k \begin{bmatrix} \hat{v}_2 & \ldots & \hat{v}_{n_{\text{iso}}-k} \end{bmatrix}$
7: **end for**

5.6 Applications

The standard (HO-)GSVD has already been applied in various fields such as bioinformatics [119], [159], medicine [92], acoustics [134] or control theory [88]. In practice, the HO-GSVD is used to compare N sets of measurements tabulated in matrices $A_1, \ldots, A_N$, where matrix i represents a different organism, class or experiment, for example. The columns of A_i usually represents a sampled coordinate, such as time or position, whereas the rows of A_i are class-specific variables that vary along the sampled coordinate.

In the form of the HO-GSVD factorisation (5.5), row j of A_i is represented as a linear combination of the right basis vectors $v_1, \ldots, v_n$, which are also weighted by the generalised singular values $\sigma_{i,k}$. In general, the right basis vectors are not orthogonal. However, suppose $A_1, \ldots, A_N$ are such that there exists $v \in \mathbb{R}^n$ such that $A_i^{\mathrm{T}} A_i v \neq 0$ for some i and $A_j^{\mathrm{T}} A_j v = 0$ for $j \neq i$, i.e. v contributes exclusively to the rows of A_i, then, according to Corollary 5.14, v will be an eigenvector of T_π associated with an eigenvalue equal to $\tau_{\max}$. Due to the continuity of the eigenvalues of T_π, it also follows that if $\tau_k \approx \tau_{\max}$, the corresponding right basis vector is almost exclusively used to represent the rows of A_i (see also [118, Ch. 2.3.3]). Similarly, if there exists $\tilde{v} \in \mathbb{R}^n$ such that $A_i^{\mathrm{T}} A_i \tilde{v} = A_j^{\mathrm{T}} A_j \tilde{v}$ for $i, j = 1, \ldots, N$, then according to Statement 5.10c of Corollary 5.10, $D_{j,\pi}^{-1} \tilde{v}$ will be an eigenvector of T_π associated with an eigenvalue equal to $\tau_{\min}$. Among other cases, the condition $A_i^{\mathrm{T}} A_i \tilde{v} = A_j^{\mathrm{T}} A_j \tilde{v}$ holds if $A_1, \ldots, A_N$ share a singular vector $\tilde{v}$ associated with an identical singular value.

Table 5.1: Sample matrices extracted from the first batch of the CIFAR-10 dataset. The rows of each $A_i \in \mathbb{R}^{m_i \times n}$ represent vectorised 32×32 pixels large images.

Matrix	Class	m_i	$\operatorname{rank}(A_i)$	$\lvert\mathcal{I}_1^i\rvert$	$\dim(\mathcal{T}_1\{A_1, \ldots, A_4\})$
A_1	Automobile	974	974	51	
A_2	Cat	1016	1016	92	
A_3	Ship	1025	1025	100	
A_4	Truck	981	981	57	
A		3996	3072		300

To examine the effect of certain right basis vectors onto the rows of class i, A_i can be reconstructed by using a reduced set of right basis vectors, e.g. computing

$$A_{i,\text{iso}} \coloneqq \sum_{k \in \mathcal{I}_1} \sigma_{i,k} u_{i,k} v_k^{\mathrm{T}}, \tag{5.32}$$

yields the reconstruction of A_i using the right basis vectors that are, in the sense of (5.30), exclusively used by class i. To see the effect of the right basis vectors associated with the common subspace, A_i can be reconstructed by summing over $k \in \mathcal{I}_N$.

5.6.1 Image Classification

To illustrate an example application of the HO-GSVD for rank-deficient matrices, consider the CIFAR-10 dataset, which is a collection of images used to evaluate machine learning and computer vision algorithms [91]. The CIFAR-10 dataset provides 6 batches of $10,000$ 32×32 color images in 10 different classes, and here the rank-deficient HO-GSVD is used to analyse a subset of $N = 4$ classes shown in Table 5.1. The following example can be downloaded from [82].

The images are vectorized and grouped in the matrices $A_i \in \mathbb{R}^{m_i \times n}$, where $n = 32 \times 32 \times 3 = 3072$ and $0 \leq A_i \leq 1$ (element-wise). Each A_i is such that $r_i \coloneqq \operatorname{rank}(A_i) < n$, but the stacked $A \in \mathbb{R}^{M \times n}$ satisfies $M > n$ and $\operatorname{rank}(A) = n$. The first row of Fig. 5.2 displays row j_i for each class i as an image[1].

Using the HO-GSVD, the image j of class i can be represented as $\sum_k (e_j^{\mathrm{T}} u_{i,k}) \sigma_{i,k} v_k^{\mathrm{T}}$, where e_j is a standard basis vector and $\lvert e_j^{\mathrm{T}} u_{i,k}\rvert \leq 1$. The columns v_k of the matrix

[1]The rows j_i for class i are $j_1 = 16, j_2 = 19, j_3 = 40$ and $j_4 = 50$, and have been selected to yield an interpretable reconstruction in the isolated subspace.

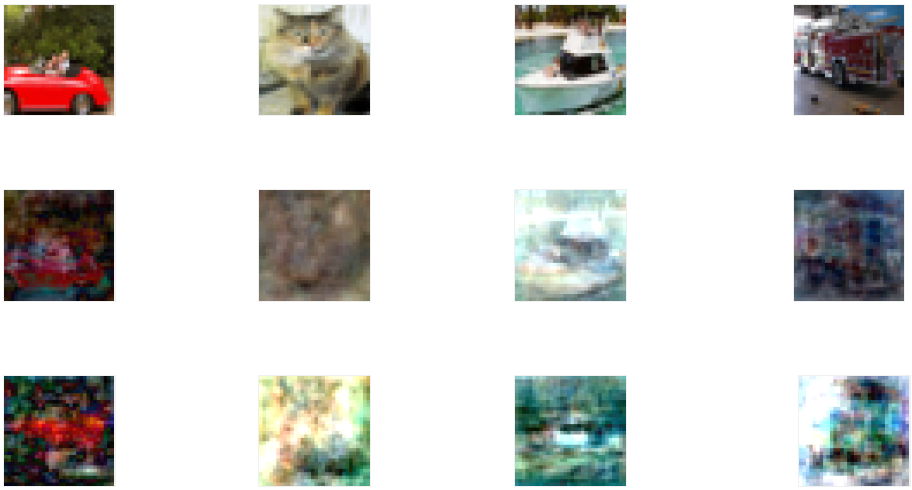


Figure 5.2: First row: Example rows of $A_1, \ldots, A_4$ (left to right) reshaped into 32×32 pixels large images. Second row: Moduli of example rows of $A_{1,\text{iso}}, \ldots, A_{4,\text{iso}}$, where $A_{i,\text{iso}}$ is reconstructed using right basis vectors from the isolated subspace only. Third row: Moduli of isolated right basis vectors that have the largest weight in each image.

$V \in \mathbb{R}^{n \times n}$ with $\det(V) \neq 0$ can be interpreted as "basis images" for the space of 32×32 images, and class i uses r_i columns of V to represent its sample images. Note that the columns of V are not orthogonal, and some right basis vectors can therefore "cancel out" each other. The third row of Fig. 5.2 visualises right basis vectors v_{20}, v_{82}, v_{203} and v_{278}, which are all associated with the isolated subspace (see the subsequent paragraphs).

To limit $\kappa(Q_i^{\mathrm{T}} Q_i + \pi I)$ but retain a large enough difference $\tau_{\max} - \tau_{\min}$ (see Fig. 5.1), the parameter π is chosen as $\pi = 1/N = 0.25$, which results in $\kappa(Q_i^{\mathrm{T}} Q_i + \pi I) \leq 5$, $\tau_{\min} = 2$ and $\tau_{\max} = 3.2$. The $n = 3072$ eigenvalues τ_k of T_π are shown in the first row of Fig. 5.3, where τ_k is displayed relative to $\tau_{\min}$ and $\tau_{\max}$ as $(\tau_k - \tau_{\min})/(\tau_{\max} - \tau_{\min})$ sorted in descending order. It can be seen that most eigenvalues are closer to $\tau_{\max}$ than $\tau_{\min}$, and that $\tau_k \gg \tau_{\min} \quad \forall k$, i.e. the common HO-GSVD subspace is empty. Using a tolerance of $\epsilon = 10^{-6}$, the dimension of the isolated HO-CSD subspace is estimated as $n_{\text{iso}} = 300$. The number of isolated directions per class is computed from (5.31) with $\tilde{\epsilon} = \epsilon$, and $|\mathcal{I}_1^i|$ is shown in Table 5.1 for each class.

The generalised singular values $\Sigma_i = \operatorname{diag}(\sigma_{i,1}, \ldots, \sigma_{i,n})$ are shown on the second to fifth row of Fig. 5.3. For indices $k \in \mathcal{I}_1$ that are associated with the isolated subspaces, the generalised singular values are either 0 or 1. Due to numerical

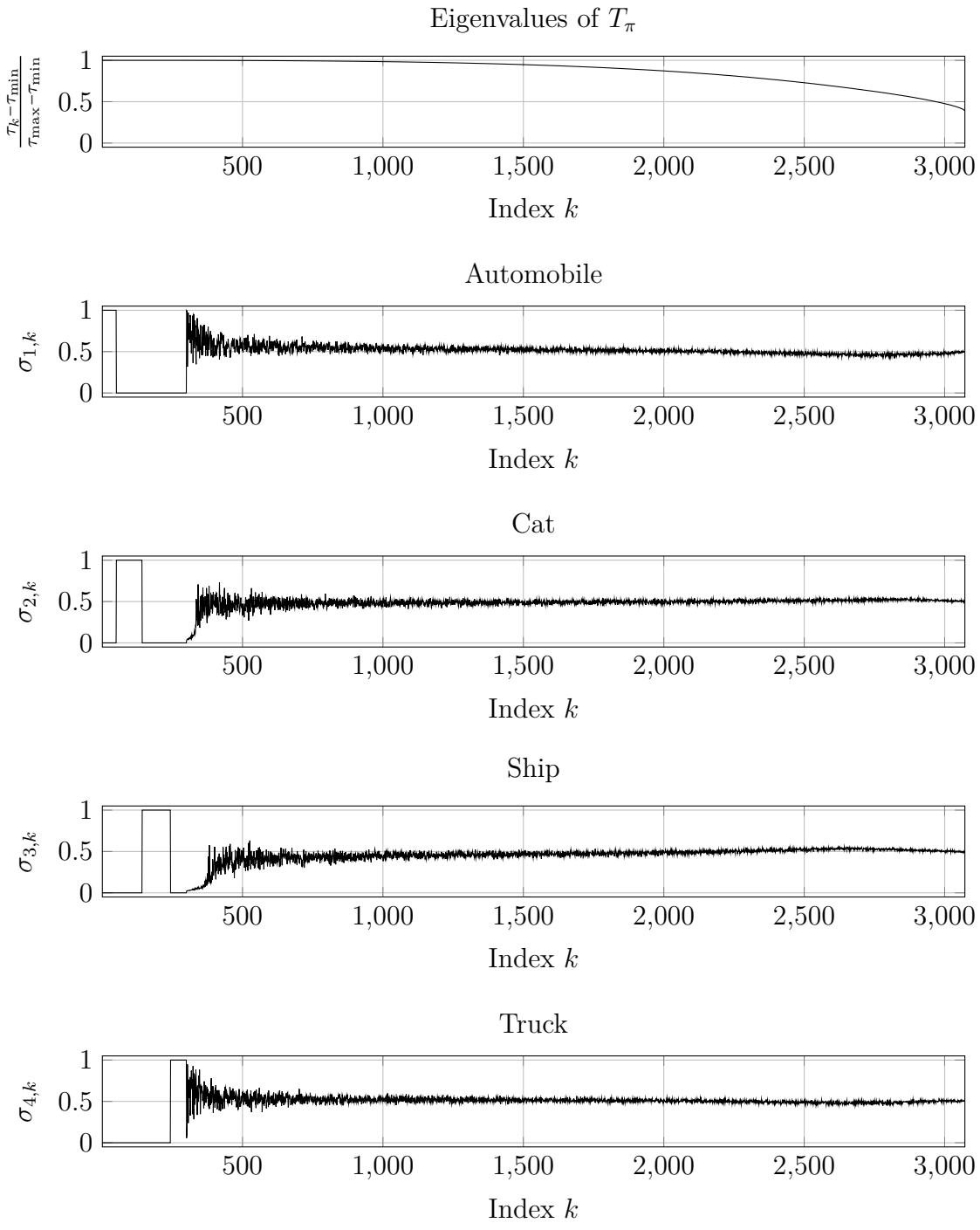


Figure 5.3: Row 1: The $n = 3072$ eigenvalues of T_π relative to the bounds $\tau_{\min}$ and $\tau_{\max}$. Rows 2–5: Corresponding generalised singular values $\Sigma_i = \operatorname{diag}(\sigma_{i,1}, \ldots, \sigma_{i,n})$ for classes $i = 1, \ldots, 4$.

inaccuracies, the separation between $\sigma_{i,k}$, $k \in \mathcal{I}_1$, and $\sigma_{i,j}$, $j \notin \mathcal{I}_1$, is not sharp, i.e. the generalised singular values of the automobile class soar at index $k = 301 \notin \mathcal{I}_1$ before decreasing at larger indices. Note that even though some $\sigma_{i,k}$ equal $1/\sqrt{N} = 0.5$, which is the same magnitude as expected for an index k associated with the common subspace, the common subspace is empty, as can be seen from the first row of Fig. 5.3.

From Fig. 5.3, it becomes clear that each class i uses its own subset of isolated basis images as well as $n - n_{\text{iso}} = 2072$ other columns of V to form its m_i samples. Class i can be reconstructed using (5.32) to obtain $A_{i,\text{iso}}$, which considers indices $k \in \mathcal{I}_1^i$ only. The second row of Fig. 5.2 shows row j_i of $A_{i,\text{iso}}$, where some degree of resemblance between the original and reconstructed image exists. Examples of right basis vectors are given in the third row of Fig. 5.2 that shows v_{20}, v_{82}, v_{203} and v_{278}, each of which is associated with the isolated subspace of classes $i = 1, \ldots, 4$. The right basis vectors have been selected by determining those k that maximise $|e_{j_i}^{\mathrm{T}} u_{i,k}|$ for each image j_i, i.e. those right basis vectors have a large contribution to image j_i. As for the second row of Fig. 5.2, it can be seen that the third row of Fig. 5.2 resembles the original image.

To complement the numerical example, the dataset A is modified in order to artificially introduce a non-empty common subspace. According to Corollary 5.10, the common HO-GSVD subspace, $\mathcal{S}_4\{A_1, \ldots, A_4\}$, is non-empty iff the condition $A_i^{\mathrm{T}} A_i \tilde{v} = A_j^{\mathrm{T}} A_j \tilde{v}$ holds $\forall i, j = 1, \ldots, 4$ and for some $\tilde{v}$, which can be written out as

$$\left(\begin{bmatrix} a_{i,1}^{\mathrm{T}} a_{i,1} & \ldots & a_{i,1}^{\mathrm{T}} a_{i,n} \\ \vdots & \ddots & \vdots \\ a_{i,n}^{\mathrm{T}} a_{i,1} & \ldots & a_{i,n}^{\mathrm{T}} a_{i,n} \end{bmatrix} - \begin{bmatrix} a_{j,1}^{\mathrm{T}} a_{j,1} & \ldots & a_{j,1}^{\mathrm{T}} a_{j,n} \\ \vdots & \ddots & \vdots \\ a_{j,n}^{\mathrm{T}} a_{j,1} & \ldots & a_{j,n}^{\mathrm{T}} a_{j,n} \end{bmatrix} \right) \tilde{v} = 0, \tag{5.33}$$

where $a_{i,k} \in \mathbb{R}^{m_i}$ denotes column k of matrix A_i. If $\tilde{v}$ is chosen as $\begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}^{\mathrm{T}}$, condition (5.33) is tantamount to requiring that $a_{i,k}^{\mathrm{T}} a_{i,1} = a_{j,k}^{\mathrm{T}} a_{j,1}$ $\forall i, j = 1, \ldots, 4$ and for $k = 1, \ldots, n$, i.e. the projection of column k onto the first column of class i must equal the projection of column k onto the first column of class j. Note that condition (5.33) is *not* equivalent to inserting an identical image $x \in \mathbb{R}^n$ in each A_i, but a simple way to satisfy (5.33) is to set $a_{i,1} = \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}^{\mathrm{T}}$ and zero out the

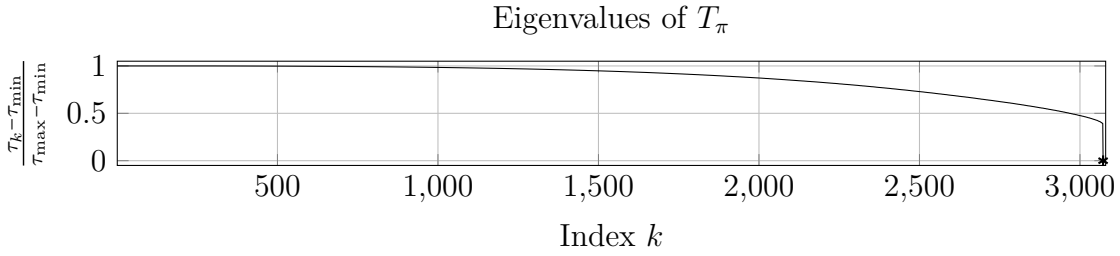


Figure 5.4: The $n = 3072$ eigenvalues of T_π relative to the bounds $\tau_{\min}$ and $\tau_{\max}$ for the modified dataset, which has a one-dimensional common subspace associated with index $k = 3072$ (marked by an asterisk).

first element of $a_{i,k}$, $k = 1, \ldots, n$, for all classes $i = 1, \ldots, 4$. This way the first image of each A_i is replaced with a black square that has one red pixel in the left corner.

The eigenvalues of T_π for the modified dataset are shown in Fig. 5.4. It can be seen that $\tau_k = \tau_{\min}$ for $k = n$, i.e. the modification successfully introduces a non-empty common subspace. The corresponding generalised singular values (not shown) equal $1/\sqrt{N} = 0.5$ for each class. By construction, the first row (image) of each A_i is orthogonal to all other rows of A_i, and therefore aligned with a shared standard right singular vector. The right basis vector associated with the common subspace, v_n, is therefore orthogonal to all other basis vectors, which is not the case in general. However, for this example it follows that v_n contributes equally to each of the matrices A_i, and for each class i, v_n is used to represent the first image only.

5.6.2 Multi-Array Cross-Directional Control

In order to outline its application to multi-array CD systems, the HO-GSVD from Theorem 5.6 is applied to $R_{\mathrm{s}}^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{s}} \times n_y}$, $R_{\mathrm{f},1}^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{f},1} \times n_y}$ and $R_{\mathrm{f},2}^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{f},2} \times n_y}$ from the three-array dynamics (5.1) with $n_y = n_{\mathrm{s}} > n_{\mathrm{f},1} \geq n_{\mathrm{f},2}$. Left-multiplying (5.1) with X^{-1} and introducing the new variables

$$\tilde{y}(s) \coloneqq X^{-1} y(s), \qquad \tilde{u}_{(\cdot)}(s) \coloneqq U_{(\cdot)}^{\mathrm{T}} u_{(\cdot)}(s), \qquad \tilde{d}(s) \coloneqq X^{-1} d(s), \tag{5.34}$$

where $U_{(\cdot)} \in \mathbb{R}^{n_{(\cdot)} \times n_y}$ and $(\cdot) = \{\mathrm{s}, \text{“f},1\text{”}, \text{“f},2\text{”}\}$, decouples the three-array dynamics (5.1) as

$$\tilde{y}(s) = \Sigma_{\mathrm{s}} g_{\mathrm{s}}(s) \tilde{u}_{\mathrm{s}}(s) + \Sigma_{\mathrm{f},1} g_{\mathrm{f},1}(s) \tilde{u}_{\mathrm{f},1}(s) + \Sigma_{\mathrm{f},2} g_{\mathrm{f},2}(s) \tilde{u}_{\mathrm{f},2}(s) + \tilde{d}(s). \tag{5.35}$$

Based on (5.35), one could proceed with designing controllers for the decoupled three-input single-output systems and including them in an IMC structure with the (static) compensators from Chapter 4. In contrast to the two-array case, the matrices $U_{(.)}$ are *not* orthonormal and after computing $\tilde{u}_{(.)}(s)$, the transformation (5.34) would need to be inverted using $u_{(.)}(s) = U_{(.)}^{\dagger}\tilde{u}_{(.)}(s)$. However, this would also require an additional static compensator to be introduced between the controller and the plant (e.g. after $Q(s)$ in Fig. 4.1).

5.7 Conclusion

In this chapter, the standard HO-GSVD [119] has been extended to accommodate column rank-deficient matrices. By adding the term $\pi A^{\mathrm{T}}A$ to each of the quotient terms $D_{i,\pi} = A_i^{\mathrm{T}}A_i + \pi A^{\mathrm{T}}A$, their eigenvalues were shifted and bounded away from zero. This allowed the full-rank requirement on each A_i to be omitted and to extend the HO-GSVD with the notion of isolated subspaces.

The choice of adding a multiple of $A^{\mathrm{T}}A$ was motivated by the relationship between S_π and T_π, which yielded the same relationship than in [119] for $\pi = 0$. The eigenvalues of T_π were bounded and it was shown that the extremal eigenvalues are attained iff the corresponding eigenvectors are standard right singular vectors for each Q_i associated with a particular singular value. This led to the definition of the common and isolated HO-CSD (HO-GSVD) subspaces. In Appendix 5.B, it was also shown that if the Q_i share a right singular vector v associated with a zero singular value for P matrices Q_i and with an identical singular value for the other $N - P$ matrices Q_j, then T_π will have a particular eigenvalue $\tau(P)$ associated with the eigenvector v. Future research could investigate whether a biconditional ("iff") connection holds.

The parameter π was assumed to be positive, but otherwise left unspecified. The common and isolated HO-CSD and HO-GSVD subspaces are identified irrespective of the value of π, but other right basis vectors can be rotated for increasing values of π, and the behavior of these vectors has been investigated for $\pi \to 0$ and $\pi \to \infty$. For $\pi \to 0$, the eigenvectors of T_π are solely determined by the rank-deficient

Q_i, whereas for $\pi \to \infty$, the eigenvectors of T_π converge to those of the mean of symmetrised products of $Q_i^{\mathrm{T}} Q_i$ and $Q_j^{\mathrm{T}} Q_j$.

In addition, the choice of π also affects the condition number of $Q_i^{\mathrm{T}} Q_i + \pi I$, which must be inverted to obtain T_π, as well as the range of admissible eigenvalues of T_π, $\tau_{\max} - \tau_{\min}$. A large π improves the conditioning of $Q_i^{\mathrm{T}} Q_i + \pi I$, but also tightens the range of eigenvalues, which can lead to a wrong estimate of the common or isolated subspaces in the presence of numerical errors. The optimal choice of π remains unclear and future research could investigate the role of the weight π.

Most of the developments were based on the HO-CSD. Using the QR factorisation of $A = [A_1^{\mathrm{T}}, \ldots, A_N^{\mathrm{T}}]^{\mathrm{T}}$, each A_i was represented as $A_i = Q_i R$ and the factorisation was developed for $Q_1, \ldots, Q_N$, which required A to have full column rank. For rank-deficient A, it was shown how A can be padded using an additional matrix A_{N+1} to guarantee that $\det(R) \neq 0$. The properties of $A_1, \ldots, A_N$ were inferred from the HO-CSD, which made it possible to avoid computing the inverse of a potentially ill-conditioned R, but a full factorisation still requires inversion of the terms $Q_i^{\mathrm{T}} Q_i + \pi I$, which can lead to significant numerical errors for large-scale matrices. Future research could focus on finding a possibly iterative algorithm that finds the eigenvectors of T_π without the need for inverting the terms $Q_i^{\mathrm{T}} Q_i + \pi I$.

For the full-rank case, it has been shown that the common subspace can be found using a variational approach [157] and that the vectors v spanning the common subspace are stationary vectors for the function $f_\pi(v)$ (5.28) with $\pi = 0$. It was shown that the same holds for $\pi > 0$. It remains unclear how the right basis vectors, which are not in the common or isolated subspaces, are related to $f_\pi(v)$ and whether an eventual connection would lead to a particular choice of the parameter π.

Appendix

5.A Relation between S_π and T_π

Let $D_{i,\pi} = A_i^{\mathrm{T}} A_i + \pi A^{\mathrm{T}} A$ and define $K_i \coloneqq Q_i^{\mathrm{T}} Q_i + \pi I$. Using (5.9), the matrix S_π is written as

$$\begin{aligned} S_\pi &= \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(D_{i,\pi} D_{j,\pi}^{-1} + D_{j,\pi} D_{i,\pi}^{-1} \right) \\ &= \frac{1}{N(N-1)} R^{\mathrm{T}} \left(\sum_{i=1}^{N} \sum_{j=i+1}^{N} K_i K_j^{-1} + K_j K_i^{-1} \right) R^{-\mathrm{T}}, \end{aligned}$$

so that by considering $\sum_{i=1}^{N} K_i = (1+\pi N) I$

$$\begin{aligned} R^{-\mathrm{T}} S_\pi R^{\mathrm{T}} &= \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} K_i K_j^{-1} + K_j K_i^{-1} = \frac{1}{N(N-1)} \sum_{i=1}^{N} K_i \sum_{j=1}^{N} K_j^{-1} - \frac{1}{N-1} I \\ &= \frac{1}{N-1} ((1+\pi N) T_\pi - I). \end{aligned}$$

5.B Intermediate Eigenvalues of T_π

If there exists a vector t with $\|t\|_2 = 1$ in the nullspace of P matrices Q_j, but in the range of all other Q_i with index $i \in \mathcal{R}$, then the inequalities (5.16a)-(5.16b) can be reformulated as

$$\begin{aligned} t^{\mathrm{T}} T_\pi t &= \frac{1}{N} \sum_{i=1}^{N} t^{\mathrm{T}} \left(Q_i^{\mathrm{T}} Q_i + \pi I \right)^{-1} t = \frac{P}{\pi N} + \frac{1}{N} \sum_{i \in \mathcal{R}} t^{\mathrm{T}} \left(Q_i^{\mathrm{T}} Q_i + \pi I \right)^{-1} t \\ &\geq \frac{P}{\pi N} + \frac{1}{N} \sum_{i \in \mathcal{R}} \frac{1}{t^{\mathrm{T}} (Q_i^{\mathrm{T}} Q_i + \pi I) t} \qquad (5.36a) \\ &\geq \frac{P}{\pi N} + \frac{N-P}{N} \frac{N-P}{\pi(N-P) + \underbrace{\sum_{i \in \mathcal{R}} t^{\mathrm{T}} (Q_i^{\mathrm{T}} Q_i) t}_{=1}} = \frac{P(1-\pi N) + \pi N^2}{\pi N (1 + \pi (N-P))}. \qquad (5.36b) \end{aligned}$$

The term on the right-hand side of (5.36b) corresponds to the minimum and maximum eigenvalues of T_π for $P = 0$ and $P = N-1$, respectively. If there exists a

shared vector t in the nullspace of P matrices Q_j, but in the range of all other Q_i, then an eigenvalue of T_π will be equal to the corresponding value on the right-hand side of (5.36b). Note that (5.36) does *not* prove the converse.

5.C The Arithmetic Mean of Amplification Quotients

The HO-GSVD is related to the function $f_\pi(v)$ (5.28), which can be simplified using the stacked QR decomposition (5.6) as

$$g_\pi(z) = \frac{1}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=1}^{N} \left(\frac{z^{\mathrm{T}}(Q_i^{\mathrm{T}} Q_i + \pi I)z}{z^{\mathrm{T}}(Q_j^{\mathrm{T}} Q_j + \pi I)z} + \frac{z^{\mathrm{T}}(Q_j^{\mathrm{T}} Q_j + \pi I)z}{z^{\mathrm{T}}(Q_i^{\mathrm{T}} Q_i + \pi I)z} \right) \geq 1,$$

where $z \coloneqq Rv$. The gradient $\nabla g_\pi(z)$ of $g_\pi(z)$ is given by

$$\nabla g_\pi(z) = \frac{1}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=1}^{N} \left(\frac{1}{z^{\mathrm{T}} W_{j,\pi} z} \left(W_{i,\pi} z - \frac{z^{\mathrm{T}} W_{i,\pi} z}{z^{\mathrm{T}} W_{j,\pi} z} W_{j,\pi} z \right) + \frac{1}{z^{\mathrm{T}} W_{i,\pi} z} \left(W_{j,\pi} z - \frac{z^{\mathrm{T}} W_{j,\pi} z}{z^{\mathrm{T}} W_{i,\pi} z} W_{i,\pi} z \right) \right),$$

where $W_{i,\pi} \coloneqq Q_i^{\mathrm{T}} Q_i + \pi I$. To show that $\nabla g_\pi(z) = 0$ for $z \in \mathcal{T}_N\{Q_1, \ldots, Q_N\}$ or $z \in \mathcal{T}_1\{Q_1, \ldots, Q_N\}$, note that z must be a right singular vector for each Q_i. It follows that $W_{i,\pi} z = (\sigma_{i,1} + \pi) z$ and

$$W_{i,\pi} z - \frac{z^{\mathrm{T}} W_{i,\pi} z}{z^{\mathrm{T}} W_{j,\pi} z} W_{j,\pi} z = (\sigma_{i,1} + \pi) z - \frac{\sigma_{i,1} + \pi}{\sigma_{j,1} + \pi} (\sigma_{j,1} + \pi) z = 0, \tag{5.37}$$

so that $\nabla g_\pi(z) = 0$ if $z \in \mathcal{T}_N\{Q_1, \ldots, Q_N\}$ or $z \in \mathcal{T}_1\{Q_1, \ldots, Q_N\}$ for any value of π. The proof is analogous for the HO-GSVD subspaces.

6

Cross-Directional Control using Model Predictive Control

In synchrotrons, the corrector magnet inputs are usually constrained by the magnet power supplies that limit the input amplitude and input rate [50]. When these limits are reached, the actuators saturate, which can lead to a severe performance degradation of the control system or even cause instabilities [130, Ch. 6.9]. The potential for actuator saturation is exacerbated by the ill-conditioned plant [129], and can be reduced by decreasing controller gains, though this usually comes at the expense of reducing the closed-loop bandwidth.

As an alternative to reducing controller gains, one solution is to introduce an anti-windup compensator [130, Ch. 12.4] that takes action when the actuators are saturating. For the electron beam dynamics (1.3), an anti-windup compensator has been proposed to account for the performance deterioration caused by rate constraints [50]. Because the proposed compensator does not consider amplitude

This chapter is based on [84] I. Kempf, P. J. Goulart, and S. R. Duncan, "Fast gradient method for model predictive control with input rate and amplitude constraints," in *Proc. IFAC World Congr.*, Berlin, Germany, Jul. 2020, pp. 6542–6547; [89] I. Kempf, P. J. Goulart, S. R. Duncan, *et al.*, "Model predictive control for electron beam stabilization in a synchrotron," in *Proc. Eur. Contr. Conf. (ECC)*, London, UK, Jul. 2022, pp. 814–819; [83] I. Kempf, P. J. Goulart, and S. R. Duncan, "Alternating direction method of multipliers for block circulant model predictive control," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Nice, France, Dec. 2019, pp. 4311–4316.

constraints and is based on the modal decomposition for single-array systems, the anti-windup compensator still has to be adapted for its application to Diamond-II.

Model predictive control (MPC) is an optimisation-based algorithm that selects the control inputs based on the future evolution of the system. The advantages of MPC for constraint handling and feedforward disturbance modelling are widely recognised [46]. Moreover, MPC does not rely on modal decomposition and can therefore consider CD system with an arbitrary number of actuator arrays. However, the applicability of MPC is limited by the requirement to solve optimisation problems in real-time to compute the control law [47, Ch. 8.2]. This constraint has inhibited the application of MPC to large-scale and high speed applications. While some approaches for accelerating the computing speed have focused on implementing optimisation routines on specialised high-performance hardware [72], other approaches have exploited the particular symmetric structure encountered in some classes of large-scale problems [32].

In this chapter, the computational efficiency of the *fast gradient method* [110, Ch. 6.1.3] is leveraged to obtain an MPC implementation that does not rely on structural symmetries nor on highly specialised hardware. By considering input constraints only, the MPC algorithm is simplified and then parallelised on a general-purpose DSP (Chapter 7), resulting in an implementation that computes the control law of an MPC scheme with a horizon consisting of a single time-step in less than 69 µs (14.4 kHz). Based on the single-array CD controller from Section 1.6, a tuning procedure is developed to account for the ill-conditioned ORM, which impacts both the solver convergence and the behaviour of MPC under actuator saturation. As a first-of-its-kind application to electron beam stabilisation, the MPC algorithm is tested on the existing Diamond storage ring and compared with a single-array controller, showing that MPC meets the theoretical expectations and demonstrating its practical feasibility.

This chapter is organised as follows. The MPC algorithm is formulated in Section 6.1 followed by introducing optimisation routines – the fast gradient method and ADMM – in Section 6.2. Section 6.3 addresses the input projection method

and evaluates the optimisation routines. In Section 6.4, the observer is designed by reverse engineering the existing Diamond controller from Section 1.6. The MPC problem is then tuned with respect to actuator saturation and constraints in Section 6.5. Section 6.6 summarises the implementation and the chapter is concluded with real-world results from Diamond Light Source in Section 6.7.

6.1 Model Predictive Control

6.1.1 Discrete-Time State-Space Representation

System (1.3) is the Laplace transform of the input-output representation of the continuous-time dynamics, but the MPC formulation of this chapter uses the discrete-time *state-space representation* of the continuous-time system [111, Ch. 3.3]. At Diamond, the controller is implemented using a "sample and hold" mode, where the input $u(t)$ is held constant for $t \in [t_0, t_0 + T_s)$ with $T_s = 100\,\mu s$ being the sample time, so that the discrete-time representation with $u(kT_s) =: u_k$, $k \in \mathbb{Z}$, accurately replicates the continuous-time dynamics [38]. Mapping (1.3) to the $\mathcal{Z}$ domain [111, Ch. 13.3] yields

$$y(z) = Rg(z)u(z) + d(z), \tag{6.1}$$

where z is the $\mathcal{Z}$ variable (or the forward shift operator: $zy_k = y_{k+1}$) and $g(z)$, the discrete-time representation of $g(s)$, is defined as

$$g(z) := \frac{1 - \pi_g}{z - \pi_g} z^{-n_\tau}, \tag{6.2}$$

where $\pi_g := \exp(-aT_s)$ is the discrete-time pole and n_τ the delay in terms of time steps. The representation (6.2) assumes that τ_d is an integer multiple of the sample time T_s, so that $n_\tau := \tau_d/T_s = 9$. Otherwise, a discrete-time zero must be added to $g(z)$ [66], which does not change the developments of this chapter and is therefore omitted.

To obtain a state-space representation of (6.1), the $\mathcal{Z}$ transform is inverted,

$$\begin{aligned} y_k &= \mathcal{Z}^{-1}\{Rg(z)u(z) + d(z)\}, \\ &= R\mathcal{Z}^{-1}\{g(z)u(z)\} + d_k, \\ &= Rx_{k-n_\tau} + d_k, \end{aligned} \tag{6.3}$$

where $x_k \coloneqq \mathcal{Z}^{-1}\left\{\frac{1-\pi_g}{z-\pi_g}u(z)\right\} \in \mathbb{R}^{n_u}$, the *state* of the system, can be interpreted as the current through the correctors:

$$x_{k+1} = \pi_g x_k + (1-\pi_g)u_k. \tag{6.4}$$

Equation (6.3) is referred to as the *measurement equation* and (6.4) as the *state-transition equation*, and together they form the state-space representation [111, Ch. 3.3] of the CD system (6.1). After defining the matrices $A_x \coloneqq \pi_g I$, $B_x \coloneqq (1-\pi_g)I$ and $C_x \coloneqq R$, the state-space system reads as

$$x_{k+1} = A_x x_k + B_x u_k, \tag{6.5a}$$

$$y_k = C_x x_{k-n_\tau} + d_k, \tag{6.5b}$$

and by augmenting the state as

$$\bar{x}_k \coloneqq \begin{pmatrix} x_k^{\mathrm{T}} & x_{k-1}^{\mathrm{T}} & \dots & x_{k-n_\tau}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^{\bar{n}_x}, \tag{6.6}$$

where $\bar{n}_x \coloneqq n_u(1+n_\tau)$, the state-space system can be rewritten in delay-free form:

$$\bar{x}_k = A_{\bar{x}}\bar{x}_{k-1} + B_{\bar{x}}u_{k-1}, \tag{6.7a}$$

$$y_k = C_{\bar{x}}\bar{x}_k + d_k, \tag{6.7b}$$

where the matrices $A_{\bar{x}} \in \mathbb{R}^{\bar{n}_x \times \bar{n}_x}$, $B_{\bar{x}} \in \mathbb{R}^{\bar{n}_x \times \bar{n}_x}$ and $C_{\bar{x}} \in \mathbb{R}^{n_y \times \bar{n}_x}$ are given by

$$A_{\bar{x}} \coloneqq \begin{bmatrix} \pi_g I & 0 & \dots & 0 \\ I & & & \vdots \\ & \ddots & & \\ & & I & 0 \end{bmatrix}, \quad B_{\bar{x}} \coloneqq \begin{bmatrix} 1-\pi_g \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad C_{\bar{x}} \coloneqq \begin{bmatrix} 0 & \dots & 0 & R \end{bmatrix}. \tag{6.8}$$

6.1.2 Problem Formulation

Given a discrete-time linear dynamical system and an initial condition $x(t)$ at time t, a standard MPC scheme computes a control law by predicting the future evolution of the system and minimising a quadratic objective function over some planning horizon N. This can be achieved via repeated solution of the following

quadratic program (QP):

$$\begin{aligned} \underset{\substack{x_1,\dots,x_N\in\mathbb{R}^{n_u}\\ u_0,\dots,u_{N-1}\in\mathbb{R}^{n_u}}}{\text{minimise}} \quad & \sum_{k=0}^{N-1} \|x_k - x_{\text{sp}}\|_Q^2 + \|u_k - u_{\text{sp}}\|_{R_u}^2 + \|x_N - x_{\text{sp}}\|_P^2, \\ \text{subject to} \quad & x_{k+1} = A_x x_k + B_x u_k, \quad k = 0,\dots,N-1, \\ & x_0 = x(t), \\ & \begin{pmatrix} u_0^{\mathrm{T}} & \dots & u_{N-1}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \in \mathcal{U}(u(t-1)), \end{aligned} \tag{6.9}$$

where $\|x_k\|_{(\cdot)}^2 := x_k^{\mathrm{T}}(\cdot)x_k$, returning at each step the optimal first input stage $u_0^\star = u(t)$ as a control law. The inputs $u_k \in \mathbb{R}^{n_u}$ are constrained to the closed convex set $\mathcal{U}(u(t-1)) \subset \mathbb{R}^{n_u}$, which is determined by the input slew-rate and amplitude constraints and addressed in Section 6.3. The state estimate $x(t)$ and the setpoints x_{sp} and u_{sp}, which are introduced for offset-free control, are obtained from an observer that is reverse engineered from the existing Diamond controller in Section 6.4. It is assumed that no constraints are imposed on the states $x_k \in \mathbb{R}^{n_x}$. The terminal cost matrix $P = P^{\mathrm{T}} \succ 0$ is obtained from the discrete-time algebraic Riccati equation (DARE) associated to the unconstrained infinite horizon regulator problem [168, Ch. 21.3]. The QP (6.9) has a unique solution if $R_u \succ 0$, $Q \succeq 0$ and the pairs (A_x, B_x) and $(A_x, Q^{\frac{1}{2}})$ are controllable and observable, respectively [17, Chapter 12]. Note that compared to the state-space system (6.5), the time delay does *not* appear in (6.9), but is considered by the observer instead.

By stacking the optimisation variables as

$$\mathbf{x} := \begin{pmatrix} x_1^{\mathrm{T}} & \dots & x_N^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}, \qquad \mathbf{u} := \begin{pmatrix} u_0^{\mathrm{T}} & \dots & u_{N-1}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}, \tag{6.10}$$

the state-transition equations $x_{k+1} = A_x x_k + B_x u_k$ can be rewritten in the form

$$\mathbf{x} = \mathbf{A}\mathbf{u} + \mathbf{B}x(t), \tag{6.11}$$

where

$$\mathbf{A} := \begin{bmatrix} 0 & \dots & & 0 \\ B_x & & & \vdots \\ A_x B_x & B_x & & \\ \vdots & & \ddots & 0 \\ A_x^{N-1}B_x & A_x^{N-2}B_x & \dots & B_x \end{bmatrix}, \qquad \mathbf{B} := \begin{bmatrix} I \\ A_x \\ A_x^2 \\ \vdots \\ A_x^N \end{bmatrix}. \tag{6.12}$$

By substituting (6.11) in (6.9), the states $x_1, \ldots, x_N$ can be eliminated from the MPC formulation, producing the equivalent condensed problem

$$\begin{aligned} \underset{\mathbf{u}\in\mathbb{R}^{Nn_u}}{\text{minimise}} \quad & \tfrac{1}{2}\mathbf{u}^{\mathrm{T}}\mathbf{J}\mathbf{u} + (\mathbf{q}(x(t)))^{\mathrm{T}}\mathbf{u}, \\ \text{subject to} \quad & \mathbf{u} \in \mathcal{U}(u(t-1)), \end{aligned} \tag{6.13}$$

where $\mathbf{J}$, which is referred to as the *Hessian*, and $\mathbf{q}(x(t))$ are defined as

$$\mathbf{J} \coloneqq \mathbf{A}^{\mathrm{T}}((I_N \otimes Q) \oplus P)\mathbf{A} + (I_N \otimes R_u), \tag{6.14a}$$

$$\mathbf{q}(x(t)) \coloneqq \mathbf{A}^{\mathrm{T}}((I_N \otimes Q) \oplus P)\mathbf{B}\,x(t) - \mathbf{A}^{\mathrm{T}}\begin{bmatrix} \mathbf{1}_N \otimes Q \\ P \end{bmatrix} x_{\mathrm{sp}} - (\mathbf{1}_N \otimes R_u)u_{\mathrm{sp}}, \tag{6.14b}$$

with $\otimes$ and $\oplus$ denoting the Kronecker product and block-diagonal concatenation, respectively, I_N the identity matrix of size $N \times N$, $\mathbf{1}_N$ a vector of ones of length N and where $\mathbf{J} = \mathbf{J}^{\mathrm{T}} \succ 0$ by the assumptions of the previous paragraphs. Note that the slew-rate constraints couple the inputs across horizon stages and the set $\mathcal{U}(u(t-1))$ depends on the previously calculated input $u(t-1)$. After finding a solution to (6.13), the set $\mathcal{U}(u(t-1))$ must therefore be updated as well as the vector $\mathbf{q}(x(t))$ on the arrival of a new measurement. In the following, the arguments of $\mathcal{U}(u(t-1))$ and $\mathbf{q}(x(t))$ will be omitted.

In contrast to (6.9) and (6.13), standard MPC formulations usually include a terminal constraint, $x_N \in \mathcal{X}_N \subset \mathbb{R}^{n_u}$, where $\mathcal{X}_N$ is chosen as a *control invariant set* [17, Def. 11.9]. Together with additional assumptions, the terminal constraint guarantees the stability when (6.9) is operated in closed loop [17, Thm. 13.2]. For the application at Diamond Light Source, this constraint is omitted as it would yield a complicated constraint set that would considerably slow down the computing speed.

6.2 Solver

To solve the constrained QP (6.13) associated with the MPC algorithm, two approaches are commonly used. The first approach uses techniques from multiparametric programming to obtain the solutions *offline* as piecewise affine functions of the initial conditions [17, Ch. 15]. However, the complexity of this approach grows exponentially with the number of states and is therefore not applicable to

the electron beam stabilisation problem at Diamond Light Source[1]. The second approach updates the QP (6.13) at every time step and solves it *online* [17, Ch. 14.2].

For the beam stabilisation problem at Diamond Light Source, the constrained QP is solved online, for which two first-order methods are considered: ADMM [20] and the *fast gradient method* (FGM) [110, Ch. 6.1.3]. For solving the constrained QP (6.13), both ADMM and FGM require a projection onto the constraint set $\mathcal{U}$. In the absence of rate constraints, this projection is usually straightforward and can be computed using a closed-form formula, e.g. projection onto a box-shaped set of upper and lower actuator limits. However, if slew-rate constraints are included then the projection is more complicated. The simplicity of the box-projection can be recovered by introducing an augmented problem form, e.g. one that includes additional state variables, such as implemented in the ADMM-based solver [132] for example. While this approach is versatile in the sense that it can cope with most reasonable sets encountered in practice, the augmentation of decision variables curtails the computation speed while increasing the memory usage. It also introduces additional equality constraints to the problem, leading to difficulties in applying methods such as the FGM. The question arises whether it is actually necessary to augment the decision variables in the particular case of constraints arising from input rate and amplitude constraints.

In Section 6.3.3, an approach is suggested that does *not* require augmenting the decision variables of the optimisation problem. By combining a closed-form solution for the projection onto a 2D constraint set with Dykstra's algorithm [21], it is shown that the projection of a vector of arbitrary finite dimension onto the space of rate-constrained signals can be found iteratively, avoiding the need for additional state or other problem variables. The projection algorithm is then embedded in a FGM, which has been shown to converge in the presence of a (bounded) projection error [114, Thm. III.5]. The following subsections summarise ADMM and FGM, and the performance of these algorithms applied to the constrained QP (6.13) is compared in Section 6.3.

[1] Alternatively, *approximate* explicit MPC [17, Ch. 14] could be used.

6.2.1 Fast Gradient Method

The FGM belongs to the family of first-order methods that seek solutions of convex optimisation problems using only the first derivative of the objective function of (6.13). A formulation of this algorithm using the *constant step scheme II* [110, Ch. 2.2] is presented in [72] and repeated in Algorithm 6.1. Lines marked with circled arrows (🔄) denote the number of synchronisation steps in the parallelised implementation (Section 6.6) and the observer on Line 1 of Algorithm 6.1 is introduced in Section 6.4.3. The fixed step size $\beta = (\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}})/(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})$ is based on the minimum and maximum curvature of the convex objective function, implying that the objective must be strongly convex. For (6.13), finding these values amounts to computing the minimum and maximum eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$ of $\mathbf{J} \succ 0$, respectively. The projection operator $\mathcal{P}_{\mathcal{U}}$ for the set $\mathcal{U}$ is analysed in Section 6.3.

In order to reduce the computation effort and complexity, no termination criterion will be used and the algorithm is instead run for a fixed number of iterations $I_{\max}$. Based on the convergence rate results for the FGM, a maximum number of iterations $I_{\max}$ can be derived that guarantees a certain level of suboptimality for all initial states $x_0 = x(t)$ within a bounded set [122].

Algorithm 6.1 FGM applied to MPC problem (6.13).

Input: Previous input $u(t-1)$
Output: $u(t) = w_{I_{max}}$

1: Update observer 2× 🔄
2: Update $\mathcal{U}$ and $\mathbf{q}$ and set $v_1 = w_1 = 0$ 1× 🔄
3: **for** $i = 0$ to $I_{\max} - 1$ **do**
4: $\quad t_i = (I - \mathbf{J}/\lambda_{\max})v_i - \mathbf{q}/\lambda_{\max}$ 1× 🔄
5: $\quad w_{i+1} = \mathcal{P}_{\mathcal{U}}(t_i)$
6: $\quad v_{i+1} = (1+\beta)w_{i+1} - \beta w_i$
7: **end for** 1× 🔄

6.2.2 Alternating Direction of Multipliers Method

The ADMM algorithm belongs to the class of *augmented Lagrangian* methods and is, like FGM, a first-order method. ADMM algorithms are based on repeatedly

minimising the augmented Lagrange function w.r.t. the primal variables and maximising the same function w.r.t. to the dual variables [18, Chapter 5]. Assuming that the input constraint set $\mathcal{U}$ can be represented as a polyhedron, i.e.

$$\mathcal{U} = \left\{ \mathbf{u} \in \mathbb{R}^{Nn_u} \mid \underline{v} \leq K\mathbf{u} \leq \bar{v} \right\} \tag{6.15}$$

the optimisation problem (6.13) can be reformulated as

$$\begin{aligned} \underset{\substack{\mathbf{u}\in\mathbb{R}^{Nn_u},\\ v\in\mathbb{R}^{n_v}}}{\text{minimise}} \quad & \tfrac{1}{2}\mathbf{u}^{\mathrm{T}}\mathbf{J}\mathbf{u} + \mathbf{q}^{\mathrm{T}}\mathbf{u}, \\ \text{subject to} \quad & K\mathbf{u} - v = 0, \\ & \underline{v} \leq v \leq \bar{v}, \end{aligned} \tag{6.16}$$

where the constraint variables $v \in \mathbb{R}^{n_v}$ and equality constraints $K\mathbf{u} - v = 0$ were introduced. The augmented Lagrangian for (6.16) can be written as

$$L(\mathbf{u}, v, \gamma) = \frac{1}{2}\mathbf{u}^{\mathrm{T}}\mathbf{J}\mathbf{u} + \mathbf{q}^{\mathrm{T}}\mathbf{u} + \frac{\rho}{2}\|K\mathbf{u} - v\|_2^2 + \gamma^{\mathrm{T}}(K\mathbf{u} - v) + \mathcal{I}_{[\underline{v},\bar{v}]}(v), \tag{6.17}$$

where $\mathcal{I}_{[\underline{v},\bar{v}]} : \mathbb{R}^{n_v} \mapsto \mathbb{R}_+$ is the indicator function [18, Ex. 3.1] for the set $\mathcal{V} = \{v \mid \underline{v} \leq v \leq \bar{v}\}$ and the penalty parameter $\rho \in \mathbb{R}_{++}$ and the dual variables γ are associated with the constraint $K\mathbf{u} - v = 0$. A standard ADMM scheme solves (6.16) by repeatedly minimising (6.17) w.r.t. $\mathbf{u}$ and v and updating the dual variables γ using an approximate gradient ascent method. The algorithm is summarised in Algorithm 6.2, where the saturation function $\mathrm{sat}_{[\underline{v},\bar{v}]}(\cdot)$ was used, which limits its argument to $\underline{v}$ and $\bar{v}$. Reformulation (6.16) simplifies the projection involved in the algorithm: Instead of projecting onto the polyhedron $\mathcal{U}$, which might be as hard as solving (6.13), the projection onto $\mathcal{V}$ is given by the saturation function. The critical distinction between the optimisation problem in the form (6.13) (and its solution via FGM) and (6.16) (and its solution via ADMM) is that the FGM form (6.13) has a positive definite $\mathbf{J}$ and does not allow for equality constraints. The ADMM form (6.16) makes neither restriction.

Algorithm 6.2 ADMM applied to MPC problem (6.9) with constraint set (6.15).

Input: Previous input $u(t-1)$
Output: $u(t) = v_{I_{max}}$
1: Update observer
2: Update $\underline{v}$, $\bar{v}$ and $\mathbf{q}$ and set $\gamma_0 = 0$
3: **for** $i = 1$ to $I_{\max}$ **do**
4: Solve for t_i:
$(\mathbf{J} + \rho K^{\mathrm{T}} K) t_i = K^{\mathrm{T}}(\rho v_{i-1} - \gamma_{i-1}) - \mathbf{q}$
5: $v_i = \mathrm{sat}_{[\underline{v},\bar{v}]}\{K t_i + \rho^{-1}\gamma_{i-1}\}$
6: $\gamma_i = \gamma_{i-1} + \rho(K t_i - v_i)$
7: **end for**

6.3 Input Constraint Projection Method

Given a nonempty closed convex set $\mathcal{U} \subseteq \mathbb{R}^N$, the Euclidean projection $\mathbf{u}^*$ of a point $\mathbf{u}^\circ \in \mathbb{R}^N$ is defined as the minimiser of the following optimisation problem:

$$\mathbf{u}^* = \arg\min_{\mathbf{u}\in\mathcal{X}} \|\mathbf{u} - \mathbf{u}^\circ\|_2^2. \tag{6.18}$$

By the assumptions on the set $\mathcal{U}$, the optimisation problem (6.18) admits a unique solution [11, Chapter 3.2]. In the following, the notation $\mathbf{u}^* =: \mathcal{P}_{\mathcal{U}}(\mathbf{u}^\circ)$ will be used as shorthand for projecting a point $\mathbf{u}^\circ$ onto the set $\mathcal{U}$.

6.3.1 Rate and Amplitude Constraint Set

Given a maximum allowable input amplitude $\bar{a} \in \mathbb{R}^{n_u} > 0$ and rate $\bar{r} \in \mathbb{R}^{n_u} > 0$, define the amplitude constraint set as

$$\mathcal{A}_k := \left\{ \mathbf{u} = \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix} \in \mathbb{R}^{n_u(N-1)} \;\middle|\; |u_k| \le \bar{a} \right\}, \tag{6.19}$$

and the rate constraint set as

$$\mathcal{R}_k := \left\{ \mathbf{u} = \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix} \in \mathbb{R}^{n_u(N-1)} \;\middle|\; |u_k - u_{k-1}| \le \bar{r} \right\}, \tag{6.20}$$

for $k = 0, \ldots, N-1$ and where the inequalities are applied element-wise. Note that for $k = 0$, the set $\mathcal{R}_0$ is a function of the input $u_{-1} = u(t-1)$ that is treated as a fixed constant stemming from the actual input of the system at time $(k-1)T_s$. The trivial

case where $\mathcal{A}_k$ is entirely contained in $\mathcal{R}_k$ is excluded by assuming that $0 \le \bar{r} \le 2\bar{a}$ for at least one element. The input rate and amplitude constraint set for problem (6.9) is obtained as the intersection of $\mathcal{A} := \mathcal{A}_0 \cap \cdots \cap \mathcal{A}_{N-1}$ and $\mathcal{R} := \mathcal{R}_0 \cap \cdots \cap \mathcal{R}_{N-1}$, i.e.

$$\mathcal{U} := \left\{ \mathbf{u} = \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix} \in \mathbb{R}^{n_u(N-1)} \;\middle|\; |u_k| \le \bar{a} \;\forall\; k = 0, \ldots, N-1, \; |u_k - u_{k-1}| \le \bar{r} \;\forall\; k = 0, \ldots, N-1 \right\}. \tag{6.21}$$

Because the constraints are not coupled among the elements of $u_k \in \mathbb{R}^{n_u}$, it will be assumed that $n_u = 1$ for clarity of exposition. However, the following results apply in the case that $n_u > 1$.

While the projection onto $\mathcal{A}$ is given by saturating the elements of $\mathbf{u}$ to $\pm\bar{a}$, no tractable closed-form solution for $\mathcal{P}_\mathcal{R}$ is known to the author (see also [12]) and hence also not for $\mathcal{P}_\mathcal{U}$. One approach to obtain $\mathcal{P}_\mathcal{R}(\mathbf{u}^\circ)$ or directly $\mathcal{P}_\mathcal{U}(\mathbf{u}^\circ)$ is to define a multi-parametric program [17, Ch. 2] with parameters $\mathbf{u}^\circ$ and $u(t-1)$. The multi-parametric solution of the projection results in a piecewise affine function (PWA) of the parameters $u(t-1)$ and $\mathbf{u}^\circ$, i.e. n affine functions defined on n disjoint sets. An explicit solution for larger horizons could be computed using dedicated software, e.g. [64], but in view of the implementation it is solved graphically in Section 6.3.2.

6.3.2 2-Dimensional Projection

Consider the input rate and amplitude constraint set (6.21) for $n_u = 1$ and $N = 2$, which can be represented as

$$\mathcal{U}_1 := \left\{ \mathbf{u} = \begin{pmatrix} u_0 \\ u_1 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; \underline{a}_0 \le u_0 \le \bar{a}_0, \; |u_1| \le \bar{a}, \; |u_1 - u_0| \le \bar{r} \right\}, \tag{6.22}$$

where $\underline{a}_0 := \max(-\bar{a}, -\bar{r} + u(t-1))$ and $\bar{a}_0 := \min(\bar{a}, \bar{r} + u(t-1))$. The set is illustrated in Fig. 6.1, where a coordinate system (ν_k, ν_{k-1}) rotated by 45° and different regions $\mathcal{A}_i$, $\mathcal{B}_i$ and $\mathcal{C}_i$ and the corner points c_i have been added. When $\bar{r}$ and $\bar{a}$ are fixed, the shape of set (6.21) depends on parameter $u(t-1)$. As $u(t-1)$ changes, the corner points c_i are moved along the rate constraint diagonals until they eventually stop at an amplitude constraint boundary. In addition, regions B_i might lose their

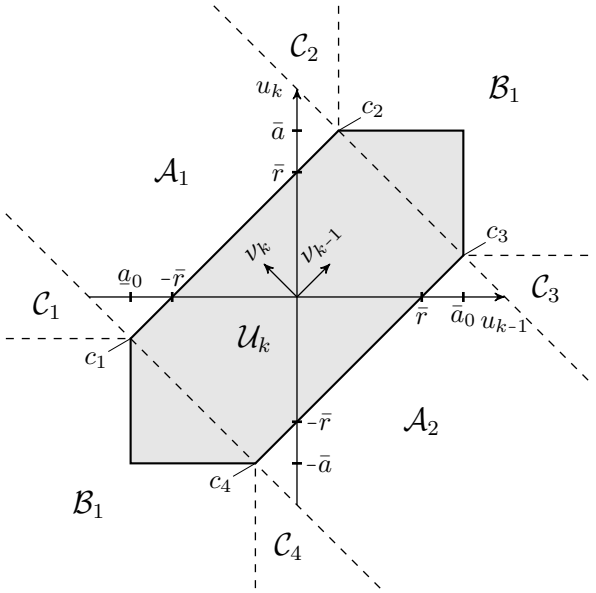


Figure 6.1: Input rate and amplitude constraint set for the 2-dimensional projection as defined in (6.22). The regions $\mathcal{A}_i$, $\mathcal{B}_i$ and $\mathcal{C}_i$ define different maps used to project onto the shaded set.

horizontal facet and regions A_i vanish. Assume a point $\mathbf{u}^\circ = \begin{pmatrix} u_0^\circ \\ u_1^\circ \end{pmatrix}$ is to be projected. The projection consists of two steps: Firstly, determine to which region the point belongs. Secondly, apply the projection function of that particular region. In other words, the PWA solution to the problem of projection onto the set $\mathcal{U}_1$ is explicitly identified following the general method of [17, Chapter 2].

After representing $\mathbf{u}^\circ$ in the (ν_k, ν_{k-1})-basis, the point location problem can be solved using Fig. 6.1 and by comparing the ν_{k-1}-component with the projection of the corner points c_i onto the ν_{k-1}-diagonal determined by $\bar{r}$, $\bar{a}$ and u_{-1}. If the p-component is beyond these limits, the u_0- or u_1-components can be compared with the corner points c_i to complete the point location problem.

If the point lies in one of the corner regions $\mathcal{C}_i$, then it is mapped to the corresponding corner c_i. If the point lies in one of the diagonal regions $\mathcal{A}_i$, then it can be represented in terms of the rotated basis (ν_k, ν_{k-1}), its ν_k-component saturated to $\pm\bar{r}/\sqrt{2}$ and rotated back. Finally, if the point lies in one of the box-regions $\mathcal{B}_i$, then the formula for a box-projection can be applied [11, Chapter 28.3] using limits $(\underline{a}_0, \bar{a}_0)$ and $(\underline{a}_1, \bar{a}_1)$ for the u_0- and u_1-direction, respectively, where

$\underline{a}_1 = \max(-\bar{a}, -\bar{r} + \underline{a}_0)$ and $\bar{a}_1 = \min(\bar{r}, \bar{r} + \bar{a}_0)$. A C-language implementation of the 2-dimensional projection is provided in [75].

6.3.3 Dykstra's Algorithm

Dykstra's algorithm [21] was first published in 1983 as an extension to Von Neumann's *alternating projections method* [161], which is suitable for finding a point lying in the intersection $\mathcal{U} = \mathcal{U}_1 \cap \ldots \cap \mathcal{U}_N$ of N closed convex sets $\mathcal{U}_i$ by cyclically projecting onto the sets $\mathcal{U}_i$. While Von Neumann's algorithm only finds *some* point in $\mathcal{U}$, Dykstra's algorithm determines the Euclidean projection $\mathcal{P}_{\mathcal{U}}(\mathbf{u}^\circ)$ of $\mathbf{u}^\circ$ onto $\mathcal{U}$. Both algorithms circumvent the potentially complicated projection $\mathcal{P}_{\mathcal{U}}$ by iteratively applying the (known) projections $\mathcal{P}_{\mathcal{U}_i}$.

Dykstra's algorithm is summarised in Algorithm 6.3 for the case that $\mathcal{U} = \mathcal{U}_e \cap \mathcal{U}_o$. The algorithm is initialised on Line 1 and then proceeds by successively projecting onto $\mathcal{U}_e$ and $\mathcal{U}_o$ on Lines 3 and 5. What distinguishes Dykstra's algorithm from Von Neumann's is the choice of variables μ_i and γ_i that track the residuals from projecting onto $\mathcal{U}_e$ and $\mathcal{U}_o$ [150]. It can be shown that Algorithm 6.3 always converges to the Euclidean projection onto $\mathcal{U}$, provided that the sets $\mathcal{U}_e$ and $\mathcal{U}_o$ are closed convex sets and their intersection is nonempty [11], [21]. However, as Dykstra's method will be used in a practical context and therefore run for a finite number of iterations, the output of Algorithm 6.3 is referred to as $\tilde{\mathcal{P}}_{\mathcal{U}}(\mathbf{u}^\circ)$ highlighting a potential projection error $\left\|\tilde{\mathcal{P}}_{\mathcal{U}}(\mathbf{u}^\circ) - \mathcal{P}_{\mathcal{U}}(\mathbf{u}^\circ)\right\|_2 \geq 0$.

Algorithm 6.3 Dykstra's Algorithm for two sets and with fixed iteration number.

Input: $\mathbf{u}^\circ$
Output: $\tilde{\mathcal{P}}_{\mathcal{U}}(\mathbf{u}^\circ) = x_{i+1}$

1: Set $x_0 = \mathbf{u}^\circ$, $\mu_0 = 0$ and $\gamma_0 = 0$
2: **for** $i = 0$ to $I_{\max}$ **do**
3: $\quad y_i = \mathcal{P}_{\mathcal{U}_e}(x_i + \mu_i)$
4: $\quad \mu_{i+1} = \mu_i + x_i - y_i$
5: $\quad x_{i+1} = \mathcal{P}_{\mathcal{U}_o}(y_i + \gamma_i)$
6: $\quad \gamma_{i+1} = \gamma_i + y_i - x_{i+1}$
7: $\quad$ **if** $\|x_{i+1} - x_i\|_\infty < \epsilon$ **then**
8: $\quad\quad$ **break**
9: $\quad$ **end if**
10: **end for**

Before applying Algorithm 6.3 to the rate and amplitude constraint set (6.21), it remains to show that the set (6.21) can be formulated as the intersection of two closed convex sets with known projection operators. In Section 6.3.2, it was demonstrated how to project onto (6.21) for $N = 2$ using a geometrical approach. For horizons $N \geq 2$, set (6.21) can be represented as the intersection of $N-1$ closed convex sets $\mathcal{U}_k$, where

$$\mathcal{U}_1 := \left\{ \mathbf{u} = \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix} \in \mathbb{R}^{n_u(N-1)} \;\middle|\; \underline{a}_0 \leq u_0 \leq \bar{a}_0,\ |u_1| \leq \bar{a},\ |u_1 - u_0| \leq \bar{r} \right\}, \tag{6.23a}$$

$$\mathcal{U}_k := \left\{ \mathbf{u} = \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix} \in \mathbb{R}^{n_u(N-1)} \;\middle|\; |u_k| \leq \bar{a},\ |u_{k-1}| \leq \bar{a},\ |u_k - u_{k-1}| \leq \bar{r} \right\}, \tag{6.23b}$$

and $k = 2, \ldots, N-1$. Let π_k denote the 2-dimensional projection from Section 6.3.2 applied element-wise to the n_u elements of u_{k-1} and u_k, respectively, with $\bar{a}_0 = -\underline{a}_0 = \bar{a}$ for $k > 1$. Let π_k^1 and π_k^2 be the resulting n_u projections of u_{k-1} and u_k, respectively. Then $\mathcal{P}_{\mathcal{U}_k} \in \mathbb{R}^{n_u(N-1)}$ can be written as

$$\mathcal{P}_{\mathcal{U}_k}(\mathbf{u}) = \begin{pmatrix} u_0^{\mathrm{T}} & u_1^{\mathrm{T}} & \ldots & (\pi_k^1)^{\mathrm{T}} & (\pi_k^2)^{\mathrm{T}} & \ldots & u_{N-1}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}, \tag{6.24}$$

where elements u_{k-1} and u_k have been replaced by π_k^1 and π_k^2, respectively. Assume for the purpose of explanation that N is even and let $\mathcal{U}_e$ and $\mathcal{U}_o$ denote the intersection of sets $\mathcal{U}_i$ grouped by even and odd indices, respectively. The projections $\mathcal{P}_{\mathcal{U}_e}(\mathbf{u}^\circ) \in \mathbb{R}^{n_u(N-1)}$ and $\mathcal{P}_{\mathcal{U}_o}(\mathbf{u}^\circ) \in \mathbb{R}^{n_u(N-1)}$ are obtained by combining the corresponding projections from (6.24) and given by

$$\mathcal{P}_{\mathcal{U}_e}(\mathbf{u}) = \begin{pmatrix} u_0^{\mathrm{T}} & (\pi_2^1)^{\mathrm{T}} & (\pi_2^2)^{\mathrm{T}} & (\pi_4^1)^{\mathrm{T}} & \ldots & (\pi_{N-2}^2)^{\mathrm{T}} & u_{N-1}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}, \tag{6.25a}$$

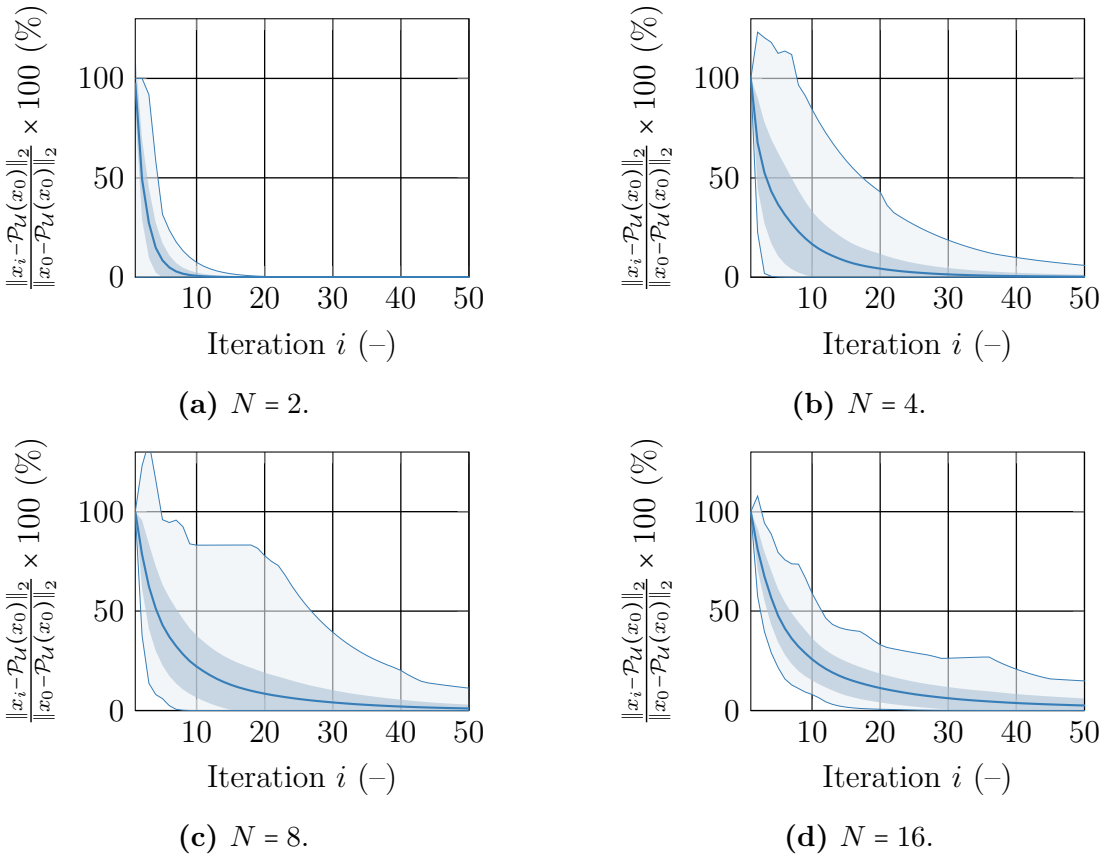
$$\mathcal{P}_{\mathcal{U}_o}(\mathbf{u}) = \begin{pmatrix} (\pi_1^1)^{\mathrm{T}} & (\pi_1^2)^{\mathrm{T}} & (\pi_3^1)^{\mathrm{T}} & (\pi_3^2)^{\mathrm{T}} & \ldots & (\pi_{N-1}^1)^{\mathrm{T}} & (\pi_{N-1}^2)^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}}. \tag{6.25b}$$

By using (6.25), Algorithm 6.3 can be applied to project onto the input rate and amplitude constraint set (6.21). The choice of using a 2-dimensional projection in combination with Dykstra's method is mainly motivated by the geometrical approach of Section 6.3.2. Another possibility would be to compute a formula for a 3-dimensional projection $\tilde{\pi}_k$ onto the set $\mathcal{U}_k \cap \mathcal{U}_{k+1}$ for $n_u = 1$ using one

of the methods outlined in Section 6.3.1. Compared to the 2-dimensional case, fewer $\tilde{\pi}_k$ would have to be evaluated at the expense of increased complexity. How this would affect the computational performance and the convergence rate of the method is not clear a-priori [21]. Note that as opposed to (6.24), with which Dykstra's method would require the sequential execution of $\mathcal{P}_{\mathcal{U}_k}$, partitioning (6.25) is suitable for a parallel implementation.

While in [21] it is shown that Dykstra's method eventually converges to the Eucledian projection, the convergence rate of the algorithm has not been analysed for $\mathcal{U}$ (6.21) and the partitioning (6.25). The convergence rate is analysed in [36] for the case that the set $\mathcal{U}$ is given as the intersection of r halfspaces $\mathcal{H}_i$. In this case, there exist constants $0 \leq c < 1$ and $\rho > 0$ such that [36, Thm. 3.8] $\|x_i - \mathcal{P}_{\mathcal{H}}(x^\circ)\|_2 \leq \rho c^i$ [36, Thm. 3.8], but the constant ρ cannot be computed in advance and depends on x°. In fact, in [13] it is shown that Dykstra's method can *stall* for an arbitrary number of iterations for certain x°, so that $\rho \to \infty$.

Fig. 6.2 compares the output of Algorithm 6.3 applied to the input rate and amplitude constraint set (6.21) to the solution $\mathcal{P}_{\mathcal{U}}(x_0)$ obtained using an interior-point method for different horizons $N \in \{2, 4, 8, 16\}$ and $u_{-1} = 0$. The rate and amplitude parameters are chosen as $\bar{r} = 1$ and $\bar{a} = 5$, which corresponds to the values of the Diamond application. Fig. 6.2 shows the mean of the relative error $\frac{\|x_i - \mathcal{P}_{\mathcal{U}}(x_0)\|_2}{\|x_0 - \mathcal{P}_{\mathcal{U}}(x_0)\|_2}$, the mean plus or minus the standard deviation (dark shade) and the minimum/maximum value (light shade) for 100 starting points x_0 that were drawn from a normal distribution $\mathcal{N}(0, 100)$. While the mean values decreases proportionally to c^i for some constant $0 < c < 1$ for all $N \in \{2, 4, 8, 16\}$, the stalling of Dykstra's method is evident in Fig. 6.2c for $i = 1, \ldots, 18$. In Section 6.3.4, Dykstra's method is nevertheless combined with the FGM to demonstrate the performance benefit, bearing in mind that the FGM also converges using an inexact projection [114, Thm. III.5].



(a) $N = 2$. **(b)** $N = 4$.

(c) $N = 8$. **(d)** $N = 16$.

Figure 6.2: Application of Dykstra's method to the set (6.21) with $\bar{r} = 1$, $\bar{a} = 5$, $u_{-1} = 0$ and $N \in \{2, 4, 8, 16\}$. The subfigures show the mean value, the standard deviation and the minimum/maximum value of the relative error $\frac{\|x_i-\mathcal{P}_\mathcal{U}(x_0)\|_2}{\|x_0-\mathcal{P}_\mathcal{U}(x_0)\|_2}$ of the iterate x_i of Algorithm 6.3 for 100 starting points x_0, which were drawn from a random normal distribution with zero mean and a standard deviation of 10.

6.3.4 Numerical Studies

For solving the QP (6.13) associated with the MPC formulation, the FGM and Dykstra's projection method are combined in Algorithm 6.4 by substituting $\tilde{\mathcal{P}}_\mathcal{U}$ obtained from Algorithm 6.3 for the exact projection $\mathcal{P}_\mathcal{U}$ on Line 5 of Algorithm 6.1. Compared to the ADMM implementation, Algorithm 6.4 bears several advantages. Dykstra's algorithm makes the augmentation of decision variables superfluous. While the ADMM formulation (6.16) requires $2N - 1$ decision variables, Algorithm 6.4 reduces the number of decision variables to N. This not only greatly reduces the computation time, but also lowers the memory footprint. If it is assumed that all matrices are dense and neglect the storage of vectors, then Algorithm 6.4

reduces the memory footprint by approximately $N^2/(N^2 + (N-1)^2 n_v/n_u)$, which tends to a reduction of 50 % for set (6.21) and large N. Dykstra's algorithm barely introduces any memory footprint because it only involves Boolean operations and vector additions. Moreover, arrays allocated by Algorithm 6.1 can be used as temporary placeholders to execute Dykstra's method. Note that Algorithms 6.2, 6.1 and 6.4 could be warm-started using the solution computed at time step $N-1$, e.g. setting $y_1 = u_{-1} = u(t-1)$ in Algorithm 6.4.

Algorithm 6.4 FGM and Dykstra's algorithm applied to MPC problem (6.13) with set (6.21).

Input: Initial state $x(t)$, previous input $u(t-1)$
Output: $u(t) = w_{I_{max}}$

1: Update observer 2× ↻
2: Update $\mathcal{U}_e$, $\mathcal{U}_o$ and $\mathbf{q}$ and set $v_1 = w_1 = 0$ 1× ↻
3: **for** $i = 0$ to $I_{\max} - 1$ **do**
4: $\quad t_i = (I - \mathbf{J}/\lambda_{\max})v_i - \mathbf{q}/\lambda_{\max}$ 1× ↻
5: $\quad w_{i+1} = \tilde{\mathcal{P}}_{\mathcal{U}}(t_i)$ 20× ↻
6: $\quad v_{i+1} = (1+\beta)w_{i+1} - \beta w_i$ 1× ↻
7: **end for** 1× ↻

To show its speed benefits, Algorithm 6.4 is implemented in C [76] and compared on a desktop computer (Intel i7-7700 CPU @ 3.1 GHz, 8 GB, single-core) against the Operator Splitting Quadratic Program (OSQP) solver [132], which is a C language solver that is based on ADMM and uses an augmentation of decision variables to simplify the projection as in Algorithm 6.2. In order to avoid refactoring the matrix on the left-hand side of Line 4 of Algorithm 6.2, the benchmark ADMM implementation uses a constant penalty parameter ρ. Neither of the C programs includes a non-standard C-library.

Fig. 6.3 compares the average execution times of one iteration of the C-language implementation of the ADMM (——) and the FGM (- - -) applied to problem (6.13) with $n_u = 1$ and the input rate and amplitude constraint set $\mathcal{U}$ as defined in (6.21) with $\bar{r} = \bar{a} = 1$. The OSQP solver uses a matrix-factorisation to solve the linear system on Line 4 of Algorithm 6.2 and the time for factorising the matrix is excluded from Fig. 6.3. The problem data $(\mathbf{J}, \mathbf{q})$ is randomly generated and the

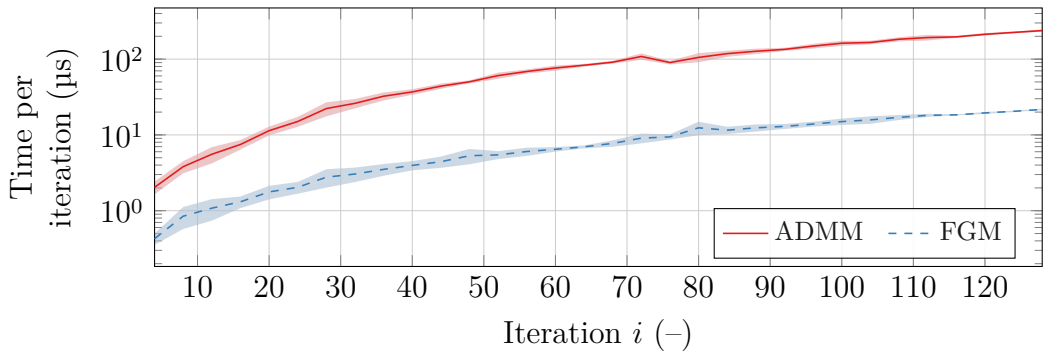
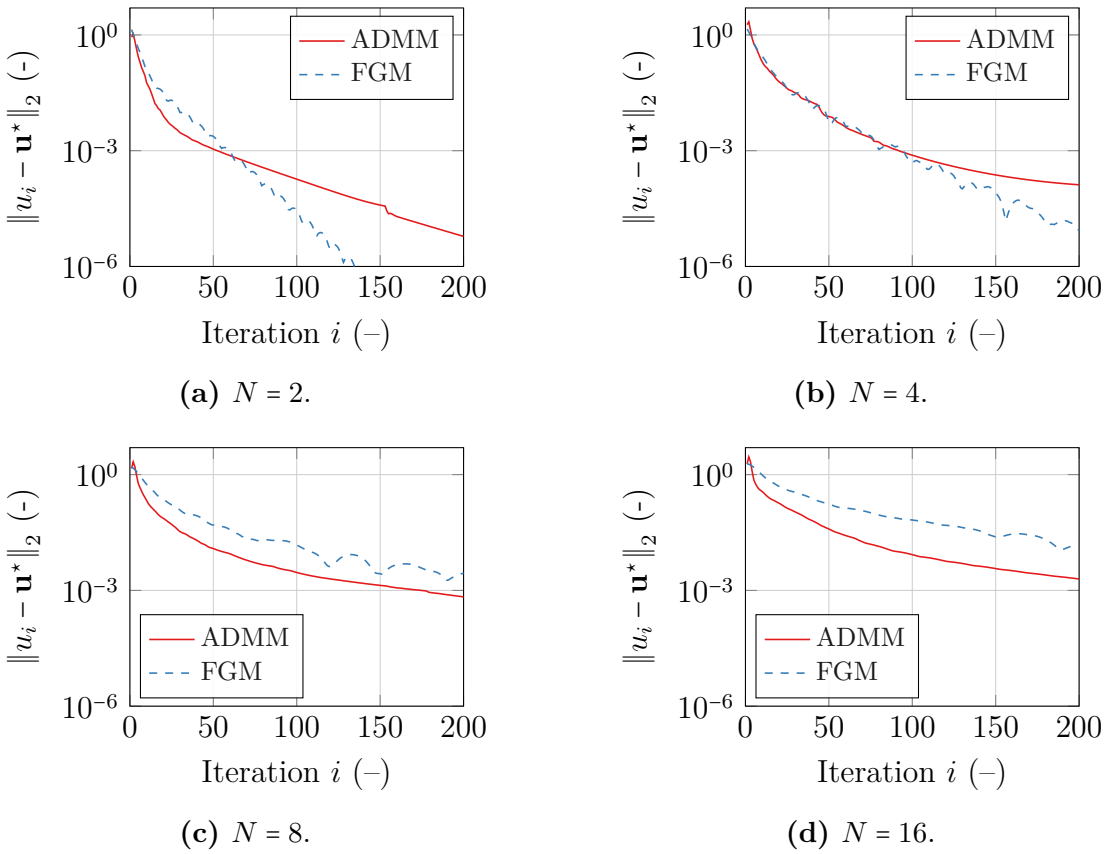


Figure 6.3: Comparison of the logarithmically-scaled average execution times of one iteration of Algorithm 6.2 (ADMM) and Algorithm 6.4 (FGM), respectively, applied to problem (6.13) with $n_u = 1$, set (6.21) and $\bar{r} = \bar{a} = 1$. Both algorithms were benchmarked using a stand-alone C language implementation and using 100 randomly generated problems per horizon N. The FGM includes Dykstra's method with a termination check on every 10th iterate with $\epsilon = 10^{-6}$.

average execution times are benchmarked over 100 problems per horizon. Fig. 6.3 reveals that the combination of the FGM and Dykstra's method greatly reduces the computation time for one solver iteration. The performance gain is due to the fact that Dykstra's method makes the augmentation of decision variables (6.16) unnecessary, which in case of the input rate and amplitude set amounts to tripling the number of decision variables. Because Dykstra's method involves only vector additions and Boolean operations, the projection algorithm only requires a few processor cycles. The termination criterion for Dykstra's algorithm is checked on every 10th iterate with $\epsilon = 10^{-6}$.

Fig. 6.4 compares the practical convergence behaviour of Algorithm 6.2 (ADMM, ——) and Algorithm 6.4 (FGM, - - -) applied to problem (6.13) with the input rate and amplitude constraint set as defined in (6.21) with $\bar{r} = \bar{a} = 1$ and $n_u = 1$. Depicted is the average distance between a high-accuracy solution $\mathbf{u}^\star$ calculated using an interior-point method and the solution at iteration i of Algorithm 6.2 and 6.4, respectively. The problem data $(\mathbf{J}, \mathbf{q})$ is randomly generated and the distances are averaged over 100 problems per horizon N. For Algorithm 6.4, Dykstra's method is implemented as follows: An initial verification is applied to avoid executing the algorithm for vectors t_i that lie inside set (6.21). If $t_i \notin \mathcal{U}$, Dykstra's method is run for a fixed number of iterations $\mathbf{J}_{max} = 50$.



(a) $N = 2$. **(b)** $N = 4$.

(c) $N = 8$. **(d)** $N = 16$.

Figure 6.4: Practical convergence behaviour of Algorithm 6.2 (ADMM) and Algorithm 6.4 (FGM) applied to problem (6.13) with set (6.21) for $N = \{2, 4, 8, 16\}$. The figure shows the distance between the iterates and a solution $\mathbf{u}^*$ computed using an interior-point method. The problem data is randomly generated and the distances are averaged over 100 problems.

In Fig. 6.4, it can be seen that for horizon $N = 4$, the FGM and ADMM converge at the same speed, but as N increases, the convergence behaviour of the FGM becomes worse. This can be associated with the convergence behaviour of Dykstra's method from Fig. 6.2, which shows that the projection error increases for increasing N. For larger N, the maximum iteration number of Dykstra's method would need to be increased to achieve a similar convergence as ADMM.

6.4 Observer and Regulator

As for the existing Diamond controller, the ill-conditioned plant complicates the design of the MPC algorithm (6.9), i.e. the tuning of the state weights, $Q \in \mathbb{R}^{n_x \times n_x}$, the input weights, $R_u \in \mathbb{R}^{n_u \times n_u}$, and the terminal cost matrix, $P \in \mathbb{R}^{n_x \times n_x}$, as well

as the design of the observer that is required to estimate the initial condition $x(t)$. Analogous to the IMC algorithm, the MPC algorithm produces large gains in directions that are difficult to control if the ill-conditioned ORM is not actively considered for choosing the state and inputs weights and designing the observer. As a starting point for tuning the MPC algorithm, the Diamond IMC algorithm is reverse engineered to obtain state and input weights and to design the observer.

This section is organised as follows. Subsection 6.4.1 analyses the Diamond IMC algorithm and the resulting closed-loop system in discrete-time form. Based on these results, a *linear quadratic regulator* (LQR) is designed in Subsection 6.4.2, which can be interpreted as the unconstrained form of MPC, and combined with an observer in Subsection 6.4.3. Throughout the remainder of this chapter, it will be assumed that system (6.1) is square, i.e. $n_y = n_u$. This requirement is imposed by the step-size choice of the FGM (Section 6.2.1), which requires a strongly convex objective function. In addition, a square system will result in a one-to-one mapping of the Diamond IMC dynamics to the LQR algorithm.

6.4.1 Discrete-Time Internal Model Control

The discrete-time input-output dynamics (6.1) can be mapped using the SVD of $R \in \mathbb{R}^{n_y \times n_y}$ (1.6), $R = U\Sigma V^{\mathrm{T}}$, to modal space, where the system is decoupled as

$$\tilde{y}_i(z) = \sigma_i g(z) \tilde{u}_i(z) + \tilde{d}_i(z), \qquad i = 1, \ldots, n_y, \tag{6.26}$$

with the modal variables $\tilde{y}_i : \mathbb{C} \mapsto \mathbb{C}$, $\tilde{u}_i : \mathbb{C} \mapsto \mathbb{C}$ and $\tilde{d}_i : \mathbb{C} \mapsto \mathbb{C}$ defined in (1.8). Following the continuous-time controller design procedure from Section 1.6, the control law is obtained in standard negative-feedback form as $u(z) = -Kc(z)y(z)$ with the gain matrix $K \in \mathbb{R}^{n_y \times n_y}$ given by

$$K \coloneqq V \operatorname{diag}(k_1, \ldots, k_n) U^{\mathrm{T}}, \qquad k_i \coloneqq \frac{\sigma_i}{\sigma_i^2 + \mu}. \tag{6.27}$$

In discrete-time, the scalar controller $c : \mathbb{C} \mapsto \mathbb{C}$ reads as [49]

$$c(z) \coloneqq \frac{1 - \pi_\lambda}{1 - \pi_g} \frac{z^{n_\tau}(z - \pi_g)}{z^{n_\tau + 1} - \pi_\lambda z^{n_\tau} - (1 - \pi_\lambda)}, \tag{6.28}$$

where π_g is the discrete-time pole of $g(z)$ (6.2), $\pi_\lambda \coloneqq \exp(-2\pi\lambda T_s)$ the discrete-time closed-loop pole and $\lambda = 1/(\tau_d T_s) = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$ is the desired closed-loop bandwidth. Note that $c(z)$ has a pole at $z = 1$.

With (6.28), the mode-by-mode output sensitivity $S_i : \mathbb{C} \mapsto \mathbb{C}$ is

$$S_i(z) \coloneqq \frac{1}{1 + k_i c(z)\sigma_i g(z)} = \frac{z^{n_\tau+1} - \pi_\lambda z^{n_\tau} - (1-\pi_\lambda)}{z^{n_\tau+1} - \pi_\lambda z^{n_\tau} - (1-\pi_\lambda)\left(1 - \frac{\sigma_i^2}{\sigma_i^2+\mu}\right)}. \tag{6.29}$$

By Descartes' rule of signs [29] and noting that n_τ is odd, $S_i(z)$ has one positive real pole π_i^+ that satisfies

$$(\pi_i^+)^{n_\tau}(\pi_i^+ - \pi_\lambda) = (1-\pi_\lambda)\left(1 - \frac{1}{1+\mu/\sigma_i^2}\right), \tag{6.30}$$

with $1 > \pi_i^+ > \pi_\lambda$ for $\mu > 0$ and the limiting cases $\pi_i^+ = \pi_\lambda$ for $\mu/\sigma_i^2 \to 0$ and $\pi_i^+ = 1$ for $\mu/\sigma_i^2 \to \infty$. In addition, $S_i(z)$ has one negative real pole and $(n_\tau - 1)/2$ pairs of complex conjugated poles. Fig. 6.5a shows the root locus of all closed-loop poles for $\mu/\sigma_i^2 = \{10^{-4}, 10^{-3}, \ldots, 10^{-2}\}$. For $\mu/\sigma_i^2 \to 0$, n_τ poles are at the origin and gradually move towards the unit circle as $\mu/\sigma_i^2 \to \infty$, whereas the real pole π_i^+ moves from π_λ along the real axis towards the point $1 + \mathrm{j}0$.

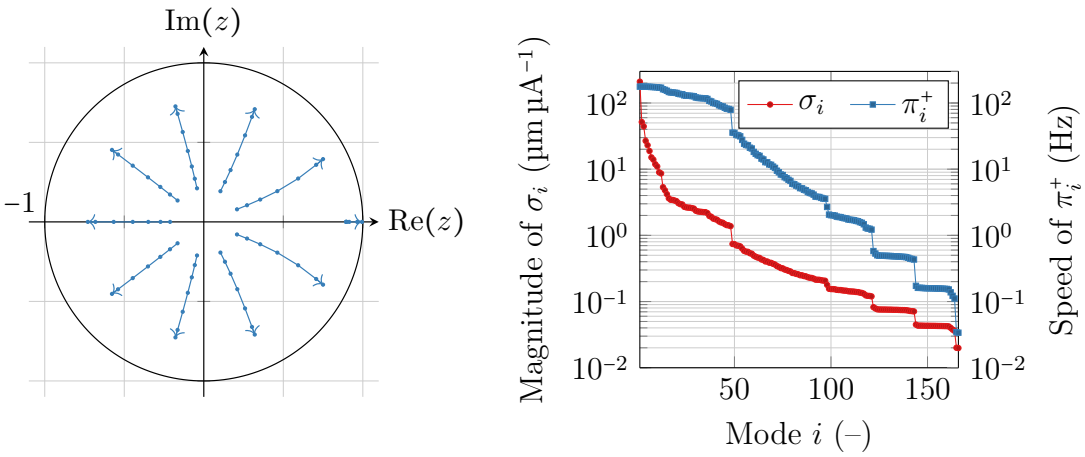
At Diamond, the regularisation parameter is fixed to $\mu = 1$. Fig. 6.5b compares the dominant pole π_i^+ of each mode against the corresponding singular value in the vertical direction, where it can be seen that the closed-bandwidth decreases with decreasing singular value. Due to the large condition number of R, the closed-loop bandwidths range from 176 Hz ($\mu/\sigma_1^2 = 210$) down to 0.03 Hz ($\mu/\sigma_{n_y}^2 = 0.02$).

6.4.2 Mode-By-Mode Linear Quadratic Regulator

As in the IMC case, the LQR is designed in mode space where a state-space representation of the modal system (6.26) is given by

$$\tilde{\bar{x}}_{k+1,i} = A_{\bar{x},i}\tilde{\bar{x}}_{k,i} + B_{\bar{x},i}\tilde{u}_{k,i}, \tag{6.31a}$$

$$\tilde{y}_{k,i} = C_{\bar{x},i}\tilde{\bar{x}}_{k,i} + \tilde{d}_{k,i}, \tag{6.31b}$$



(a) Root locus for varying μ/σ_i^2. **(b)** Singular values and dominant pole.

Figure 6.5: Left: Root locus of IMC closed loop for $\mu/\sigma_i^2 = \{10^{-4}, 10^{-3}, \ldots, 10^{2}\}$ (blue markers). Right: Singular values σ_i of R (red) and speed of closed-loop IMC poles π_i^+ (blue) for $\mu = 1$.

where subscript $i = 1, \ldots, n_y$ refers to the ith mode, $\tilde{\tilde{x}} : \mathbb{Z} \mapsto \mathbb{R}^{n_y(n_\tau+1)}$ is the delay-augmented state vector (6.6) in mode space ($\tilde{\tilde{x}}_i : \mathbb{Z} \mapsto \mathbb{R}^{n_\tau+1}$) and the state-space matrices are given by

$$A_{\bar{x},i} := \begin{bmatrix} \pi_g & 0 & \ldots & 0 \\ 1 & & & 0 \\ & \ddots & & \vdots \\ & & 1 & 0 \end{bmatrix}, \qquad B_{\bar{x},i} := \begin{bmatrix} 1-\pi_g \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \qquad C_{\bar{x},i} := \begin{bmatrix} 0 & \ldots & 0 & \sigma_i \end{bmatrix}. \tag{6.32}$$

Note that the state-space system (6.31) can be obtained from either applying the modal transformation (1.8) to the state-space representation (6.7) or from directly rewriting the input-output representation (6.26).

In contrast to IMC, the standard LQR scheme does not implement an integrator for $g(z)$ as defined in (6.2) and must be modified to achieve offset-free control. For that purpose, system (6.31) is augmented with a disturbance model [17, Ch. 13.6] and the feedback law extended with setpoints. The disturbance dynamics are modelled as

$$\tilde{d}_{k+1,i} = \tilde{d}_{k,i} + \tilde{w}_{k,i}, \tag{6.33}$$

where $\tilde{w}_{k,i}$ is zero-mean white noise and in practice, $\tilde{d}_{k,i}$ may have a non-zero initial condition, i.e. $\tilde{d}_{0,i} \neq 0$. For offset-free control, the dynamics (6.31) and the

disturbance model (6.33) are evaluated for $k \to \infty$ and $\lim_{k\to\infty} E\{\tilde{y}_{k,i}\} = 0$. Noting that $\lim_{k\to\infty} E\{\tilde{d}_{k,i}\} = \tilde{d}_{0,i}$ and denoting the setpoints for the states and inputs as

$$\tilde{x}_{\mathrm{sp},i} \coloneqq \lim_{k\to\infty} E\{\tilde{x}_{k,i}\}, \tag{6.34a}$$

$$\tilde{u}_{\mathrm{sp},i} \coloneqq \lim_{k\to\infty} E\{\tilde{u}_{k,i}\}, \tag{6.34b}$$

where $\tilde{x}_{\mathrm{sp},i} \in \mathbb{R}$ is the setpoint for the state in modal space before the augmentation (6.6) and $\bar{\tilde{x}}_{\mathrm{sp},i} \coloneqq \mathbf{1}\tilde{x}_{\mathrm{sp},i}$, the setpoints are obtained as

$$\tilde{u}_{\mathrm{sp},i} = \tilde{x}_{\mathrm{sp},i} = -\frac{1}{\sigma_i}\tilde{d}_{0,i}. \tag{6.35}$$

Using (6.35), the LQR is designed for the new variables $\delta\bar{\tilde{x}}_{k,i} \coloneqq \bar{\tilde{x}}_{k,i} - \bar{\tilde{x}}_{\mathrm{sp},i}$ and $\delta\tilde{u}_{k,i} \coloneqq \tilde{u}_{k,i} - \tilde{u}_{\mathrm{sp},i}$. According to (6.31a), the state transition of $\delta\bar{\tilde{x}}_{k,i}$ is

$$\delta\bar{\tilde{x}}_{k+1,i} = A_{\bar{x},i}\delta\bar{\tilde{x}}_{k,i} + B_{\bar{x},i}\delta\tilde{u}_{k,i}. \tag{6.36}$$

Since the delayed components of the augmented state vector $\bar{\tilde{x}}_{k,i}$ bear no new information, the state feedback is chosen as

$$\delta\tilde{u}_{k,i} = -\begin{bmatrix} k_{c,i} & 0 & \dots & 0 \end{bmatrix}\delta\bar{\tilde{x}}_{k,i}, \tag{6.37}$$

so that designing a controller for (6.36) is equivalent to designing a controller for the single-state system

$$\delta\tilde{x}_{k+1,i} = \pi_g\,\delta\tilde{x}_{k,i} + (1-\pi_g)\,\delta\tilde{u}_{k,i}, \tag{6.38a}$$

$$\delta\tilde{y}_{k,i} = \sigma_i\delta\tilde{x}_{k,i} + \tilde{d}_{k,i}, \tag{6.38b}$$

where $\delta\tilde{x}_{k,i} \coloneqq \tilde{x}_{k,i} - \tilde{x}_{\mathrm{sp},i}$ and $\delta\tilde{y}_{k,i} \coloneqq \tilde{y}_{k,i} - \sigma_i\tilde{x}_{\mathrm{sp},i}$. Substituting $\delta\tilde{u}_{k,i} = -k_{c,i}\delta\tilde{x}_{k,i}$ in (6.38a) yields the closed-loop dynamics for $\delta\tilde{x}_{k,i}$,

$$\delta\tilde{x}_{k+1,i} = (\pi_g - k_{c,i}(1-\pi_g))\,\delta\tilde{x}_{k,i}. \tag{6.39}$$

The gain $k_{c,i}$ is obtained from the corresponding infinite-horizon LQR problem [130, Ch. 9.2.1], which minimises the expected value of the cost function,

$$\mathbf{J}_{\mathrm{LQR}}(\tilde{x}_{0,i}) \coloneqq \frac{1}{2}\sum_{k=0}^{N} q_i\tilde{x}_{k,i}^2 + r_i\tilde{u}_{k,i}^2, \tag{6.40}$$

for $N \to \infty$ with $q_i \in \mathbb{R}_{++}$ and $r_i \in \mathbb{R}_{++}$ being the mode-by-mode state and input weighting gains, respectively. The LQR gain $k_{c,i}$ is computed as [130, Eq. 9.12]

$$k_{c,i} := \frac{\pi_g(1-\pi_g)p_i}{r_i + (1-\pi_g)^2 p_i}, \tag{6.41}$$

where $p_i \in \mathbb{R}_{++}$ is obtained from the DARE [130, Eq. 9.13]:

$$p_i = \pi_g^2 p_i - \frac{\pi_g^2(1-\pi_g)^2 p_i^2}{r_i + (1-\pi_g)^2 p_i} + q_i. \tag{6.42}$$

The solutions of (6.42) Equation (6.42) is a quadratic in p_i, which can be solved for the (unique positive) solution:

$$p_i := \frac{1}{2}\left(q_i - \frac{1-\pi_g^2}{(1-\pi_g)^2} r_i + \sqrt{\left(q_i - \frac{1-\pi_g^2}{(1-\pi_g)^2} r_i \right)^2 + \frac{4}{(1-\pi_g)^2} q_i r_i} \right). \tag{6.43}$$

The solutions of (6.43) characterise the infinite-horizon LQR cost as $\mathbf{J}_{\mathrm{LQR}}(\tilde{x}_{0,i}) = p_i(\tilde{x}_{0,i})^2$ and will be mapped back to original space to obtain the terminal cost matrix of the MPC formulation (6.9) as $P := V \operatorname{diag}(p_1, \ldots, p_{n_y}) V^{\mathrm{T}}$. From (6.43), it can be seen that for $q_i/r_i \to \infty$, $p_i \to q_i$ and $k_{c,i} = \pi_g/(1-\pi_g)$, which – substituted in (6.36) – results in a deadbeat controller. For positive feedback, i.e. $k_{c,i} > 0$, the regulator poles are therefore always *faster* than the pole of the plant π_g. However, the plant is subject to a large time delay of $n_\tau = 9$ steps and, to avoid large amplification of measurement noise, the closed-loop bandwidth must not be higher than $1/(n_\tau T_s) = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$, which is lower than the plant pole at 700 Hz. With the present structure, the control system must therefore be slowed down through the observer, so that – in the absence of input constraints – the choice of state and input weights becomes secondary. However, in the presence of input constraints and the ill-conditioned ORM R, the choice of input and state weights is particularly important and will be analysed in Subsection 6.5. To this end, the setpoint-related transformation from (6.36) is inverted to obtain the state-feedback law with setpoints as

$$\begin{aligned} \tilde{u}_{k,i} &:= \tilde{u}_{\mathrm{sp},i} - \begin{bmatrix} k_{c,i} & 0 & \ldots & 0 \end{bmatrix} (\tilde{\tilde{x}}_{k,i} - \tilde{\tilde{x}}_{\mathrm{sp},i}), \\ &= -\tilde{K}_{c,i} \begin{pmatrix} \tilde{\tilde{x}}_{k,i} \\ \tilde{d}_{0,i} \end{pmatrix}, \end{aligned} \tag{6.44}$$

where

$$\tilde{K}_{c,i} := \begin{bmatrix} k_{c,i} & 0 & \ldots & 0 & \frac{1+k_{c,i}}{\sigma_i} \end{bmatrix}. \tag{6.45}$$

6.4.3 Observer

In addition to estimating the state $\tilde{\bar{x}}_{k,i}$ and the disturbance $\tilde{d}_{0,i}$ required by the state feedback (6.44), the observer also controls the rate at which the setpoints (6.35) change. In this section, the dominant pole of the observer is tuned to match the dominant pole of the existing controller (6.30). Together, the LQR and the observer are referred to as a *linear quadratic Gaussian* (LQG) controller.

The dynamics of the filter-form observer [14], i.e. one that uses the measurements at time kT_s to form the a-posteriori estimate $\tilde{\bar{x}}_{k|k,i}$, are given by

$$\begin{pmatrix}\tilde{\bar{x}}_{k+1|k,i}\\ \tilde{d}_{k+1|k,i}\end{pmatrix} = A_i\begin{pmatrix}\tilde{\bar{x}}_{k|k,i}\\ \tilde{d}_{k|k,i}\end{pmatrix} + B_i\tilde{u}_{k,i}, \tag{6.46a}$$

$$\begin{pmatrix}\tilde{\bar{x}}_{k|k,i}\\ \tilde{d}_{k|k,i}\end{pmatrix} = \begin{pmatrix}\tilde{\bar{x}}_{k|k-1,i}\\ \tilde{d}_{k|k-1,i}\end{pmatrix} + \tilde{K}_{f,i}\left(\tilde{y}_{k,i} - C_i\begin{pmatrix}\tilde{\bar{x}}_{k|k-1,i}\\ \tilde{d}_{k|k-1,i}\end{pmatrix}\right), \tag{6.46b}$$

for $i = 1, \ldots, n$ and where the state model (6.31) and the disturbance model (6.33) are concatenated as

$$A_i \coloneqq \begin{bmatrix}A_{\bar{x},i} & 0\\ 0 & 1\end{bmatrix}, \qquad B_i \coloneqq \begin{bmatrix}B_{\bar{x},i}\\ 0\end{bmatrix}, \qquad C_i \coloneqq \begin{bmatrix}C_{\bar{x},i} & 1\end{bmatrix}. \tag{6.47}$$

Because the state model (6.31a) assumes zero noise for $\bar{x}_k$, the observer gain $\tilde{K}_{f,i} \in \mathbb{R}^{(n_\tau+2)\times 1}$ is of the form

$$\tilde{K}_{f,i} \coloneqq \begin{bmatrix}0 & \ldots & 0 & k_{f,i}\end{bmatrix}^{\mathrm{T}}, \tag{6.48}$$

with $n_\tau + 1$ leading zeros and a scalar gain $0 < k_{f,i} < 1$ associated with the disturbance (6.33).

With the observer (6.46), the LQR control law (6.44) becomes

$$\tilde{u}_{k,i} = -\tilde{K}_{c,i}\begin{pmatrix}\tilde{\bar{x}}_{k|k,i}\\ \tilde{d}_{k|k,i}\end{pmatrix}, \tag{6.49}$$

where $\tilde{K}_{c,i}$ is the LQR gain (6.45). Substituting (6.49) and (6.46b) in (6.46a) yields

$$\begin{pmatrix}\tilde{\bar{x}}_{k+1|k,i}\\ \tilde{d}_{k+1|k,i}\end{pmatrix} = \big(A_i - B_i\tilde{K}_{c,i}\big)\big(I - \tilde{K}_{f,i}C_i\big)\begin{pmatrix}\tilde{\bar{x}}_{k|k-1,i}\\ \tilde{d}_{k|k-1,i}\end{pmatrix} + \big(A_i - B_i\tilde{K}_{c,i}\big)\tilde{K}_{f,i}\tilde{y}_{k,i}, \tag{6.50}$$

from which the mode-by-mode LQG controller, $c_{LQG,i} : \mathbb{C} \mapsto \mathbb{C}$, is obtained as (Appendix 6.A)

$$c_{LQG,i}(z) \coloneqq \frac{(1-\pi_{f,i})(1-\pi_{c,i})(z-\pi_g)z^{n_\tau}/(\sigma_i(1-\pi_g))}{z^{n_\tau+1} - z^{n_\tau}(\pi_{f,i}+\pi_{c,i}) + z^{n_\tau-1}\pi_{f,i}\pi_{c,i} - (1-\pi_{f,i})(1-\pi_{c,i})}, \tag{6.51}$$

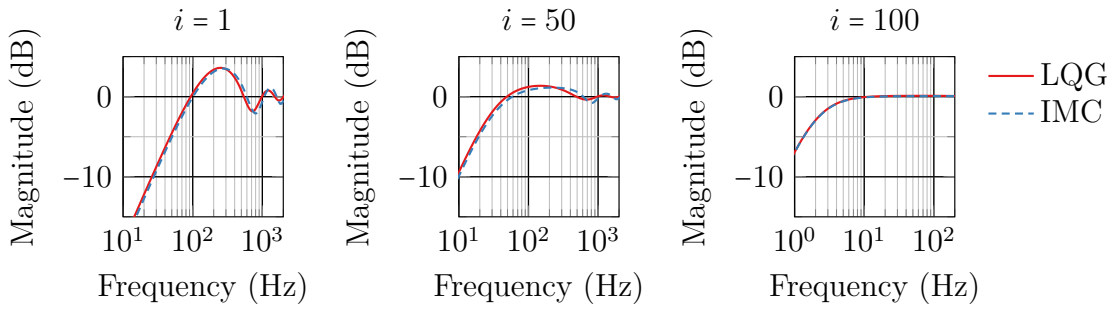


Figure 6.6: Comparison of LQG ($q_i = r_i = 1$) and IMC output sensitivities for modes 1, 50 and 100.

where the $\pi_{c,i}$ and $\pi_{f,i}$ refer to the regulator and observer poles, respectively, and are given by

$$\pi_{c,i} := \pi_g - (1 - \pi_g)k_{c,i}, \tag{6.52a}$$

$$\pi_{f,i} := 1 - k_{f,i}. \tag{6.52b}$$

With (6.51), the mode-by-mode output sensitivity $S_{LQG,i} : \mathbb{C} \mapsto \mathbb{C}$ becomes

$$\begin{aligned} S_{LQG,i}(z) &:= \frac{1}{1 + c_{LQG,i}(z)\sigma_i g(z)}, \\ &= 1 - \frac{(1 - \pi_{f,i})(1 - \pi_{c,i})}{z^{n_\tau - 1}(z - \pi_{f,i})(z - \pi_{c,i})}. \end{aligned} \tag{6.53}$$

Given a dominant pole π_i^+ from the existing controller (6.30), setting $\pi_{f,i} = \pi_i^+$ in (6.52b) results in the following observer gain:

$$k_{f,i} = 1 - \pi_i^+. \tag{6.54}$$

The LQG controller implements one additional pole $\pi_{c,i}$ (6.52a) that is over 3 times faster than π_i^+ for $k_{c,i} > 0$. Fig. 6.6 compares the mode-by-mode LQG output sensitivity (6.53) with the IMC output sensitivity (6.29) for modes $i \in \{1, 50, 100\}$ and $r_i = q_i = 1$. For all modes, the choice (6.54) leads to a good match between the LQG and IMC sensitivity. For lower-order modes that have a high bandwidth, the additional LQG poles lead to bandwidth reduction of approximately 5 Hz, which could be accounted for by increasing the observer bandwidth.

Suppose one were to obtain the observer gain (6.48) with (6.54) through a Kalman filter formulation. For the steady-state Kalman filter, the gain is computed as [128, Ch. 7.3]

$$\tilde{K}_{f,i} = P_{f,i}C_i^{\mathrm{T}}\left(C_iP_{f,i}C_i^{\mathrm{T}} + r_{f,i}\right)^{-1}C_iP_{f,i}A_i^{\mathrm{T}} + Q_{f,i}, \tag{6.55}$$

where $r_{f,i}$ refers to the measurement noise variance, $Q_{f,i}$ to the state noise variance and $P_{f,i} = P_{f,i}^{\mathrm{T}} \succeq 0$ is the solution to the DARE

$$P_{f,i} = A_iP_{f,i}A_i^{\mathrm{T}} - A_iP_{f,i}C_i^{\mathrm{T}}\left(C_iP_{f,i}C_i^{\mathrm{T}} + r_{f,i}\right)^{-1}C_iP_{f,i}A_i^{\mathrm{T}} + Q_{f,i}. \tag{6.56}$$

By substituting (6.55) in (6.56) and setting $Q_{f,i} \coloneqq \mathrm{diag}(0,\ldots,0,q_{f,i})$, it can be seen that $P_{f,i} \coloneqq \mathrm{diag}(0,\ldots,0,p_{f,i})$ solves the DARE with

$$p_{f,i} = \frac{q_{f,i}}{k_{f,i}}, \qquad r_{f,i} = \frac{1-k_{f,i}}{k_{f,i}}q_{f,i}, \tag{6.57}$$

where $k_{f,i}$ is the observer gain from (6.54) that satisfies $0 < k_{f,i} < 1$. The weights

$$q_{f,i} = 1, \qquad r_{f,i} = \frac{1-k_{f,i}}{k_{f,i}^2} = \frac{\pi_i^+}{(1-\pi_i^+)^2} \tag{6.58}$$

therefore produce the same gain as in (6.54). Comparing (6.58) with the root locus (Fig. 6.5a), it can be seen that for higher-order modes, i.e. modes for which μ/σ_i^2 is large and therefore $\pi_i^+ \approx 1$, the Kalman filter parameter $r_{f,i}$ is large. Tuning the Kalman filter to match the dynamics of the regularised IMC controller therefore requires large measurement noise for higher-order modes to be assumed.

In original space, the MIMO equivalent of the mode-by-mode observer (6.46) is given by

$$\begin{pmatrix}\bar{x}_{k+1|k}\\ d_{k+1|k}\end{pmatrix} = A\begin{pmatrix}\bar{x}_{k|k}\\ d_{k|k}\end{pmatrix} + Bu_k, \tag{6.59a}$$

$$\begin{pmatrix}\bar{x}_{k|k}\\ d_{k|k}\end{pmatrix} = \begin{pmatrix}\bar{x}_{k|k-1}\\ d_{k|k-1}\end{pmatrix} + K_f\left(y_k - C\begin{pmatrix}\bar{x}_{k|k-1}\\ d_{k|k-1}\end{pmatrix}\right), \tag{6.59b}$$

where $A \in \mathbb{R}^{(\bar{n}_x+n_y)\times(\bar{n}_x+n_y)}$, $B \in \mathbb{R}^{(\bar{n}_x+n_y)\times n_u}$ and $C \in \mathbb{R}^{n_y\times(\bar{n}_x+n_y)}$ are defined as

$$A \coloneqq \begin{bmatrix}A_{\bar{x}} & \\ & I\end{bmatrix}, \qquad B \coloneqq \begin{bmatrix}B_{\bar{x}}\\ 0\end{bmatrix}, \qquad C = \begin{bmatrix}C_{\bar{x}} & I\end{bmatrix}, \tag{6.60}$$

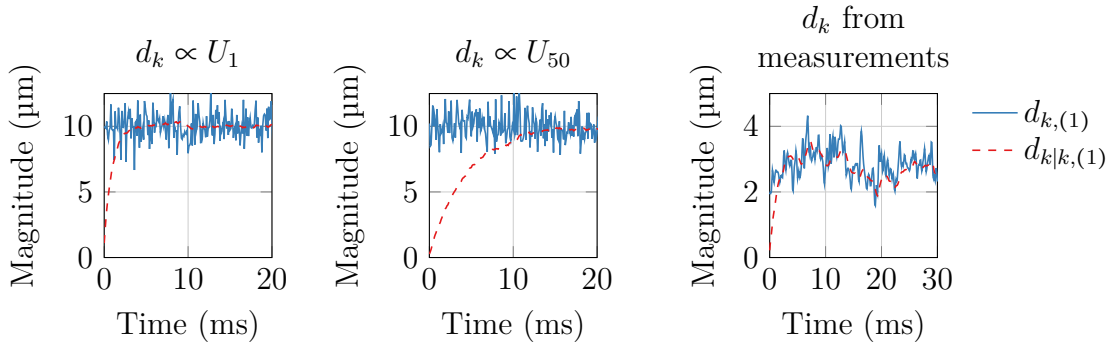


Figure 6.7: First component of the disturbance d_k and observer estimate $d_{k|k}$ for a disturbance in direction of mode 1 (left) and mode 50 (middle) and for d_k from measurements (right).

with $A_{\bar{x}}$, $B_{\bar{x}}$ and $C_{\bar{x}}$ given in (6.32). The observer gain $K_f \in \mathbb{R}^{(\bar{n}_x+n_y)\times n_y}$ is obtained from the mode-by-mode gains (6.54) and given by

$$K_f := \begin{bmatrix} 0 \\ \vdots \\ V \operatorname{diag}(k_{f,1}, \ldots, k_{f,n_y}) U^{\mathrm{T}} \end{bmatrix}. \tag{6.61}$$

In combination with the condensed MPC formulation (6.13), the output (6.59b) of the MIMO observer is used to set the initial condition $x(t) := x_{k|k}$ and the setpoints $x_{\mathrm{sp}} = u_{\mathrm{sp}} := -R^{-1} d_{k|k}$, which are in turn used to initialise the Hessian $\mathbf{J}$ and the objective function vector $\mathbf{q}$.

Fig. 6.7 shows the first component of the observer estimate $d_{k|k}$ (- - -) for different disturbances d_k (——). In the first and second column the disturbance is chosen as $d_k = 10\,\mu\mathrm{m} \times U_i/|U_{(1,i)}| + n_k$, where U_i represents column i of the matrix of left singular vectors of R (1.6) and $n_k \sim \mathcal{N}(0, I)$. According to the dominant poles from Fig. 6.5b, the first mode is 4 times faster than mode 50, which is reflected in Fig. 6.7. The last column of Fig. 6.7 shows the observer output for the case that d_k is taken from measurement data.

6.5 Tuning Model Predictive Control

Motivated by the ill-conditioned plant and the large time delay, an LQG controller was designed to match the dynamics of the existing Diamond closed loop in

Section 6.4. It was shown that the LQG controller must be slowed down through the observer, which left the LQR state and input weights that are used in the MPC formulation (6.9) unspecified. The state and input weights q_i and r_i are determined on a mode-by-mode basis and the terminal cost p_i is computed from (6.43). In original space, the matrices are defined as $Q \coloneqq V\,\mathrm{diag}(q_1,\ldots,q_{n_y})V^{\mathrm{T}}$, $R_u \coloneqq V\,\mathrm{diag}(r_1,\ldots,r_{n_y})V^{\mathrm{T}}$ and $P \coloneqq V\,\mathrm{diag}(p_1,\ldots,p_{n_y})V^{\mathrm{T}}$. In this section, the horizon of the MPC scheme (6.9) is fixed to $N = 1$ and it is assumed that the MPC problem is solved using Algorithm 6.1 (FGM). As highlighted in Section 6.2.1, the convergence of Algorithm 6.1 tends to be slower for large condition numbers of the Hessian $\mathbf{J}$ (6.14a), which for $N = 1$ is given by

$$\begin{aligned}\mathbf{J} &= B_x^{\mathrm{T}} P B_x + R_u,\\ &= V\,\mathrm{diag}\big((1-\pi_g)^2 p_1 + r_1,\ldots,(1-\pi_g)^2 p_{n_y} + r_{n_y}\big)V^{\mathrm{T}}.\end{aligned} \tag{6.62}$$

For fixed π_g, the condition number $\kappa(\mathbf{J})$ therefore depends on r_i as well as on p_i, which also depends on q_i through (6.43). In the following, the state weights are fixed as $q_i = 1\ \forall i = 1,\ldots,n_y$ and different input weight choices are compared:

$$r_i = 1, \tag{6.63a}$$

$$r_i = (\mu + \sigma_i^2)/\sigma_i^2, \tag{6.63b}$$

$$r_i = \sqrt{\sigma_i}, \tag{6.63c}$$

$$r_i = \sigma_i, \tag{6.63d}$$

from which the mode-by-mode terminal cost p_i is computed using (6.43). The choice $r_i = (\mu + \sigma_i^2)/\sigma_i^2$ is motivated by the regularisation procedure from Section 1.6.3; for higher-order modes, i.e. modes associated with $\sigma_i \ll 1$, the weight $r_{1,i}$ is large compared to the state weights and the control action therefore reduced. The choices $r_i = \sqrt{\sigma_i}$ and σ_i are motivated in Section 6.5.1.

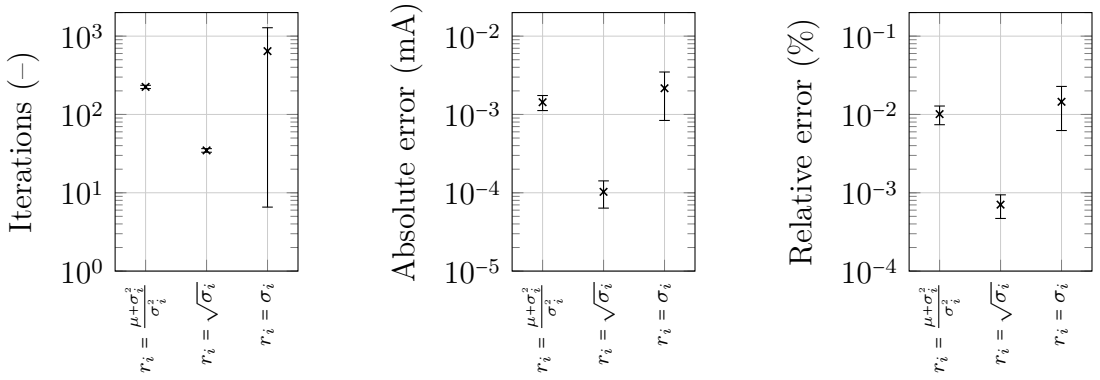
The resulting condition numbers $\kappa(P)$ and $\kappa(\mathbf{J})$ are shown in Table 6.1. For $r_i = 1$, the terminal cost is identical for all modes, so that $\kappa(\mathbf{J}) = 1$ and Algorithm 6.1 terminates in one step. For $r_i \neq 1$, the practical convergence of Algorithm 6.1 is analysed in Fig. 6.8 by measuring the number of iterations requires to solve the

Table 6.1: Comparison of the condition numbers of the DARE solution P and the Hessian J of Algorithm 6.1 for different choices of input weights r_i. The values are computed for the vertical control direction.

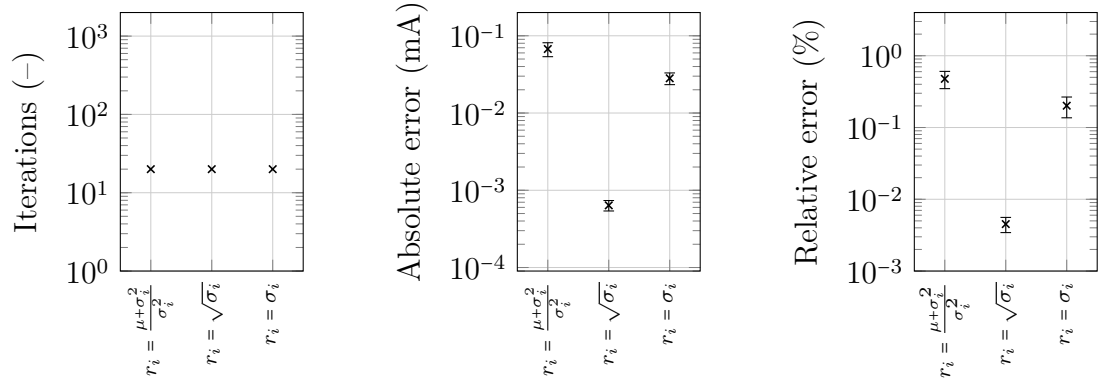
q_i	r_i	$\kappa(R_u)$	$\kappa(P)$	$\kappa(J)$
1	1	1	1	1
1	$(\mu+\sigma_i^2)/\sigma_i^2$	2538.8	1.1	2126.4
1	$\sqrt{\sigma_i}$	102.6	1.36	49.2
1	σ_i	10 531.0	1.6	1361.9

MPC problem (6.13). For each instance, the algorithm is warm-started using the previously calculated input $u^\star_{k-1}$ and in addition to the maximum iteration number $I_{\max}$, the stopping criterion $(\|w_{i+1}-w_i\|_\infty < \epsilon) \vee (\|w_{i+1}-w_i\|_\infty < \epsilon\|w_i\|_\infty)$ with $\epsilon = 10^{-3}$ is implemented. As expected from $\kappa(\mathbf{J})$ in Table 6.1, Algorithm 6.1 requires more iterations with weights $r_i = (\mu+\sigma_i^2)/\sigma_i^2$ and $r_i = \sigma_i$ than with $r_i = \sqrt{\sigma_i}$ when $I_{\max} = 4000$ (Fig. 6.8a). The number of required iterations is particularly volatile for $r_i = \sigma_i$. The second and third column of Fig. 6.8 show the average absolute error (mA) and relative error (%) that are computed as $\|u^\star_{\text{FGM}} - u^\star_{\text{IP}}\|_\infty$ and $\|u^\star_{\text{FGM}} - u^\star_{\text{IP}}\|_\infty / \|u^\star_{\text{IP}}\|_\infty \times 100\,\%$, where $u^\star_{\text{FGM}}$ is the solution obtained from Algorithm 6.1 and $u^\star_{\text{IP}}$ the solution obtained from an interior-point method. The solutions are one magnitude order less accurate for $r_i = (\mu+\sigma_i^2)/\sigma_i^2$ and $r_i = \sigma_i$ than for $r_i = \sqrt{\sigma_i}$.

In view of the implementation (Section 6.6), $I_{\max}$ is reduced to $I_{\max} = 20$ in Fig. 6.8b making the stopping criterion $(\|w_{i+1}-w_i\|_\infty < \epsilon) \vee (\|w_{i+1}-w_i\|_\infty < \epsilon\|w_i\|_\infty)$ redundant. Compared to Fig. 6.8a, all errors have been increased by one order of magnitude, but remain below 0.1 mA and 1 %. The resulting ASD is shown in Fig. 6.9 for BPM 1 (vertical direction). The controller performs worst for $r_i = (\mu+\sigma_i^2)/\sigma_i^2$ and no difference is visible between $r_i = \sigma_i$ and $r_i = \sqrt{\sigma_i}$, from which it can be concluded that the solution inaccuracy of the version with $r_i = \sigma_i$ is associated with direction aligned with higher-order modes; since the contribution of higher-order modes to the disturbance spectrum is smaller than low-order modes, the version with $r_i = \sigma_i$ produces an equally good performance than the version with $r_i = \sqrt{\sigma_i}$ that has better convergence properties.



(a) $I_{\max} = 4000$.



(b) $I_{\max} = 20$.

Figure 6.8: Convergence of Algorithm 6.1 for $N = 1$ and input weights (6.63b)–(6.63d) for 10,000 warm-started MPC instances (6.13). In addition to $I_{\max}$, Algorithm 6.1 implements the stopping criterion $(\|w_{i+1} - w_i\|_\infty < \epsilon) \vee (\|w_{i+1} - w_i\|_\infty < \epsilon\|w_i\|_\infty)$ with $\epsilon = 10^{-3}$. The figures show average and standard deviations and the error is computed using the ∞-norm and against a solution obtained from an interior-point method.

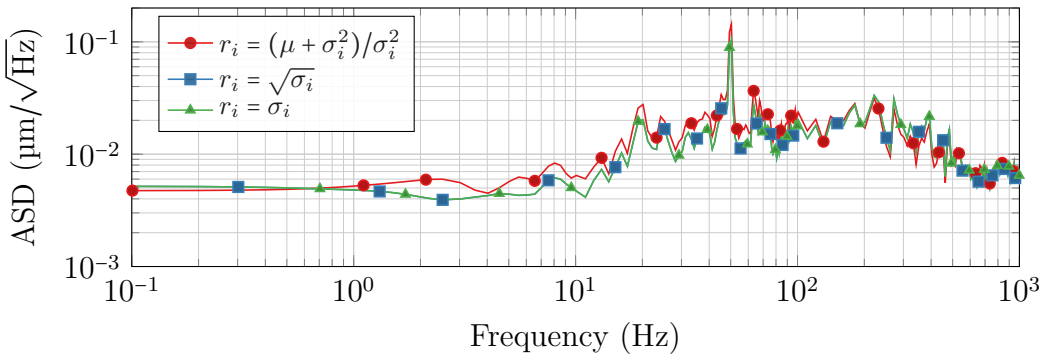


Figure 6.9: Simulated ASD (BPM 1) for $N = 1$, input weights (6.63b)–(6.63d) and Algorithm 6.1 with $I_{\max} = 20$.

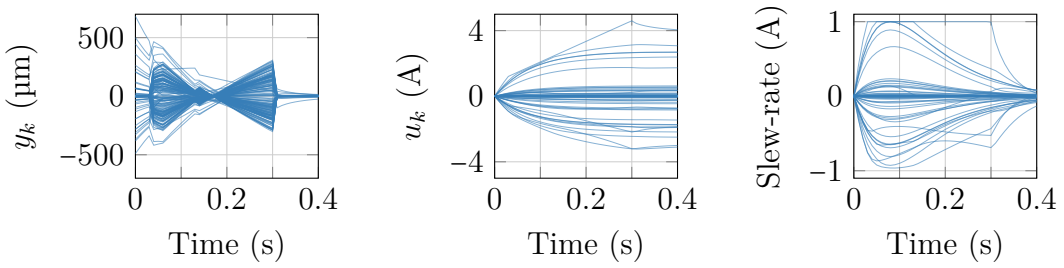


Figure 6.10: Simulation of LQG under input clipping and with $q_i = r_i = 1$.

6.5.1 Saturation of Slew-Rate Constraints

In addition to the solver, the choice of input weights (6.63) impacts the behaviour of MPC under actuator saturation. To create a situation in which the actuators saturate under slew-rate constraints, the disturbance d_k is chosen in the direction of mode 100, as $d_k := U_{100} \times 10\,\text{mm}\;\forall k \geq 0$, and the MPC algorithm simulated in closed loop for the different input weights (6.63).

The resulting output is shown in Fig. 6.11 and compared with the LQG controller ($q_i = r_i = 1$) under input clipping in Fig. 6.10, i.e. when the constraints are enforced *after* calculating the inputs. The first column of Fig. 6.10 and 6.11 shows all n_y outputs y_k, the second column all $n_u = n_y$ inputs u_k and the third column the slew-rate, which in practice is calculated as the difference between u_k and a lowpass-filtered u_k (Chapter 7). Actuator saturation occurs when the slew-rate reaches ±1 A or when the inputs reach ±5 A. In Fig. 6.10 and 6.11, the actuators start saturating under slew-rate constraints at 0.05 s. As consequence, the LQG controller produces the unusual output shown in the first column of Fig. 6.10, whereas the MPC output diverges for $r_i = 1$ and $r_i = (\mu + \sigma_i^2)/\sigma_i^2$ with some outputs reaching over 2 mm beam displacement, before converging again after 0.15 s once the actuators start desaturating. Note that throughout the simulation, the input computed by the MPC algorithm are within the constraints (6.21). For $r_i = \sqrt{\sigma_i}$, the beam divergence seen for $r_i = 1$ and $r_i = (\mu + \sigma_i^2)/\sigma_i^2$ is reduced and further improved with $r_i = \sigma_i$. The behaviour could be further improved by using $r_i = \alpha\sigma_i$, $\alpha > 1$, or $r_i = \sigma_i^2$ at the expense of increasing the condition number $\kappa(\mathbf{J})$.

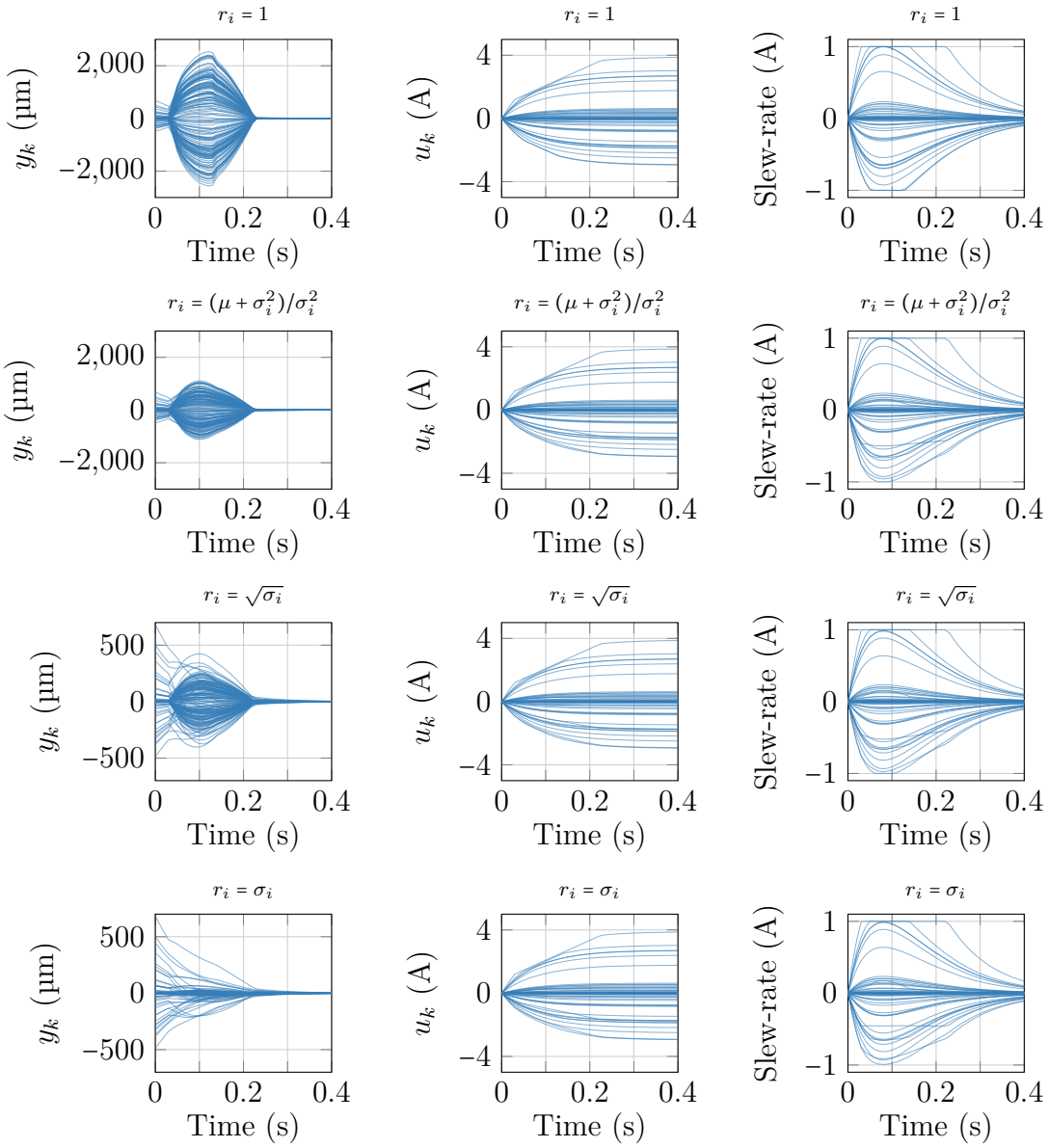


Figure 6.11: Simulation of MPC under input slew-rate saturation with state weights $q_i = 1$ and different input weights r_i (vertical control direction).

To understand the behaviour from Fig. 6.10 and 6.11, the simulation results are mapped to modal space using the modal transformation (1.8) and shown in Fig. 6.12, where the first column shows the LQG controller ($q_i = r_i = 1$) and the second and third column MPC with $r_i = 1$ and $r_i = \sigma_i$, respectively. At the beginning of the simulation, it can be seen that only one output and one input are non-zero, which are those associated with mode 100. As soon as actuator saturation occurs, the clipping (in case of LQG) or the projection onto the constraint set (6.21) (in case of MPC)

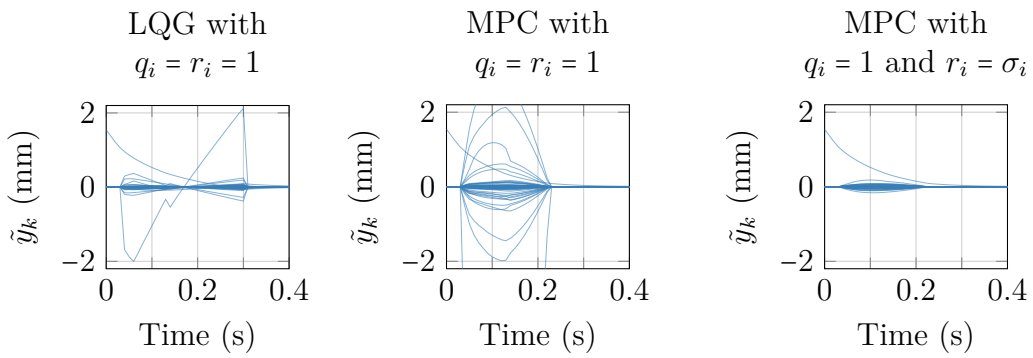


Figure 6.12: Simulation of LQG and MPC under input slew-rate saturation in modal space for the vertical control direction.

creates an input component in direction of other modes. As a consequence, the output diverges resulting in the output from Fig. 6.11. Under actuator saturation, the ill-conditioned dynamics therefore require MPC to concentrate the control action on higher-order modes to avoid producing large displacements in direction of the lower-order modes.

6.6 Implementation

For the experiments on the Diamond storage ring, the controller is implemented on a VadaTech AMC540 board that embeds an FPGA and two Texas Instruments (TI) C6678 DSPs [145] with 8 cores each (Chapter 7). For the tests, Algorithms 6.1 ($N = 1$) and 6.4 ($N = 2$) are implemented on the DSPs, which are more flexible to program, while the FPGA is responsible for signal routing. The data transfer between FPGA and DSPs takes roughly 5 µs. The DSPs are clocked at 1.4 GHz and the sampling frequency of 10 kHz therefore allows for 133,000 processor cycles (95 µs) for computing the control inputs. The control problems for the vertical and horizontal beam directions are independent and one DSP is used for each direction.

For the gradient step of Algorithms 6.1 and 6.4, an optimised matrix-vector multiplication routine is implemented that exploits the core architecture and maximises the cache efficiency, which is elaborated on in detail in Chapter 7. For $N = 1$, the algorithm can be implemented as shown in Algorithm 6.1 and all the problem data, such as the Hessian $\mathbf{J}$, can be saved in L2 memory. The cache efficiency

for the projection can be increased by permuting the data using a perfect shuffle, so that the inputs for magnet i and horizon stages 0 and 1 are contiguous in memory.

If the algorithm is run on a single core with $I_{\max} = 20$, it requires 543 µs for $N = 1$ and 3550 µs for $N = 2$ to compute the control inputs, which is far more than the desired 100 µs. The most expensive operation is the gradient step, which takes 357 µs for $N = 1$ and 3017 µs for $N = 2$. Since the algorithm is dominated by the matrix-vector multiplication of the gradient step, one would expect the computation time to quadruple when doubling the problem size. However, the transfer of problem data across memory level and cache inefficiencies incur substantial overheads.

The single-core implementation is then parallelised using the standard manager-worker framework from Section 7.2.2, but variable dependencies require core communication and cache operations that are denoted by circled arrows in Algorithm 6.1. For the problem size of the MPC problem (6.13), the cost of parallelisation is not negligible and analysed in detail in Section 7.2.2. Algorithm 6.1 is sliced into 6×32 row-blocks with 192 columns each and deployed on 6 worker cores and 1 manager core. The length of the slices must be a multiple of the cache line size (64 B) and using 7 worker cores would not yield any speed up. The master core coordinates the various steps of Algorithm 6.1 and communicates with the adjacent FPGA. A breakdown of the computation time of Algorithm 6.1 with $I_{\max} = 20$ is shown in Fig. 6.13. For $N = 1$, the algorithm uses 69 µs, which is well below the allowed 100 µs, but for $N = 2$, the computation time of 272 µs is above the time limit. The algorithm would therefore have to be implemented on the FPGA for $N = 2$. However, the very small performance improvement that is obtained when increasing N from 1 to 2 [89] may not justify a complex FPGA implementation.

Compared to the single-core implementation, the parallelisation reduces the computation time by a factor between about 8 and 13. In theory, one would expect the computation time to be reduced by a factor smaller than n_w when deployed onto n_w worker cores. It is suspected that this discrepancy is due to memory and cache bandwidth limitations on the single core implementation.

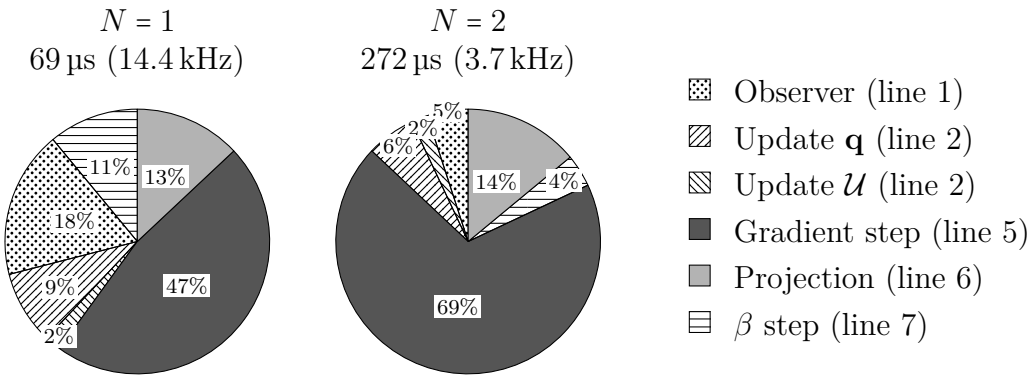


Figure 6.13: Computation times for multi-core implementation.

6.7 Results from the Diamond Storage Ring

In order to validate the MPC design and implementation, the algorithm was tested experimentally on the Diamond storage ring and compared against the single-array IMC algorithm from Section 6.4.1. Two versions of the MPC algorithm are presented: One with input weights $r_i = (\mu + \sigma_i^2)/\sigma_i^2$ (6.63b) and one with $r_i = \sigma_i$ (6.63d). Additional results for $r_i = \sqrt{\sigma_i}$ (6.63c) are shown in Appendix 6.B. For all versions of MPC, the observer is tuned to match the IMC closed-loop dynamics using the technique from Section 6.4 using a nominal closed-loop bandwidth of $\lambda = 2\pi \times 176\,\mathrm{rad\,s^{-1}}$. In the horizontal direction, the IMC and MPC algorithms control $n_y = n_u = 167$ BPMs and correctors and in the vertical direction, $n_y = n_u = 165$. The following results were obtained using a nominal beam current of 300 mA and with all other feedbacks from Table 1.1 disabled.

Outputs

Fig. 6.14 shows the output ASD measured in the first cell of the Diamond storage ring for disabled feedback (——), for the IMC algorithm (-·-·-) and for the two MPC algorithms with $r_i = (\mu + \sigma_i^2)/\sigma_i$ (- - -) and $r_i = \sigma_i$ (·····). The left-hand side of Fig. 6.14 shows the horizontal direction, the right-hand side the vertical direction, and the first to seventh rows correspond to BPMs 1–7.

For all algorithms, the disturbance attenuation is in line with the theoretical expectation from Fig. 6.6 (see also Fig. 4.5). Maximum attenuations of 20 dB = 0.1

and 40 dB = 0.01 are expected at 10 Hz and 1 Hz, which is reflected in both planes and in particular for the BPMs 1 and 7. As the MPC closed-loop dynamics are dominated by the observer, the performance is identical for MPC with $r_i = (\mu+\sigma_i^2)/\sigma_i^2$ and $r_i = \sigma_i$. According to Section 6.5.1, an eventual performance difference would appear under actuator saturation, which was not subject of the tests conducted on the Diamond storage ring because of time restrictions.

In the horizontal direction and between 20 Hz to 50 Hz, the IMC algorithm performs slightly better than the MPC algorithm, which is better visible in the IBMs from Fig. 6.15. At BPM 4, for example, the IBM associated with the IMC algorithm is 0.1 µm lower at 1 kHz. The performance difference is less pronounced in the vertical direction and it is unclear by what it is caused, but it could be related to the solver convergence discussed in Section 6.5.

Inputs

Fig. 6.16 shows the input ASD in mA measured in the first cell of the Diamond storage ring for the IMC algorithm (-·-·-) and for the two MPC algorithms with $r_i = (\mu + \sigma_i^2)/\sigma_i$ (- - -) and $r_i = \sigma_i$ (·····). The left-hand side of Fig. 6.16 shows the horizontal direction, the right-hand side the vertical direction, and the first to seventh rows correspond to correctors 1–7.

As all algorithms are tuned to have the same closed-loop dynamics, the input ASD is similar for the IMC and the MPC algorithms. For lower frequencies, the only visible differences occur for corrector 2 in the horizontal direction and correctors 4–6 in the vertical direction, where the input ASD is 0.1 mA/$\sqrt{\text{Hz}}$ to 0.2 mA/$\sqrt{\text{Hz}}$ higher for the IMC than for the MPC algorithms. At frequencies higher than 500 Hz, the input ASD for the IMC algorithm is up to two times higher than for the MPC algorithms. It is unclear what causes this difference in input ASD, which is not reflected in the output ASD from Fig. 6.14.

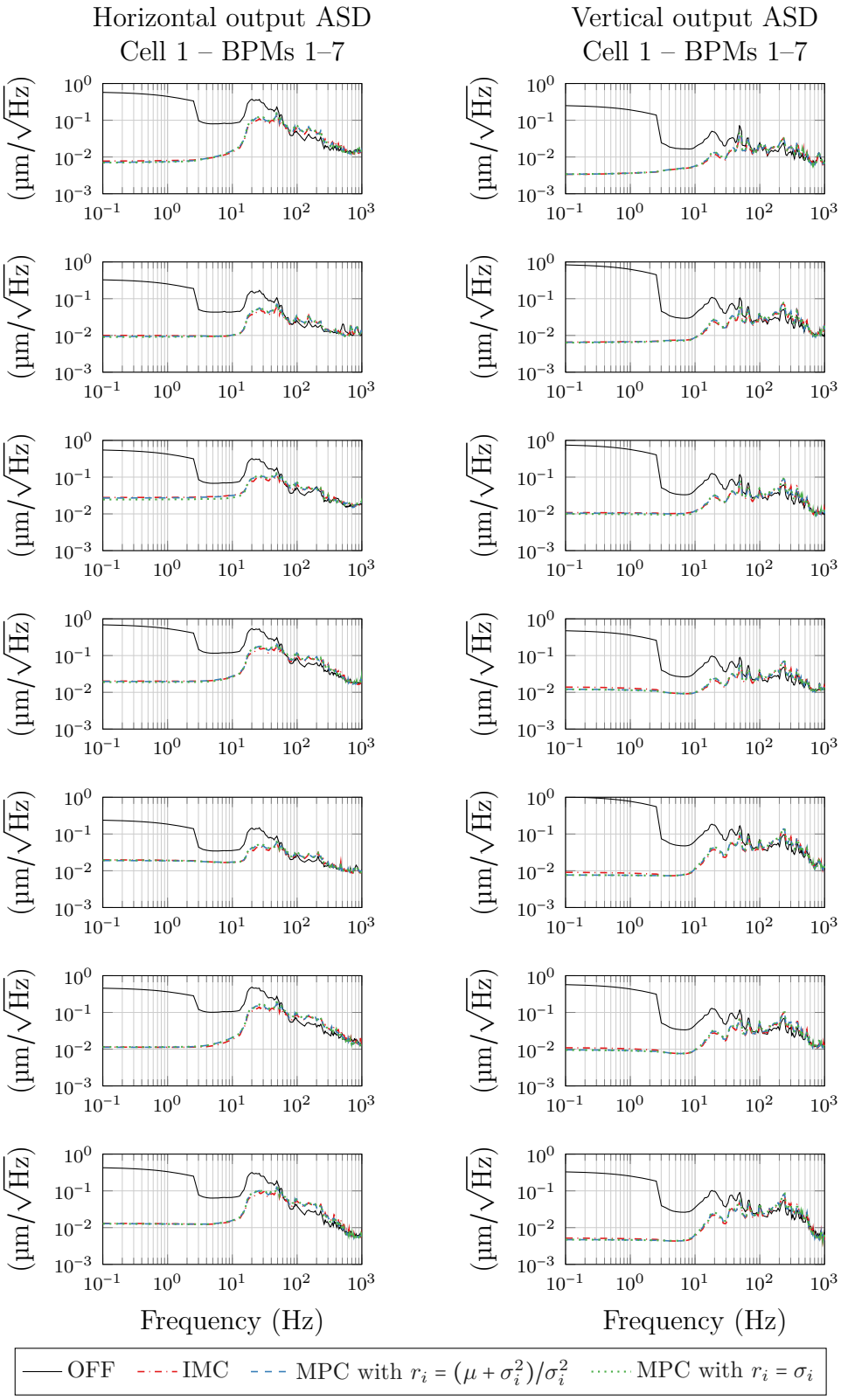


Figure 6.14: Measured output ASD in the first cell of the Diamond storage ring for disabled feedback (OFF), IMC from Section 6.4.1 and MPC. The ASDs are computed using Welch's method and using 100 s of data sampled at 10 kHz.

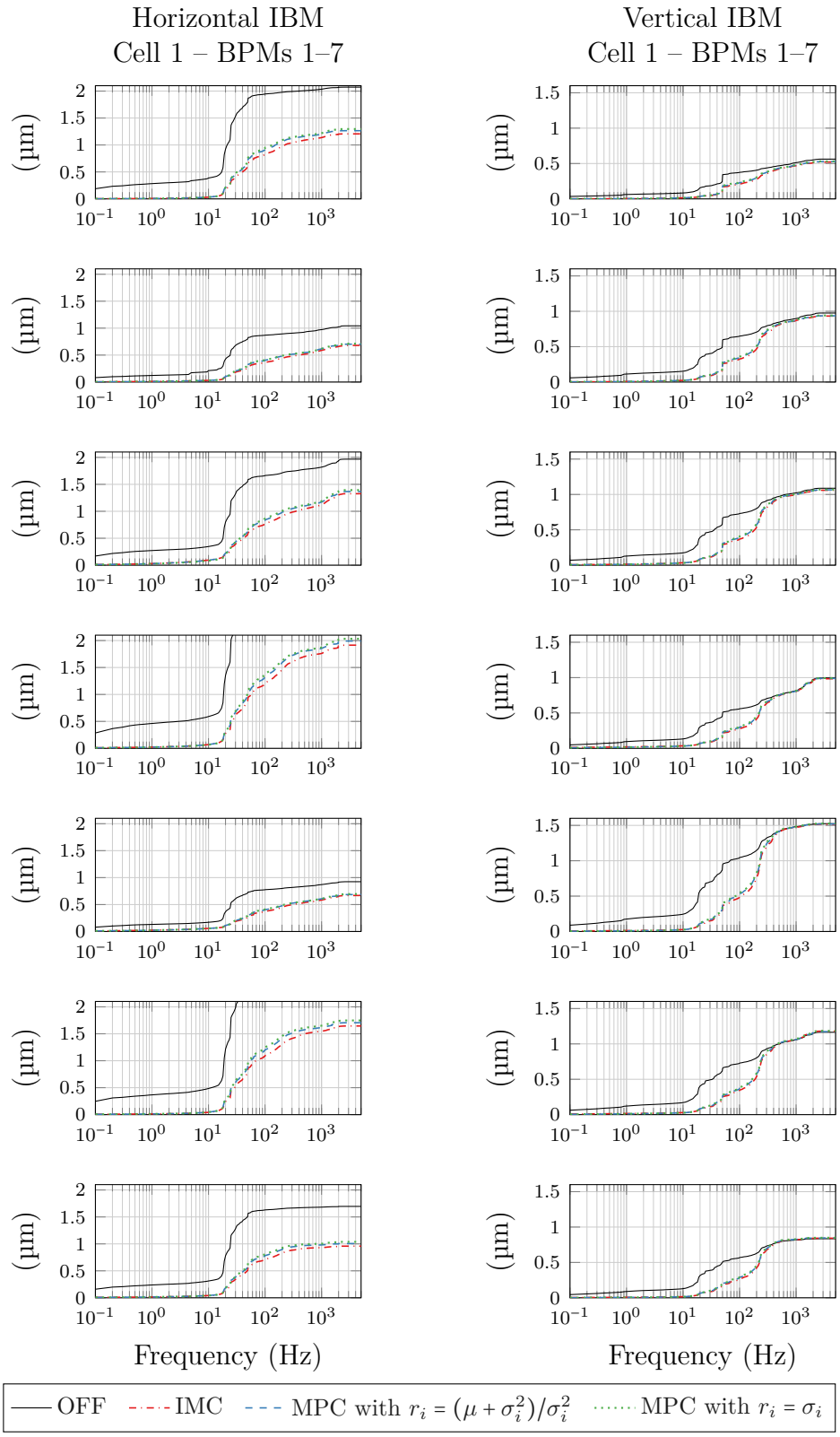


Figure 6.15: Measured IBM in the first cell of the Diamond storage ring for disabled feedback (OFF), IMC from Section 6.4.1 and MPC algorithms with $r_i = (\mu + \sigma_i^2)/\sigma_i$ and $r_i = \sigma_i$.

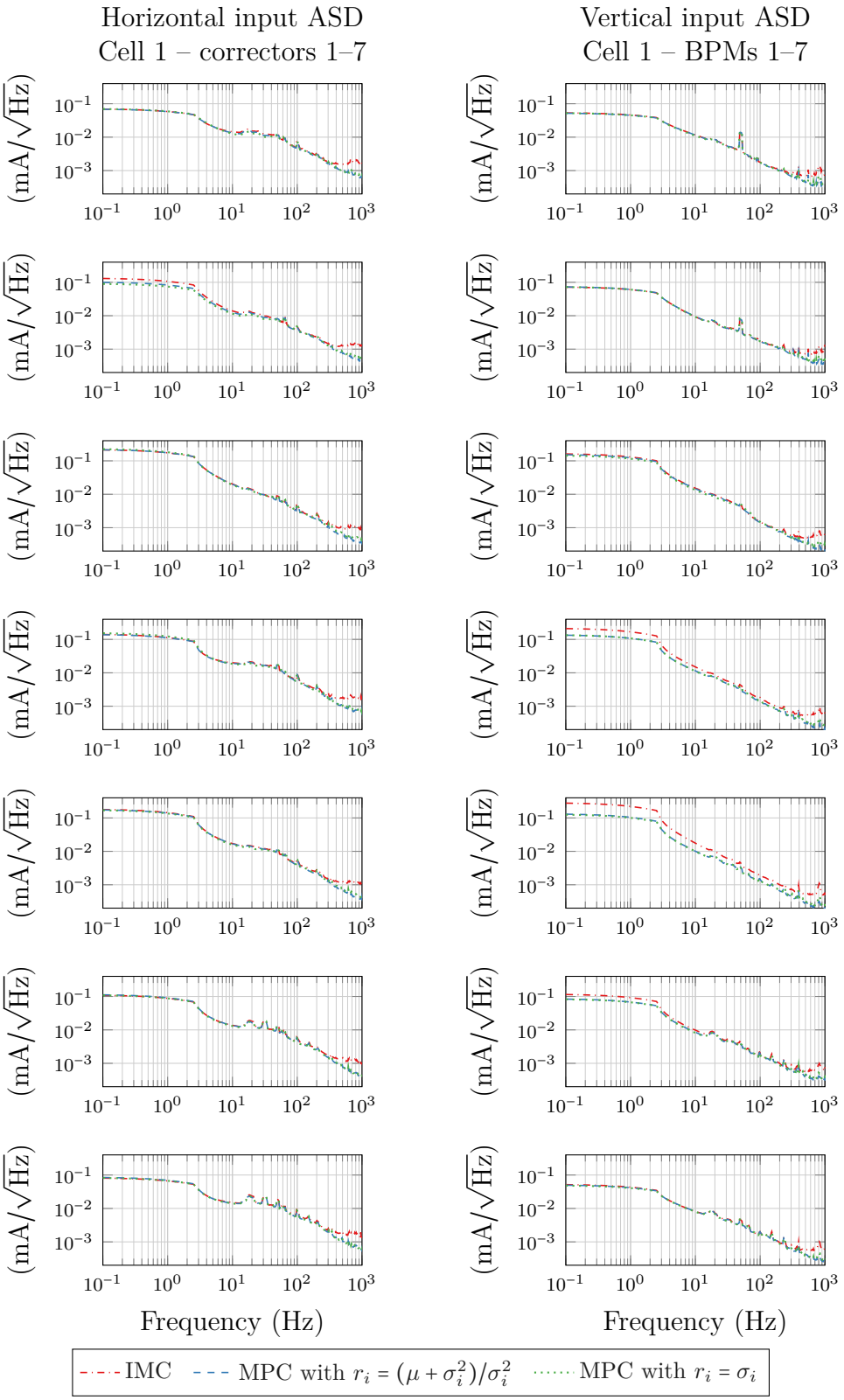


Figure 6.16: Measured ASD of the inputs in the first cell of the Diamond storage ring for the IMC and MPC algorithms with $r_i = (\mu + \sigma_i^2)/\sigma_i^2$ and $r_i = \sigma_i$.

6.8 Conclusion

In this chapter, an MPC algorithm has been designed for the CD dynamics of the electron beam. For solving the associated convex optimisation problem, an ADMM-based solver has been compared to an FGM implementation that is combined with Dykstra's projection algorithm for horizons $N \geq 2$. The FGM-Dykstra algorithm makes the augmentation of decision variables for the purpose of considering input constraints superfluous and hence reduces the computational complexity of the MPC problem. However, simulations have shown that stalling conditions of Dykstra's algorithm can occur, in which case Dykstra's algorithm can produce an arbitrarily large projection error when run for a fixed number of iterations. Even though convergence proofs of the FGM under inaccurate projections exist [114], the possibility of stalling conditions would lead to conservative convergence estimates of the FGM. To improve convergence estimates, the stalling conditions of Dykstra's method must be prevented, which is subject of current research [13]. In [150], Dykstra's method is related to a block coordinate descent method and to ADMM. Based on these relations, one could attempt at introducing varying step sizes or modifying Dykstra's algorithm otherwise to prevent stalling. Alternatively, Dykstra's method could be interpreted as a distributed optimisation problem, in which each separate projection is seen as an agents objective. Analogous to the distributed proximal minimization algorithm [99, Alg. 1], the projected iterates from Dykstra's method could be obtained from averaged iterates, but it is unclear whether Dykstra's method would still converge to the Eucledian projection.

For horizon $N = 1$ the projection onto the input constraint set is straightforward and Dykstra's method superfluous. In this case, the projections employed in the FGM (Algorithm 6.1) and ADMM (Algorithm 6.2) are identical. For speeding up ADMM, the solution of the linear system step could be obtained by inverting the KKT matrix offline and implementing a matrix-vector multiplication. In contrast to the matrix-factorisation based forward-backward substitution implemented in [132], a matrix-vector multiplication would benefit from an efficient parallelisation. As the numerical results from Section 6.3.4 suggest, the control system would benefit

from improved convergence properties of ADMM, which is known to be less susceptible to ill-conditioned Hessians compared to FGM [104]. For the control system implementation at Diamond, only the FGM has been implemented and tested for horizon $N = 1$ and future research could focus on implementing an ADMM-based solver.

To achieve offset-free control, the observer has been augmented with a disturbance model providing one integrator per output. The observer has been designed in modal space and tuned based on comparing the dominant closed-loop pole of an IMC algorithm with the dominant closed-loop pole of the resulting LQG controller. Alternatively, the LQG controller could be determined using "inverse LQR" techniques [14]. For determining the state and input weights of the MPC (and LQG) problem, from which the terminal cost matrix is computed using the associated DARE, several choices of input weights have been analysed with respect to FGM convergence, closed-loop performance and behaviour under actuator saturation. In contrast to the regularisation procedure for the standard IMC algorithm which reduces controller gains in directions associated with higher-order modes, the input weights of the MPC implementation must be chosen small for higher-order modes and large for low-order modes. In simulations, it has been shown that this counter-intuitive choice of weights prevents beam divergence when the actuators saturate, which is related to the projection onto the input constraints that can produce a large beam displacement in lower-order modes.

The large beam displacements associated with actuator saturation could be prevented by including state constraints and a terminal constraint set, which were excluded to avoid complicating the optimisation problem. However, this could lead to infeasible optimisation programs, which would need to be treated separately in practice. Future research could focus on introducing *soft* state constraints [17, pp. 272-273] in which case the optimisation problem could be solved using a tailored ADMM implementation. Alternatively, the optimisation problem could be solved using FGM and the state constraints could be considered using Dykstra's method,

but it is unclear whether the constraint set could be partitioned as in Section 6.3.3 to allow for an efficient parallelisation.

Different versions of the MPC algorithm have been successfully tested on the Diamond storage ring. The tests have shown that MPC performs as well as the standard IMC algorithm, even when the FGM uses a limited number of iterations. Actuator saturation has not been tested on the real system and future research could focus on evaluating the behaviour of MPC under actuator saturation using the suggested input weight choices. Actuator saturation could either be artificially introduced by limiting the input rate further or by steering the beam to an initial position that is aligned with higher-order modes. The latter could be achieved using a reference signal that could also be used to identify the closed-loop output sensitivity, such as suggested in Section 4.6.

Appendix

6.A Obtaining the LQG Transfer Function

For a particular time delay n_τ, the LQG controller transfer function (6.51) can be obtained using appropriate software. However, the special structure of the observer matrices allow (6.51) to be obtained for any $n_\tau \in \mathcal{Z}_+$. First, substitute the measurement equation (6.46b) in the state-transition equation (6.46a) to obtain

$$\begin{pmatrix} \tilde{\bar{x}}_{k+1|k,i} \\ \tilde{d}_{k+1|k,i} \end{pmatrix} = A_{CL,i} \begin{pmatrix} \tilde{\bar{x}}_{k|k-1,i} \\ \tilde{d}_{k|k-1,i} \end{pmatrix} + \big(A_i - B_i \tilde{K}_{c,i}\big) \tilde{K}_{f,i} \tilde{y}_{k,i}, \tag{6.64}$$

where

$$A_{CL,i} \coloneqq \big(A_i - B_i \tilde{K}_{c,i}\big)\big(I - \tilde{K}_{f,i} C_i\big). \tag{6.65}$$

Taking the $\mathcal{Z}$-transform of (6.64) and substituting it in the LQR control law (6.49) yields the LQG control law as $\tilde{u}_i(z) = -c_{LQG,i}(z)\tilde{y}_i(z)$ with

$$c_{LQG,i}(z) \coloneqq \tilde{K}_{c,i}(I - \tilde{K}_{f,i} C_i)(zI - A_{CL,i})^{-1}(A_i - B_i \tilde{K}_{c,i})\tilde{K}_{f,i} + \tilde{K}_{c,i}\tilde{K}_{f,i}. \tag{6.66}$$

To simplify the notation for the remainder of this section, subscript i is dropped in the following. The terms $A - B\tilde{K}_c$, $I - \tilde{K}_f C \in \mathbb{R}^{(n_\tau+2)\times(n_\tau+2)}$ are computed separately as

$$A - BK_c = \begin{bmatrix} \pi_c & 0 & \dots & 0 & 0 & (\pi_c - 1)/\sigma \\ 1 & & & & & 0 \\ & \ddots & & & \vdots & \vdots \\ & & & 1 & 0 & 0 \\ 0 & \dots & & 0 & 0 & 1 \end{bmatrix}, \tag{6.67a}$$

$$I - K_f C = \begin{bmatrix} 1 & & & 0 \\ & \ddots & & \vdots \\ & & 1 & 0 \\ 0 & \dots & \sigma(\pi_f - 1) & \pi_f \end{bmatrix}, \tag{6.67b}$$

from which $zI - A_{CL}$ is obtained as

$$zI - A_{CL} = \begin{bmatrix} z-\pi_c & 0 & \dots & -(\pi_c-1)(\pi_f-1) & -(\pi_c-1)\pi_f/\sigma \\ -1 & z & & 0 & 0 \\ & \ddots & \ddots & \vdots & \vdots \\ & & -1 & z & 0 \\ 0 & \dots & 0 & -(\pi_f-1)\sigma & z-\pi_f \end{bmatrix}. \tag{6.68}$$

Using the LQR gain (6.45), the observer gain (6.48) and the closed-loop matrix (6.68), the terms $\tilde{K}_c(I - \tilde{K}_f C)$ and $(A - B\tilde{K}_c)\tilde{K}_f$ are computed as

$$\tilde{K}_c(I - \tilde{K}_f C) = \begin{bmatrix} k_c \\ 0 \\ \vdots \\ 0 \\ (1+k_c)(\pi_f-1) \\ \pi_f(1+k_c)/\sigma \end{bmatrix}^{\mathrm{T}}, \quad (A - B\tilde{K}_c)\tilde{K}_f = \begin{bmatrix} (\pi_c-1)k_f/\sigma \\ 0 \\ \vdots \\ \\ 0 \\ k_f \end{bmatrix}. \tag{6.69}$$

The inverse in (6.66) can be computed as

$$(zI - A_{CL})^{-1} = \frac{1}{\det(zI - A_{CL})} \operatorname{adj}(zI - A_{CL}), \tag{6.70}$$

where $\operatorname{adj}(A)$ is the *classical adjoint* or *adjugate* of A with entry p,k defined as $\operatorname{adj}(A)_{(p,k)} \coloneqq (-1)^{p+k} M_{k,p}$, where $M_{k,p}$ is the (p,k) *minor* of A [68, Ch. 0.8.2].

Abbreviating the off-diagonal elements of $zI - A_{CL}$ as $a_2 \coloneqq (\pi_c - 1)(\pi_f - 1)$, $a_3 \coloneqq (\pi_c - 1)\pi_f/\sigma$ and $a_4 \coloneqq (\pi_f - 1)\sigma$, the determinant of $zI - A_{CL}$ is computed for odd values of n_τ as

$$\begin{aligned} \det(zI - A_{CL}) =& (z-\pi_c) \det \begin{bmatrix} z & & & & 0 \\ -1 & \ddots & & & \\ & \ddots & z & & \\ & & -1 & z & \\ 0 & & & -a_4 & z-\pi_f \end{bmatrix} \\ &+ \det \begin{bmatrix} 0 & & & -a_2 & -a_3 \\ -1 & z & & & \\ & \ddots & \ddots & & \\ & & -1 & z & 0 \\ 0 & & & -a_4 & z-\pi_f \end{bmatrix} \\ =& (z-\pi_c) z^{n_\tau} (z-\pi_f) + (z-\pi_f)(-a_2)(-1)^{n_\tau - 1} - a_3 (-1)^{n_\tau} a_4 \\ =& z^{n_\tau}(z-\pi_c)(z-\pi_f) - a_2(z-\pi_f) + a_3 a_4. \end{aligned} \tag{6.71}$$

Considering the non-zero components in (6.69), the required elements of $\mathrm{adj}(zI - A_{CL,i})$ are computed as

$$\begin{aligned}
&\mathrm{adj}(zI - A_{CL})_{(1,1)} = z^{n_\tau}(z - \pi_f),\\
&\mathrm{adj}(zI - A_{CL})_{(1,n_\tau+1)} = z^{n_\tau-1}((z - \pi_f)a_2 + a_4),\\
&\mathrm{adj}(zI - A_{CL})_{(1,n_\tau+2)} = z^{n_\tau}a_3,\\
&\mathrm{adj}(zI - A_{CL})_{(n_\tau+1,1)} = z - \pi_f,\\
&\mathrm{adj}(zI - A_{CL})_{(n_\tau+2,1)} = a_4,\\
&\mathrm{adj}(zI - A_{CL})_{(n_\tau+2,n_\tau+2)} = z^{n_\tau}(z - \pi_{c,i}) + a_2.
\end{aligned} \tag{6.72}$$

With (6.69)–(6.72), the first term on the right-hand side of (6.66) is obtained as

$$\begin{aligned}
&K_c(I - K_fC)(zI - A_{CL})^{-1}(A - BK_c)K_f = \frac{1}{\det(zI - A_{CL})}\Bigg(\\
&k_c\Big(\mathrm{adj}(zI - A_{CL})_{(1,1)}\,\frac{(\pi_c - 1)k_f}{\sigma} + \mathrm{adj}(zI - A_{CL})_{(1,n_\tau+2)}\,k_f\Big) + (1 + k_c)\times\\
&(\pi_f - 1)\Big(\mathrm{adj}(zI - A_{CL})_{(n_\tau+1,1)}\,\frac{(\pi_c - 1)k_f}{\sigma} + \mathrm{adj}(zI - A_{CL})_{(n_\tau+1,n_\tau+2)}\,k_f\Big)\\
&+ \frac{\pi_f(1 + k_c)}{\sigma}\Big(\mathrm{adj}(zI - A_{CL})_{(n_\tau+2,1)}\,\frac{(\pi_c - 1)k_f}{\sigma} + \mathrm{adj}(zI - A_{CL})_{(n_\tau+2,n_\tau+2)}\,k_f\Big)\Bigg),
\end{aligned}$$

which can be simplified using appropriate software to obtain (6.51).

6.B Additional Results

Additional results for MPC with $r_i = \sqrt{\sigma_i}$ (6.63c) are shown in Fig. 6.B.1, where they are compared against $r_i = \sigma_i$ (6.63d).

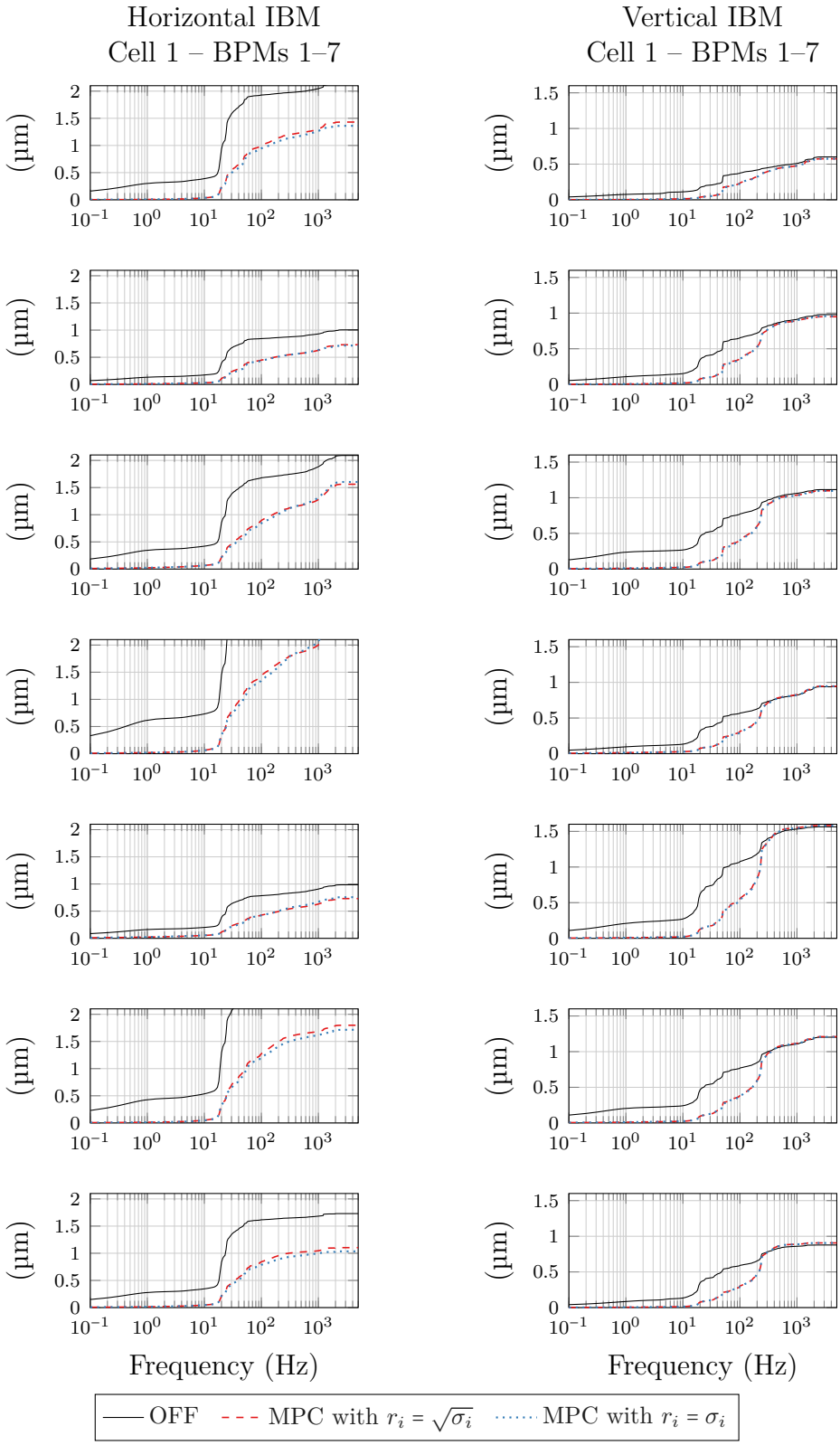


Figure 6.B.1: Measured IBM in the first cell of the Diamond storage ring for disabled feedback (OFF), single-array IMC from Section 6.4.1 and MPC algorithms with $r_i = \sqrt{\sigma_i}$ and $r_i = \sigma_i$.

7

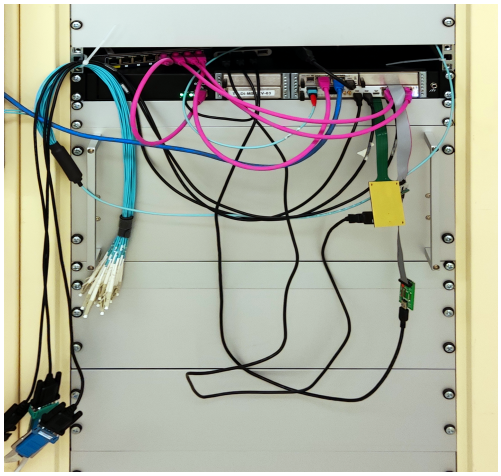
Control System Implementation at Diamond Light Source

As part of the FOFB design for Diamond-II, the controllers from Chapters 4 and 6 were implemented and validated in practice during experiments on the existing Diamond storage ring. To enable these experiments, a real-time control system was implemented on a VadaTech AMC540 – a board combining a Xilinx Virtex-7 FPGA with two Texas Instruments (TI) DSPs [152] – and interfaced with the communication infrastructure. In contrast to the existing FOFB that distributes the controller computations onto 24 processors, the experimental setup uses a single (centralised) computation node, which will also be the case for Diamond-II.

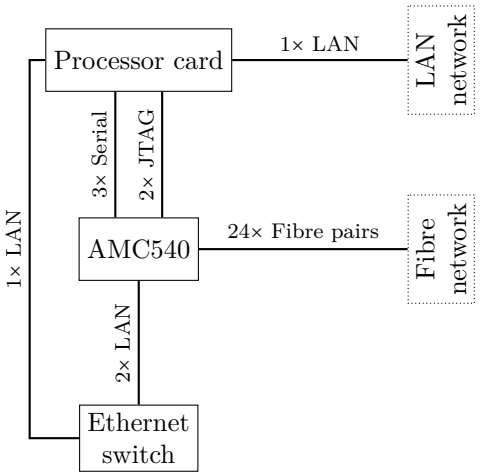
This chapter summarises the experimental setup and the implementation. Section 7.1 summarises modifications of the FOFB infrastructure, followed by the DSP in Section 7.2. The real-time control system implementation and its parallelisation on the DSP are presented in Section 7.3.

7.1 Control System Infrastructure

The existing FOFB uses a configurable number of $n_y \leq 173$ BPMs and $n_u \leq 172$ correctors. One Motorola MVME5500 processor [109] (abbreviated as VME in the following) per cell computes the setpoints for the magnets located in the



(a) AMC540 in cell 22.



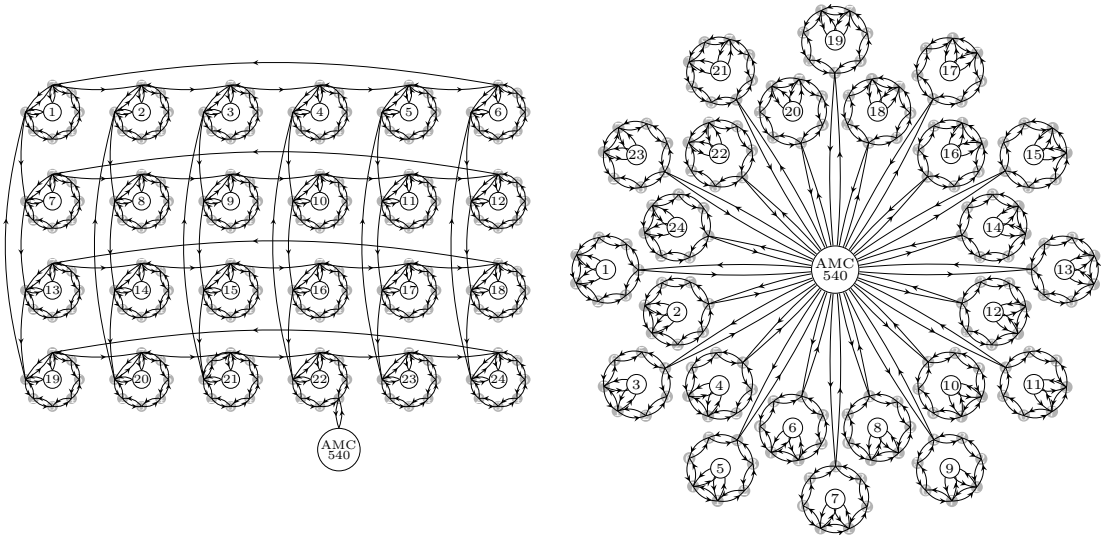
(b) AMC540 communication.

Figure 7.1: Configuration of cell 22 of the Diamond storage ring with fibres (blue), LAN (pink), JTAG and serial connections (black). The AMC540 is slotted into an μTCA with integrated eMCH. An additional processor card, an unmanaged ethernet switch and a USB hub (not shown) provide additional ports.

corresponding cell, where a lower-level power supply controller drives the magnet currents. The digitalised measurements are broadcast at a rate of 10 kHz over a fibre network that is coordinated by the Diamond *communication controller* (CC) [151]. All signals that are broadcast over the network are recorded by the fast-acquisition archiver [1].

Fig. 7.1a and 7.1b show the installation of the AMC540 in cell 22 of the Diamond storage ring. The AMC540 is slotted into a μTCA with integrated eMCH and connected[1] to the fibre network using the 24 fibre pair cables (light blue) on the left-hand side of Fig. 7.1a. For monitoring and debugging purposes, each device on the AMC540 – the FPGA and the two DSPs – is connected to the Diamond LAN network, which requires additional ports that are provided using a processor card and an unmanaged ethernet switch. The program code is loaded onto the FPGA and DSPs using two separate JTAG connections. The FPGA is used for signal routing, whereas the magnet setpoints are computed on the DSPs that are easier to program.

[1]The fibre cables were connected after the picture was taken.



(a) Torus mesh topology. **(b)** Star topology.

Figure 7.2: Communication network topology for Diamond (a) and Diamond-II (b).

7.1.1 Communication Network

Communication nodes The existing Diamond network topology is shown in Fig. 7.2a, which is also referred to as a *torus mesh topology* commonly found in parallel computing environments [151]. In Fig. 7.2a, the light gray circles represent BPM nodes and the larger nodes, which are numbered from 1 to 24, connect to the VME processors of each cell, which are referred to as VME nodes in the following. Each node consists of a Xilinx Virtex-II Pro FPGA on a PCI mezzanine card running the CC. The AMC540 is attached to the unused input-output ports of BPM 4 in cell 22. Data packets, such as measurements produced by the BPMs, can be passed from one node to another in direction of the arrows. When the system in Fig. 7.2a is started up, all BPMs are synchronised and begin to inject data packets at 10 kHz. The measurements are broadcast as 32 bit integers in nm together with an ID and a timestamp. New data packets are passed to neighbouring nodes, but packets that have already been processed are discarded. During experiments on the AMC540, the data traffic in the fibre network is roughly doubled, making it necessary to rearrange Fig. 7.2a to the *star topology* from Fig. 7.2b. By allowing each node to forward (or discard) fewer data packets, the data congestion in overloaded nodes is reduced in the case of the start topology, which will also be used for Diamond-II.

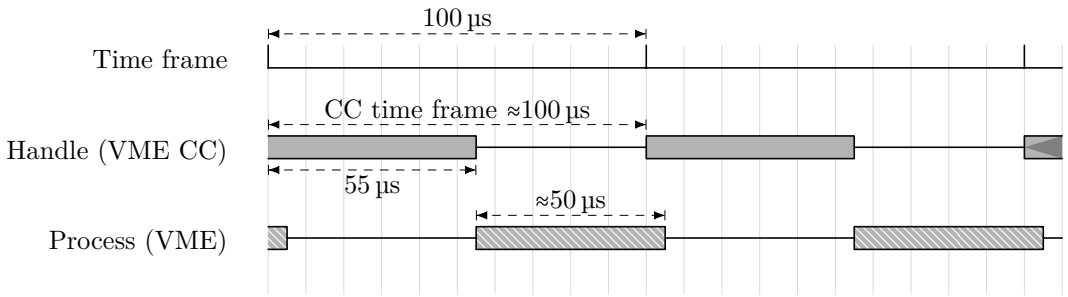


Figure 7.3: Timing diagram for network communication from the perspective of a communication controller (CC) running on a VME node. The arrival of a new BPM measurement triggers the start of the time frame and the collection period of 55 µs. At the end of the collection period, the measurements are transferred to the VME for computing corrector setpoints, which takes up roughly 55 µs [47, Tab. 2.7].

Communication controller In the communication scheme at Diamond, time is divided into time frames[2] of equal length of 100 µs. The communication scheme is event-based, and the start of the new time frame is determined by the arrival of new packets broadcast by the BPMs. The timing across the storage ring network is illustrated in Fig. 7.3 from the perspective of a VME node. The first line refers to the time frame of the VME node, which is triggered by the arrival of a new BPM measurement. The start of the time frame on the VME node marks the start of a *collection period* of configurable length, which is set to 55 µs in normal operation. In order to accommodate additional packets injected by the AMC540, the collection period on the VME node is increased to 75 µs during FOFB tests. Once the collection period has passed, the BPM nodes are idling until the start of the next time frame, and the VME nodes transfer the collected packets to the VME processor that computes the magnet setpoints and transfers them to the power supply controller. Computing the magnet setpoints on the VME processor takes roughly 50 µs.

7.1.2 Fast Orbit Feedback Code

The existing controller is implemented in C language on a VME processor. After configuration using a Python script, the following steps are executed every time new BPM data is received:

[2]Note that the sampling frequency at Diamond (and throughout the implementation) is in fact 10.072 kHz.

1. Multiplication with gain matrix and −1

2. Infinite impulse response (IIR) filtering

3. Control input slew-rate verification (max. ±1 A)

In addition to slew-rate verifications, the code also implements orbit (150 µm) and setpoint (5 A) verifications, and if any of the verification fails, the FOFB is shut down. Diamond implements several additional security measures, such as *BPM interlocks* that independently trigger a *beam dump* if a BPM measurements exceeds a particular limit.

After executing steps 1–3, the magnet setpoints are transferred as floating point numbers to the magnet power supply controllers in Amperes, where the setpoints are applied using 18 bit fixed-point arithmetic resulting in a granularity of roughly 40 µA. The following paragraphs summarise the individual components of the code and the reconfiguration required for FOFB testing. The precise limits for orbit and input amplitude verification vary depending on BPM and magnet and can be found under [79].

Configuration When the FOFB is started up, the VMEs are initialised using various parameters, such as the gain matrix or the IIR filter coefficients. Startup and configuration of the FOFB is managed by a Python program that has been modified for FOFB testing [80].

Gain matrix The BPM signals are multiplied with a fixed-size gain matrix $K \in \mathbb{R}^{172\times173}$, which is broadcast to the VMEs at startup. To provide a unity gain, the gain matrix is set to $K = \begin{bmatrix} I & 0 \end{bmatrix}$. By zeroing a row or column of K, the corresponding magnet or BPM can be disabled.

IIR filter The (scalar) controller of the IMC algorithm is an IIR filter, $IIR(z^{-1}) = \frac{b_0+b_1z^{-1}+\cdots+b_{n_b}z^{-n_b}}{1+a_1z^{-1}+\cdots+a_{n_a}z^{-n_a}}$, that is implemented in general form as

$$\begin{aligned} y[k] =& b_0u[k] + b_1u[k-1] + \cdots + b_{n_b}u[k-n_b] \\ & - a_1y[k-1] - a_2u[k-2] + \cdots + a_{n_a}u[k-n_a], \end{aligned}$$

where the arrays $[b_0, \ldots, b_{n_b}]$ and $[a_1, \ldots, a_{n_a}]$ are initialised during startup. For the Diamond controller, the IIR filter is defined as

$$IIR(z^{-1}) = \frac{(1-\epsilon)\frac{1-\lambda}{1-\tau}(1-\tau z^{-1})}{1-\lambda z^{-1}-(1-\epsilon)(1-\lambda)z^{-n_d-1}},$$

and in the Python program, the arrays b and a are initialised as

$$\begin{aligned} b &= \left[\tfrac{(1-\epsilon)(1-\lambda)}{1-\tau}, \quad \tfrac{-\tau(1-\epsilon)(1-\lambda)}{1-\tau}, \quad 0, \quad \ldots\right] \\ a &= \left[-\lambda, \quad 0\times(n_d-1), \quad -(1-\epsilon)(1-\lambda), \quad 0, \quad \ldots\right], \end{aligned}$$

where n_d denotes the total time-delay in terms of time steps and the implementation is such that $n_a, n_b \leq 9$. By providing the values $b_0 = 1$ and $a_i = b_i = 0$, $i \geq 1$, the filter is bypassed. The parameter λ is defined as $\lambda = \exp(-T_s/(\eta n_d T_s))$, where the parameter η is used to modify the closed-loop bandwidth.

Slew-rate verification The slew-rate of the computed control signal $y[k]$ is compared against a lowpass-filtered signal $\hat{y}[k]$, where

$$\hat{y}_k = \frac{a_l}{a_l+2/T_s}y_k + \frac{a_l}{a_l+2/T_s}y_{k-1} - \frac{a_l-2/T_s}{a_l+2/T_s}\hat{y}_k,$$

and $a_l = 2\times 2\pi\,\mathrm{rad\,s^{-1}}$. The feedback is then stopped if $|y_k - \hat{y}_k| > 1\,\mathrm{A}$. The same slew-rate constraint is implemented in the MPC algorithm.

Bypassing the existing system The VME nodes recognise BPM packets by their IDs, which in normal operation range from 1 to 173. For FOFB testing, the BPMs are renumbered and their IDs shifted to an unused range, such as shown in Table 7.1. After renumbering, the BPM data packets are ignored by the VME nodes, but collected by the CC on the AMC540, which passes the packets to the

Table 7.1: Reconfiguration of network IDs for FOFB testing.

Network IDs	0	1–173	174–255	256–428	429–511
Default Mode	reserved	BPM	Misc.	Misc.	-
AMC540 Mode	reserved	AMC540	Misc.	BPM	-

DSPs for computing the magnet setpoints. Upon completion of the DSP processes, the CC on the AMC540 assigns IDs 1–172 to the magnet setpoints and injects them back into the network, where they are collected by the VME nodes. By configuring the FOFB parameters as feed-through on startup, the existing FOFB controller on the VMEs is bypassed without changing the existing FOFB code. To accommodate the 32 bit integer packet format, the setpoints are scaled to µA on the DSPs, and the gain matrix K is used to scale the setpoints back to Amperes on the VMEs.

7.1.3 Latency

In regular operation, the total latency affecting the FOFB is 700 µs. With the AMC540, the latency is increased by two time steps, which is explained using the timing diagram from Fig. 7.4. The green path illustrates the path of a data packet starting with its creation at a BPM node and ending with the transfer to a power supply controller. After being collected by the CC on the AMC540, the packet is transferred to the DSP (Process AMC540), where the magnet setpoints are computed. At the time at which the magnet setpoints become available, the data collection period of the following time step has already started. In principle, the setpoints could be injected back into the network (red line). In practice, injecting the setpoints at the following time step leads to network errors, as the packets are either getting stuck in a CC queue (when injected at the beginning of the following time step), or do not reach all nodes by the required time (when injected towards the end of the following collection period). The CC on the AMC540 has therefore been modified such that it buffers the setpoints, before injecting them at the start of the next time frame. With this modification and noting that the latency introduced by the process on the VMEs remains unchanged, the total latency becomes 900 µs for FOFB testing compared to 700 µs in normal operation.

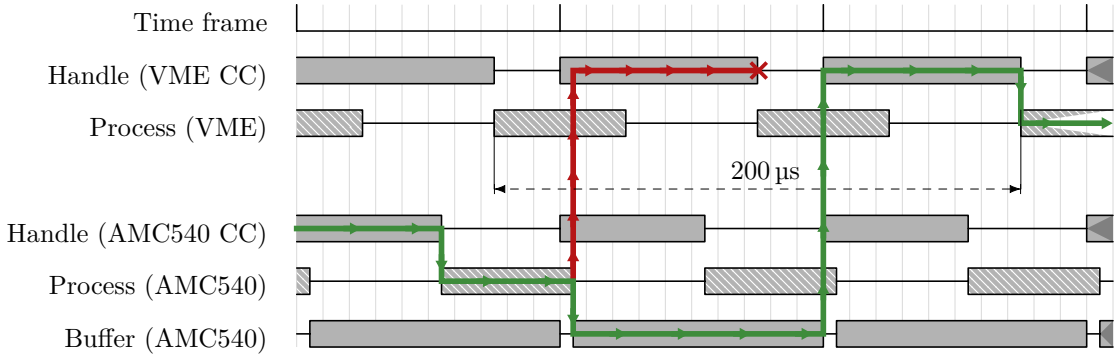


Figure 7.4: Timing diagram for network communication from the perspectives of the AMC540 node and a VME node. The green path illustrates the data path, starting from the communication controller (CC) on the AMC540 node and ending in the VME processor. Without buffering in the AMC540, the VME node is unable to collect all data packets (red path) eventually leading to a network error.

7.2 C6678 Digital Signal Processor

The AMC540 embeds two TI C6678 octacore DSPs that are used for computing the magnet setpoints [145]. One DSP (`DSP0`) is used for the control problem in the horizontal direction, and the other DSP (`DSP1`) for the vertical direction. The controller is implemented in C, compiled using TI's software generation tools and uploaded to the DSP using *code composer studio* (CCS) [143]. The C6678 is highly configurable, which is exploited to increase the computational efficiency of the controller. In particular, memory partitioning and alignment of problem data are specified to minimise cache and memory transactions; compiler optimisation is facilitated through preprocessor directives; direct memory access (DMA) engines are used to speed up data transfer; and interprocessor communication (IPC) overheads are minimised by implementing an interrupt-free IPC framework.

7.2.1 Memory and Cache

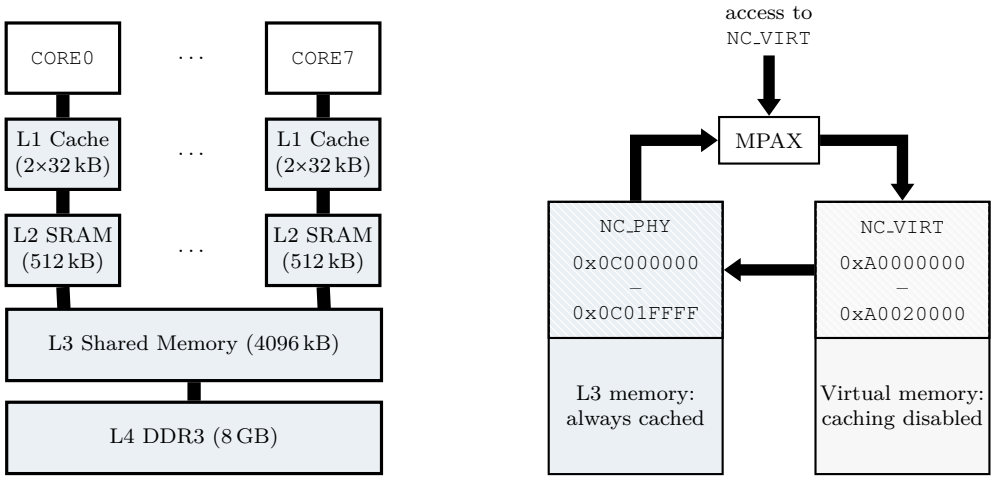
Memory configuration On the AMC540, each C6678 DSP has four (data) memory levels, L1–L4, with the L1 and L2 levels considered local [145, Ch. 5]. The L1 and L2 levels can be configured as either cache or SRAM, but using the L2 level as a cache improves the performance of DDR3 memory accesses only [138].

Without cache, the DSP requires 7 processor cycles for a single read-access to the L2 memory, 22 to the L3 memory, and 30 to the L3 memory [147, Table 3].

Fig. 7.5a shows the memory configuration used for the control system implementation. The L1 level is configured as a cache, whereas the L2 memory is used for core-local data, such as matrices and vectors that are repeatedly used in the control algorithm. The DDR3 memory is used for initialisation. In the C code, the linker command file is used to specify the available memory and create custom memory sections that are allocated to different memory levels [141]. Pre-processor directives, such as `SET_DATA_SECTION` and `DATA_ALIGN`, can be used to place variables in custom memory sections and align them to cache line boundaries.

On the TI C6678, the shared L3 memory is always cached by the L1 cache. However, some parts of the C code, such as IPC, are faster if caching of certain L3 sections is disabled. Fig. 7.5b illustrates the technique used in this project to create a non-cacheable shared memory section. Using the linker command file, two new memory levels, `NC_PHY` and `NC_VIRT`, are created, where the address range of `NC_PHY` coincides with the start of the L3 level, and the address range of `NC_VIRT` is beyond the physical address range of the DSP memory. During DSP initialisation, the *memory protection and address extension* (MPAX) is used to map `NC_VIRT` section back to `NC_PHY` [145, Ch. 7.3]. In contrast to the shared memory address range, caching can be disabled for `NC_VIRT`, and certain variables, such as those used for IPC, are allocated to the non-cacheable `NC_VIRT` section.

Cache configuration The L1 cache is a 32 kB two-way set associative cache and the two lines per set are replaced using a least-recently-used strategy, meaning that the most recently used line is kept in the cache. One cache line is 64 B long and cache operations can only be executed on one or several cache lines. The cache line size imposes constraints on the parallelisation; For a matrix-vector multiplication that is distributed onto several cores, the shortest 32 bit array that can be processed by a single core has length 16.



(a) Memory configuration. **(b)** Non-cacheable access of L3 memory.

Figure 7.5: Memory configuration of each TI C6678 on the AMC540 (a) and non-cacheable access of L3 memory used in the cIPC framework (b)
.

On the C6678, coherence between the L1 cache and the L3 memory is not guaranteed. The *chip support library* (CSL) library provides an interface to interact with the cache [136, Ch. 2] and force invalidate or write-back operations. As these instructions operate on entire cache lines, shared variables and arrays must be aligned to cache line boundaries using pre-processor directives.

The cache is a read-allocate-only cache and will fetch lines only on a read miss. Before a memory-intensive operation, the cache can be warmed-up to improve its performance by avoiding non-compulsory cache evictions [138, Ch. 3.4.6], which is also used for the control system implementation [78, `utils/touch.asm`]. To avoid cache conflict misses, which are caused by arrays being mapped to the same cache sets, arrays and variables can be re-arranged using memory sections. For the control system implementation, all shared arrays fit into the L1 cache and arranging the arrays contiguously in memory therefore guarantees that each vector is mapped to a different cache set [138, Ch. 3.4.4].

7.2.2 Interprocessor Communication

To compute the magnet setpoints at a rate of 10 kHz, the workload of large arithmetic operations, such as matrix-vector multiplications, is distributed onto several cores

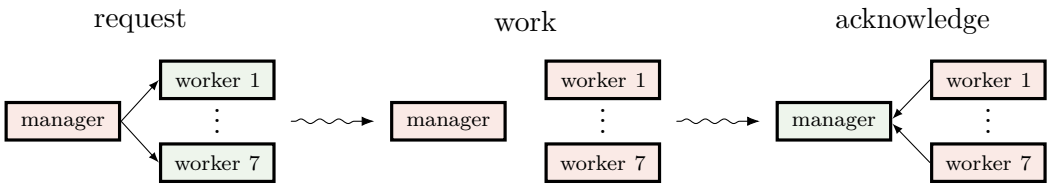


Figure 7.6: Manager-worker scheme used for the control system implementation.

using a manager-worker scheme as depicted in Fig. 7.6. Upon a manager request, the workers execute a task and upon task completion, the workers notify the manager. If the parallelised task uses shared data, it is also necessary to trigger the corresponding cache invalidate or write-back operations.

On the TI C6678, there exist several IPC modules and out of these modules, the Notify module [142, Ch. 2.7] and the Multicore Navigator [146], which benefits from its own physical layer, are the most suitable for the manager-worker framework[3]. The TI notification schemes are flexible but introduce a considerable delay. Fig. 7.7 shows the time required for executing the sequence from Fig. 7.6 under zero workload for 1-7 worker cores. For 7 worker cores, it can be seen that the notify and navigator libraries introduce an overhead of more than 10^4 cycles, which corresponds to 10 µs if the processor is clocked at 1 GHz. The 10 kHz sample frequency of the present application allows for 100 µs (10^5 processor cycles) and since the control algorithms require several communication steps, the available modules are too slow.

To reduce the overhead, a fast but minimal IPC framework was implemented, which is referred to as custom IPC (cIPC) in the following [77]. Next to initialisation and cache configuration, the cIPC framework consists of the 4 functions shown in Table 7.2. The four functions modify the integers `req_i` and `ack_i`, `i`$= 1, \ldots, n_w$, where n_w is the number of worker cores. In the code dedicated to the manager core, the function `manager_set_req` is used to place a request, while in the code dedicated to the worker cores, the (blocking) function `worker_wait_req` is used to wait for a request. The function calls to `manager_set_req` and `worker_wait_req` must be followed by `manager_wait_ack` and `worker_set_ack`. If the flags are

[3]The MessageQ module [142, Ch. 2.3] was also benchmark, but performed worse and is not further considered.

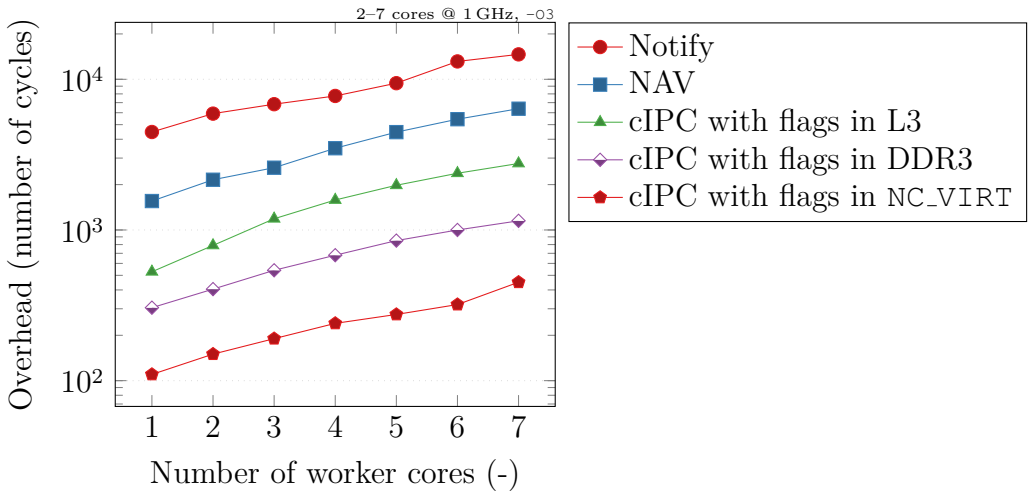


Figure 7.7: Comparison of IPC overheads for the manager-worker scheme under zero workload. If the cores are clocked at 1 GHz, 1000 cycles correspond to 1 µs.

Table 7.2: Modification of flags by custom IPC framework.

Functions	Flags	
	`req_i`	`ack_i`
`void manager_set_req(int R)`	$0 \longrightarrow R$	
`int R = worker_wait_req()`	$R \longrightarrow 0$	
`void worker_set_ack(int A)`		$0 \longrightarrow A$
`int A = manager_wait_ack()`		$A \longrightarrow 0$

initialised to zero and if the correct sequence of requests and acknowledgements is implemented, the implementation from Table 7.2 guarantees that manager and workers do not simultaneously modify the flags `req_i` and `ack_i`.

An example usage is shown in Listings 7.1–7.2, which is taken from the MPC implementation. One Line 2 of Listing 7.2, the workers are polling the flag `req_i` and waiting for it to be changed from 0 to a non-zero value by the manager core on Line 1 of Listing 7.2. When the value of `req_i` changes, the workers acknowledge the receipt of the request on Line 2, before workers and manager proceed with performing a task, which can contain additional calls of cIPC functions. Note that if the function call to `worker_set_ack` on Line 2 of Listing 7.2 was omitted, the manager core would indefinitely spin in `manager_wait_ack` on Line 2 of Listing 7.1. The library [77] contains several additional functionalities that are not detailed here.

Three different versions of the cIPC framework were benchmarked and the

resulting overheads are compared in Fig. 7.7. The three versions differ with respect to where the flags `req_i` and `ack_i` are placed in memory. For the first cIPC version, the flags are placed in the shared L3 memory for which caching cannot be disabled (Section 7.2.1). Compared to the Navigator, the overhead is reduced by a factor of 2–3 depending on the number of worker cores. For the second cIPC version, the flags are placed in DDR3 memory for which caching is disabled by default [138]. Even though read-accesses to the DDR3 memory are more than three times slower than read-accesses to the L3 memory [147, Table 2.7], the second version of cIPC is over two times faster than the first version, which suggests that repeated cache-invalidate operations are introducing a considerable overhead. This observation led to the third cIPC version, which uses a virtual memory section that is mapped back to the L3 memory, but for which caching can be disabled. With the third version, cache operations can be omitted, which further halves the overhead compared to the second version. The third and final version of the cIPC framework is over 10 times faster than TI's Navigator.

Listing 7.1: Custom IPC manager code used in the function `mpc_ctr` of the MPC implementation.

```
manager_set_req(restart);
manager_wait_ack();
// subsequent functions
   contain further requests
if (restart == RESTART) {
  obs_reset_manager();
  fgm_reset_manager();
} else {
  obs_update_manager();
  fgm_solve_manager();
}
return fgm_solution();
```

Listing 7.2: Custom IPC worker code used in the function `mpc_ctr_worker` of the MPC implementation.

```
while (1) {
  req_val =
     worker_wait_req();
  worker_set_ack(1);
  if (req_val == RESTART) {
    obs_reset_worker();
    fgm_reset_worker();
  } else {
    obs_update_worker();
    fgm_solve_worker();
  }
}
```

7.2.3 FPGA-DSP Interface

The AMC540 provides 2×4 general purpose input output (GPIO) ports [139], 2×2 peripheral component interconnect express (PCIe) lanes (generation 2) and 4 serial

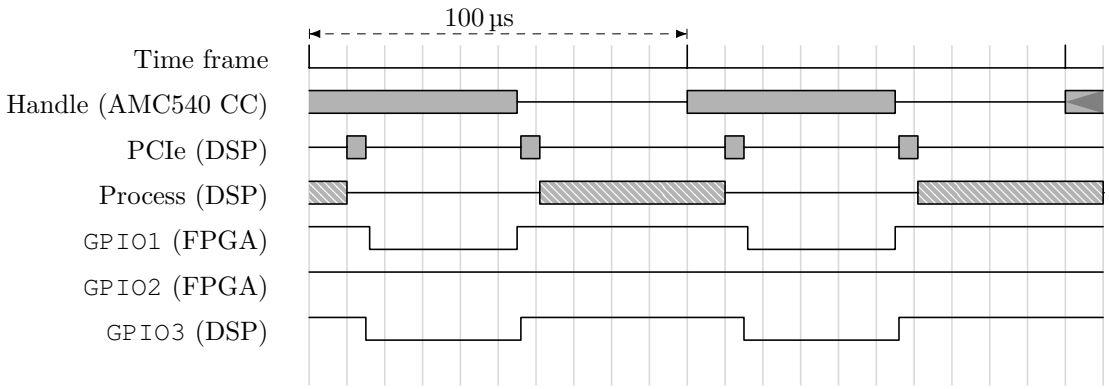


Figure 7.8: Timing diagram for DSP-FPGA communication and data transfer for a CC timeframe of 55 µs. The diagram ignores the additional buffering of outgoing data shown in Figure 7.4.

rapit input-output (SRIO) ports that interface the FPGA with each DSP [144]. For the control system implementation, three GPIO ports per DSP are used for communication and two PCIe lanes per DSP for data transfer.

Communication (GPIO) Fig. 7.8 illustrates the communication between one DSP and the FPGA, where the three GPIO ports `GPIO1` and `GPIO2` are driven by the FPGA and `GPIO3` by the DSP. Upon arrival of new measurements, the FPGA changes `GPIO1` from low to high. The DSP then sets `GPIO3` high and triggers the data transfer. `GPIO3` remains high until the data has been processed and transferred back to the FPGA, and is then set to low, upon which the FPGA sets `GPIO1` low.

`GPIO2` serves as a reset signal and is driven by manual intervention. If `GPIO2` is low, the DSP re-initialises the control algorithm and zeroes out the values transferred back to the FPGA. Because the DSP is unaware of whether the feedback system is running or not, `GPIO2` is required to avoid actuator wind-up during open-loop operation. Note that the FPGA-DSP interface does *not* implement any protection against the case that the DSP process does not finish before new data has arrived. In that case, the FPGA does not broadcast new values, which eventually leads to a network error and a shutdown of the FOFB.

Data transfer (PCIe) The PCIe link that connects the DSP to the FPGA is configured such that it is managed by the DSP, and the FPGA provides the

addresses of two 1000×32 bit large block RAMs. The data transfer is triggered by the DSP and executed by the (DSP-local) DMA engine. Using the DMA engine, the theoretical PCIe (read or write) throughput[4] on the C6678 is 6.10 Gbps-6.92 Gbps depending on the payload size and the DMA configuration [147, Ch. 6]. However, the throughput realised in practice is roughly 5.5 Gbps [147, Fig. 12–15].

The DMA transfers are configured to read or write 256 32 bit integers between the FPGA RAM and the L3 memory of the DSP. With this configuration, the data transfer requires 5.3 µs (7449 cycles), which corresponds to 1.54 Gbps and is roughly 4 times slower than the theoretical throughput [147]. However, for transferring 1024 32 bit integers, the throughput is measured as 5.05 Gbps, which corresponds to 92 % of the throughput reported in [147]. The difference in throughput was not investigated further, but is likely to be related to overheads introduced by the DMA management. The DMA configuration used for this project can be found under [78, `utils/libQDMA.h`].

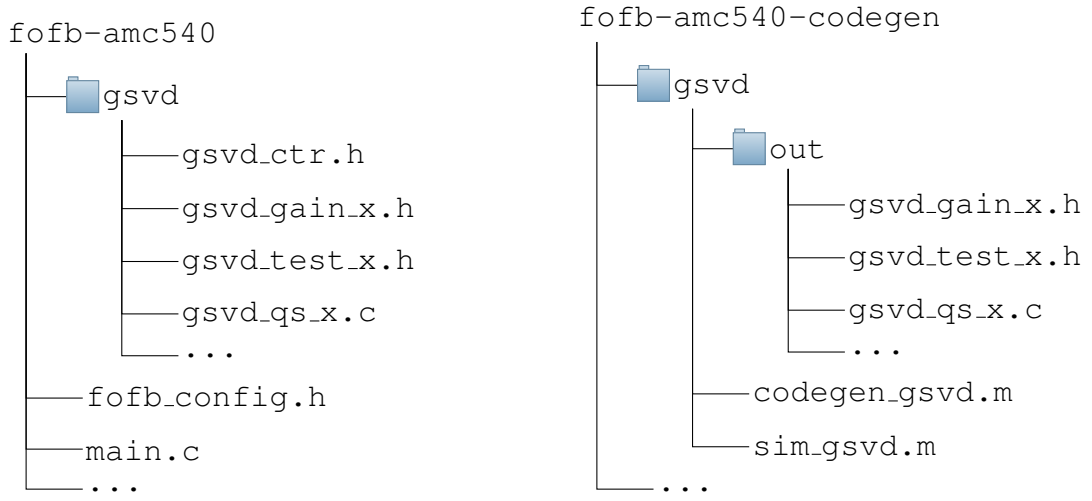
7.3 Implementation

The program code is split into code generation tools [79], which are written in Matlab, and C code [78], which is tailored to the C6678 DSP. The C code is re-used in a hardware-in-the-loop (HIL) implementation [81] for the C6678 evaluation board module that connects through Ethernet to a computer with a Matlab simulation of the plant model.

7.3.1 Code Generation Tools

The directory trees of the C code and code generation tools are illustrated in Fig. 7.9, where the folder `gsvd` refers to the GSVD-based two-array controller. The repositories [78], [79], [81] also provide implementations of single-array IMC (`imc`) and MPC (`mpc`). The code generation tools are used for producing C files that change based control algorithm, storage ring configuration and model parameters.

[4] The theoretical data throughput for 2 lanes with disabled ECRC generation and 8 bit/10 bit encoding is computed as $64\,\mathrm{B}/(64\,\mathrm{B}+20)\times(8/10)\times(2\times5\,\mathrm{Gbps}) = 6.10\,\mathrm{Gbps}$ for 64 B packets and analogously for 128 B packets.



(a) C project [78]. **(b)** Code generation tools [79].

Figure 7.9: Overview of the project structure for the control system implementation.

Fig. 7.9b shows a subset of the files produced for the `gsvd` algorithm. Based on an ORM and a storage ring configuration, which specifies which BPMs or magnets are disabled, the main script, `codegen_gsvd.m`, produces a range of C header files and functions. In the example from Fig. 7.9b, the header file `gsvd_gain_x.h` contains controller gain matrices for the horizontal control direction that are distributed onto the local memories of the eight DSP cores for parallelising matrix-vector multiplications. The header file `gsvd_test_x.h` contains simulation results that are used for unit tests and are produced using the function `sim_gsvd.m`, which accurately replicates the C implementation and considers practical aspects such as scaling and rounding of signals. The main script also produces C functions, such as `gsvd_qs_x.c`, which contains an automatically generated IIR filter that has been optimised for the DSP architecture and only considers non-zero coefficients of the IIR filter. Note that the C files produced by the Matlab script are re-used in the C code from Fig. 7.9a.

7.3.2 Program Flow

The C implementation of the controller is tailored to the C6678 DSP and parallelised using the cIPC framework from Section 7.2.2. The number of worker cores varies depending on the algorithm and the storage ring configuration and can be set in the

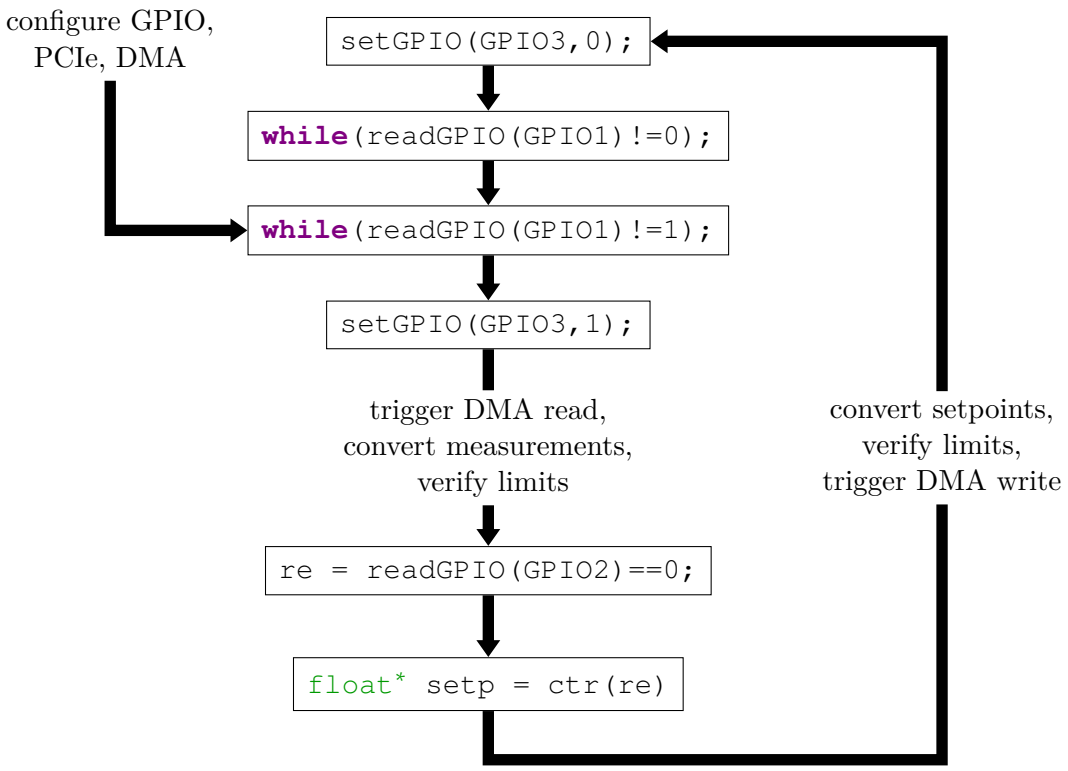


Figure 7.10: Program flow for the manager core.

code generation tools as well as in `fofb_config.h` of the C project (see Fig. 7.9a). The same executable is loaded onto all cores of the DSP and based on the core ID, the program branches off into manager or worker tasks. One core that is not used for parallelisation is reserved for communication through LAN.

After initialising the peripherals, the manager core enters an infinite while-loop shown in Fig. 7.10. As in Fig. 7.6, the GPIO port `GPIO1` is driven by the FPGA and changes from low to high when new measurements arrive. Upon a rising edge of `GPIO1`, the manager core sets `GPIO3` high, reads and converts the measurements from the FPGA memory, and reads the restart signal, `GPIO2`. The setpoints are computed in `ctr`, which calls the parallelised and algorithm-specific functions, and then scaled to µA and transferred back to the FPGA.

7.3.3 Partitioning of Matrix-Vector Multiplications

For parallelising algebraic operations on the 8 cores of the C6678 DSP, an adaptation of the *basic linear algebra subprograms* (BLAS) exists for the C6678 DSP, which

is referred to as the LINALG library [148]. The LINALG library is combined with TI's version of the *open multiprocessing* (OpenMP) toolbox, which builds upon TI's Notify library. As shown in Section 7.2.2, the Notify library introduces a considerable overhead relative to the available computation time, which is why the LINALG library is too slow for the present project.

All control algorithms tested in this project require matrix-vector multiplications of the form $y = Ax + b$, where $x \in \mathbb{R}^n$, $y, b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$. The vectors y and x that are shared between cores are saved in L3 memory and the core-specific problem data, A and b, is stored in L2 memory, and possibly updated after the execution of the matrix-vector multiplication. Note that the core-specific problem data does not fit in L1 memory, which is why the L1 memory is only used as a cache.

The partitioning of the problem data across cores is illustrated on the left-hand side of Fig. 7.12, where it is assumed that 6 worker cores are used. For the tests on the Diamond storage ring, the dimensions $m \times n$ are equal to 172×173 for MPC, and 96×96 (slow correctors) or 62×96 (fast correctors) for dual-rate IMC. With an L1 cache line size of 64 B and assuming single-precision floating-point arithmetic, the problem data can be partitioned into blocks of length $16p$, where $p \in \mathbb{Z}_{++}$. After preliminary tests, it was decided to zero-pad the MPC problem data from 172×173 to 196×196 and parallelise the matrix-vector multiplication onto 6 worker cores with each core processing $m_{\text{worker}} = 32$ elements of y. For dual-rate IMC, the problem data is zero padded from 96×96 to 128×128 or from 62×96 to 64×128, and the matrix-vector multiplication parallelised onto 4 worker cores with each core processing $m_{\text{worker}} = 32$ or $m_{\text{worker}} = 16$ elements of y.

7.3.4 Performance

Single-core performance The execution time of algebraic operations on a single core of the TI C6678 DSP can be reduced by using compiler annotations (pragmas) to provide the compiler with the necessary information to optimise the execution of the workload [140]. In addition, single-instruction multiple-data (SIMD) instructions can be used explicitly as compiler intrinsics in the C code [137], which allow parallelism

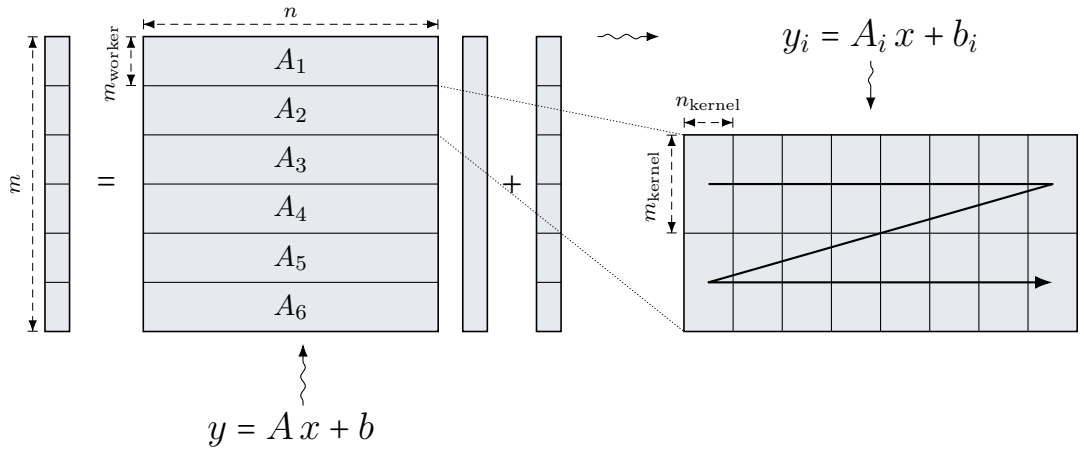


Figure 7.11: Matrix partitioning for parallelisation (left) and single-core kernel (right).

within each core by operating on 64 bit or 128 bit vectors (2 or 4 packed floats). For the control system implementation, the SIMDs `_dmpysp` and `_qmpysp` were used, which allow to simultaneously execute 2 or 4 single-precision multiplications, and `_daddsp`, which allows to simultaneously add 2 floats. For the result of `_qmpysp`, the `_lof2_128` and `_hif2_128` intrinsics were used to extract the upper or lower 64 bit result. To enable the efficiency of the SIMDs, the problem data is reorganised such that the operands of the SIMDs are arranged contiguously in memory enabling double-word loads [137].

Assuming full core-level parallelism and using all available arithmetic units, one C6678 core running at 1 GHz is capable of carrying out 16 single-precision multiply-add operations per cycle [5], or 16 *giga floating-point operations per second* (16 Gflop/s). However, for the standard `SGEMM` operation $y = AB + \alpha y$, the peak performance of the BLAS library on a C6678 core was reported as 10.3 Gflop/s for square matrices with 4096 rows [5], which is about 64 % of the maximum core performance. For matrices with 32 rows, the single-core `SGEMM` performance is reported as 2 Gflop/s only, which shows that `LINALG` is not optimised for the problem sizes of the control system implementation at Diamond.

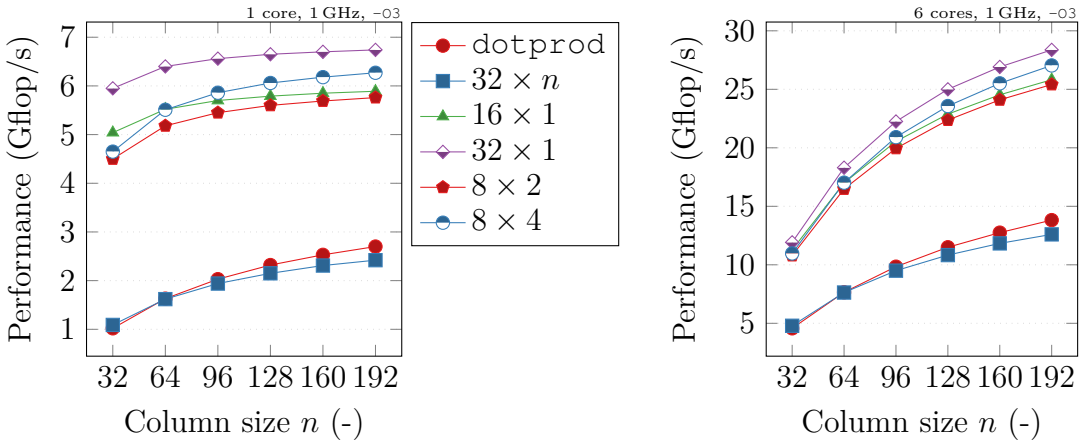
According to Section 7.3.3, each core must implement a matrix-vector multiplication $y_i = A_i x + b_i$, where y_i is of length 16 or 32 and x of length 128 or 196. For performing algebraic operations on the C6678 DSP, a single-core maths library, `DSPLIB`, exists [148]. Among the `DSPLIB` functions, the function

`DSPF_sp_dotprod`, which computes the dot-product from two single-precision arrays, can be used to perform the matrix-vector multiplication. However, the following measurements will show that `DSPF_sp_dotprod` is too slow for matrix-vector multiplications.

The right-hand side of Fig. 7.12 shows the partitioning of A_i for the customized single-core matrix-vector multiplication, $y_i = A_i x + b_i$. The matrix A_i of size $m_{\text{worker}} \times n$ is partitioned into kernels of size $m_{\text{kernel}} \times n_{\text{kernel}}$, and the multiplication of a kernel with a n_{kernel} long block of x forms the body of the innermost loop. The matrix A_i is traversed row-wise and the kernel-multiplication results are accumulated into a vector of length m_{kernel}, which is chosen small enough to retain the m_{kernel} values in core registers.

For finding the optimal kernel size, the operation $y_i = A_i x + b_i$ is timed for different combinations of $m_{\text{kernel}} \times n_{\text{kernel}}$ and different SIMDs. The results are compared in Fig. 7.12a, where version "$32 \times n$" uses no SIMDs and does not rearrange the data, versions "32×1", "16×1" and "8×2" use the `_dmpy` and "8×4" the `_qmpy` SIMD instruction. For a single core running at 1 GHz and the operation $y_i = A_i x + b_i$, the performance is measured in Gflop/s as $2 m_{\text{worker}} n$ divided by the number of elapsed processor cycles [5]. The results from Fig. 7.12a show that `DSPF_sp_dotprod` (—●—) and version "$32 \times n$" (—■—) attain a peak performance of 2.7 Gflop/s and 2.4 Gflop/s, respectively, which corresponds to 16 % of the maximum core performance and is comparable to the BLAS results for small matrix sizes [5]. With the partitioning of A_i, the performance is significantly increased, with the 32×1 (—⟡—) and 8×4 (—○—) kernels reaching peak performances of 6.7 Gflop/s (42 %) and 6.3 Gflop/s (39 %), respectively. Because some of the control algorithms at Diamond require to perform $y_i = A_i x + b_i$ with A_i having 16 rows, which is not supported by the 32×1 kernel, the control system implementation uses the 8×4 kernel.

Multi-core performance The single-core implementations from Fig. 7.12a are combined with the cIPC scheme from Section 7.2.2 to distribute $y = Ax + b$, $A \in \mathbb{R}^{192 \times n}$,



(a) $y_i = A_i x + b_i$, $A_i \in \mathbb{R}^{32 \times n}$. **(b)** $y = Ax + b$, $A \in \mathbb{R}^{192 \times n}$.

Figure 7.12: Comparison of single- (left) and multi-core (right) matrix-vector multiplication for different kernel sizes $m_{\text{kernel}} \times n_{\text{kernel}}$ (legend) and column sizes n. The arrays y, y_i and x are located in shared L3 memory, and A, A_i, b and b_i in L2 memory.

onto 6 worker cores[5]. The results are shown in Fig. 7.12b. In contrast to the expected $6 \times 6.3\,\text{Gflop/s} = 37.8\,\text{Gflop/s}$, the 8×4 kernel ( ) reaches 27 Gflop/s for $n = 192$. The performance loss is associated with the cIPC scheme, which adds roughly 300 processor cycles (15 % of the single-core cycles), and cache invalidate and writeback operations, which add roughly 400 cycles (20 % of the single-core cycles). Compared to the single-core case, parallelising the operation $y = Ax + b$ yields a speed-up of (27 Gflop/s)/(6.3 Gflop/s)= 4.3. The speed-up can be compared to the performance of the BLAS function `SGEMM` for square matrices of size 256 [5, Fig. 8]. On a single core, the BLAS library reaches a performance of 7.2 Gflop/s, whereas on 8 cores, the BLAS library reaches a performance of 24 Gflop/s, which corresponds to a speed-up of (24 Gflop/s)/(7.2 Gflop/s)= 3.3 and is therefore comparable to the results from Fig. 7.12b.

7.4 Conclusions

To enable tests of advanced feedback algorithms on the Diamond storage ring, an AMC540 board was embedded in the existing FOFB infrastructure and a framework

[5]The manager core does not contribute to the matrix-vector multiplication.

developed that provides code generation tools [79], a startup interface [80], a real-time control system [78] and a HIL simulation [81]. The real-time control system is implemented in C and can be configured to run single-rate IMC, dual-rate IMC or MPC. Using minor modifications, any other control algorithm can be implemented and tested with the provided framework.

At the start of the control system implementation, one of the 8 DSP cores was reserved for UDP communication, but the UDP functionalities were only finalised for the HIL simulation. For tests on the storage ring, the UDP functionalities could be a valuable tool for debugging as well as for initialising and modifying controller parameters. In addition, the UDP communication could be used to provide a reference signal for the orbit, which could in turn be used for closed-loop system identification.

The TI C6678 is a high-performance octacore DSP that can reach up to 8×16 Gflop/s. However, the measurements from Sections 7.2.2, 7.2.3 and 7.3.4 have shown that the computational efficiency of matrix-vector multiplications is limited, and a performance of 6×4.5 Gflop/s=27 Gflop/s was reached for the control system implementation, which corresponds to 28 % of the maximum performance only. For the small matrix sizes of the control problem at Diamond, unavoidable overheads associated with IPC or cache operations significantly increase the computation time.

The communication between FPGA and DSP was realised using GPIO connections, and the transfer of BPM measurements and magnet setpoints using PCIe. While the overhead associated with the GPIO communication is negligible, the overhead associated with the PCIe transfer is not. For transferring 1024×32 bit values over PCIe, a throughput of 5.05 Gbps was obtained, but for 256×32 bit values, a throughput of 1.54 Gbps was measured, which is 4 times slower than the theoretical throughput of 6.1 Gbps.

The AMC540 is a high-performance board, and the embedded DSPs are flexible to program and highly customisable. However, the overhead associated with the FPGA-DSP data transfer alone already exceeds the sample time of Diamond-II. While the developed control system achieves the 10 kHz speed requirement of

Diamond and can be used for testing of the Diamond-II algorithms, the controller would need to be implemented on the FPGA for achieving the Diamond-II sampling frequency of 100 kHz.

8 Conclusions and Future Work

The aim of this thesis to design FOFB controllers for the two-array CD system of Diamond-II. Compared to the existing FOFB, the number of BPMs and correctors at Diamond-II is increased from 173×172 to 252×396 and the sample frequency from 10 kHz to 100 kHz, significantly raising the computational requirements of the controller. Two control algorithms were proposed that are applicable to two-array CD systems: GSVD-based IMC and MPC. After installing a new centralised control system in the existing Diamond storage ring, both controllers were implemented on specialised hardware and tested on the real-world system. Based on the theoretical and practical results of this thesis, the novel GSVD-based controller was adopted for the Diamond-II upgrade.

While MPC is applicable to an arbitrary number of actuator arrays and can consider constraints, the results from this thesis have shown that the high sampling frequency and the large number of inputs and outputs make its application to electron beam stabilisation difficult. In addition, the ill-conditioned ORM complicates the tuning process, introduces numerical inaccuracies and reduces the convergence speed of the solver. Although the FOFB specifications of the existing Diamond storage ring were met, the parallelised implementation on the octacore TI C6678 DSP demonstrated that even using high-performance hardware, the Diamond-II sampling frequency of 100 kHz cannot be reached using the TI C6678 DSP alone.

The novel GSVD-based controller is an extension of the existing single-array IMC algorithm to two arrays and does not explicitly consider constraints, but benefits from a computationally efficient controller structure that requires 3 to 4 matrix-vector multiplications only, as opposed to more than 20 for MPC. In addition, the GSVD facilitates the interpretation of the MIMO dynamics, which are represented by decoupled SISO and TISO systems in generalised modal space. For larger control problems, controllers can be particularly difficult to tune, which is exacerbated by the ill-conditioned ORMs. The (generalised) modal decomposition allows one to concentrate the control effort in certain output directions, which was also used for tuning the MPC algorithm.

By considering structural symmetries of the orbit response matrix, it was shown that the computation time of both the GSVD-based and the MPC algorithm could be reduced by a factor of 10, which would suffice for the MPC implementation to meet the Diamond-II specification. However, the order of slow and fast correctors at Diamond-II will be irregular, breaking the symmetry of the two-array CD system and prohibiting the application of (frequency-independent) structural decompositions. A solution was proposed to recover the structural symmetry using a structural approximation, but the resulting large approximation error would require one to reduce the controller bandwidth and therefore also the performance in terms of disturbance attenuation.

To conclude, the following paragraphs address future research directions that are motivated by the results and conclusions of this thesis.

Structural approximations. The analysis of Chapter 3 focuses on CD systems that have ORMs with approximate structural symmetries, but assumes that each actuator has the same dynamics. As opposed to the Frobenius norm approximation [28], [106], the optimisation-based approaches from Chapter 3 can be extended to consider asymmetry of the actuator dynamics, yielding SDPs with frequency-dependent LMIs that could be formulated on a frequency-by-frequency basis. Such an extension would also allow multi-array CD systems to be analysed.

GSVD-based control: anti-windup scheme. Analogous to the existing single-array controller design, the ill-conditioned ORM is accommodated using a (static) output compensator. Since the Diamond-II FOFB will use slow and fast correctors with magnetic fields of different strengths, the GSVD-based controller for Diamond-II could benefit from anti-windup schemes for both slow and fast correctors. Future research could base an anti-windup scheme on an approach developed for single-array CD systems with slew-rate constraints [50] and extend it to amplitude constraints and two-array CD systems.

GSVD-based control: input and output compensators. The input and output compensators accommodate the non-orthonormal output transformation matrix that produces a performance difference between original and generalised modal space. For a different number of slow and fast actuators, the combination of input and output compensators may cause an overshoot of the output sensitivity for frequencies at which the control effort transitions from slow to fast actuator arrays. To remedy this problem, future research could consider frequency-dependent compensators or defining different output compensators for slow and fast actuator arrays.

Dykstra's method. The results from Chapter 6 show that combining Dykstra's method with a fast gradient scheme can significantly reduce the computation time of MPC for CD systems with input amplitude and slew-rate constraints. While the convergence properties of the fast gradient method under inaccurate projection are characterised in the existing literature [114], it has been shown that Dykstra's method can stall [13], which prohibits from characterising the convergence properties of the combined methods. A convergence analysis is particularly important for practical analysis and future research could focus on solving stalling of Dykstra's method, e.g. by using methods from distributed optimisation [99].

References

[1] M. G. Abbott, *FA-archiver*, Nov. 2021. https://github.com/Araneidae/fa-archiver.

[2] M. G. Abbott *et al.*, "Diamond-II technical design report," Diamond Light Source, Didcot, UK, Tech. Rep., Aug. 2022. https://www.diamond.ac.uk/Home/News/LatestNews/2022/14-10-22.html.

[3] M. G. Abbott, G. Rehm, and I. S. Uzun, "Architecture of transverse multi-bunch processor feedback at Diamond," in *Proc. Int. Conf. Accel. Large Exp. Phys. Contr. Syst. (ICALEPCS)*, Melbourne, Australia, Dec. 2015, pp. 298–301.

[4] C. Abraham *et al.*, "Diamond-II conceptual design report," Diamond Light Source, Didcot, UK, Tech. Rep., May 2019.

[5] M. Ali *et al.*, "Level-3 BLAS on the TI C6678 multi-core DSP," in *Proc. Int. Symp. Comput. Architect. High Perform. Comput.*, Oct. 2012, pp. 179–186.

[6] B. J. Allison and A. J. Isaksson, "Design and performance of mid-ranging controllers," *J. Proc. Contr.*, vol. 8, no. 5, pp. 469–474, Oct. 1998.

[7] O. Alter and H. Golub, "Reconstructing the pathways of a cellular system from genome-scale signals by using matrix and tensor computations," *Proc. Nat. Acad. Sci. USA*, vol. 102, no. 49, pp. 17 559–17 564, Dec. 2005.

[8] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.* 2019. http://docs.mosek.com/9.0/toolbox/index.html.

[9] R. Bartolini *et al.*, "Analysis of beam orbit stability and ground vibrations at the Diamond storage ring," in *Proc. Eur. Part. Accel. Conf. (EPAC)*, Genoa, Italy, Jun. 2008, pp. 1980–1982.

[10] R. Bartolini *et al.*, "Double-double bend achromat cell upgrade at the Diamond Light Source: From design to commissioning," *Phys. Rev. Accel. Beams*, vol. 21, p. 050 701, 5 May 2018.

[11] H. H. Bauschke and P. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Space*, 1st ed. Berlin, Germany: Springer, 2011, p. 468.

[12] H. H. Bauschke and V. Koch, "Projection methods: Swiss army knives for solving feasibility and best approximation problems with halfspaces," *ArXiv e-prints*, Jan. 2013.

[13] H. H. Bauschke *et al.*, "On Dykstra's algorithm: Finite convergence, stalling, and the method of alternating projections," *Optim. Lett.*, vol. 14, no. 8, pp. 1975–1987, May 2020.

[14] D. J. Bender and R. A. Fowell, "Computing the estimator-controller form of a compensator," *Int. J. Contr.*, vol. 41, no. 6, pp. 1565–1575, Sep. 1985.

[15] D. M. Bishop, *Group Theory and Chemistry*, 1st ed. Oxford, UK: Clarendon, 1973.

[16] M. Boge *et al.*, "Fast closed orbit control in the SLS storage ring," in *Proc. Part. Accel. Conf. (PAC)*, New York City, NY, Mar. 1999, pp. 1129–1131.

[17] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*, 1st ed. Cambridge, UK: Cambridge Univ. Press, 2017.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*, 1st ed. Cambridge, UK: Cambridge Univ. Press, 2004.

[19] S. Boyd *et al.*, *Linear Matrix Inequalities in System and Control Theory*, 1st ed. Philadelphia, PA: SIAM, 1994.

[20] S. Boyd *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[21] J. P. Boyle and R. L. Dykstra, "A method for finding projections onto the intersection of convex sets in Hilbert spaces," in *Proc. Symp. Adv. Order Restricted Statist. Inference*, Iowa City, IA, Sep. 1986, pp. 28–47.

[22] R. D. Braatz *et al.*, "Identification and cross-directional control of coating processes: Theory and experiments," in *Proc. Amer. Contr. Conf. (ACC)*, Chicago, IL, Jun. 1992, pp. 1556–1561.

[23] T. F. Chan, "An optimal circulant preconditioner for Toeplitz systems," *SIAM J. Sci. Statist. Comput.*, vol. 9, no. 4, pp. 766–771, Jul. 1988.

[24] A. W. Chao *et al.*, *Handbook of Accelerator Physics and Engineering*, 2nd ed. Singapore: World Scientific, 2013.

[25] C. Christou and V. C. Kempson, "Operation of the Diamond Light Source injector," in *Proc. Part. Accel. Conf. (PAC)*, Albuquerque, NM, Jun. 2007, pp. 1112–1114.

[26] M. T. Chu, E. Funderlic, and G. H. Golub, "On a variational formulation of the generalized singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 18, no. 4, pp. 1082–1092, Oct. 1997.

[27] M. T. Chu and R. J. Plemmons, "Real-valued, low rank, circulant approximation," *SIAM J. Matrix Anal. Appl.*, vol. 24, no. 3, pp. 645–659, Jan. 2003.

[28] F. Chuang, C. Danielson, and F. Borrelli, "Robust approximate symmetric model predictive control," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Osaka, Japan, Dec. 2015, pp. 2400–2405.

[29] D. R. Curtiss, "Recent extentions of Decartes' rule of signs," *Ann. Math.*, vol. 19, no. 4, pp. 251–278, 1918.

[30] R. D'Andrea and G. E. Dullerud, "Distributed control design for spatially interconnected systems," *IEEE Trans. Automat. Contr.*, vol. 48, no. 9, pp. 1478–1495, Sep. 2003.

[31] C. R. Danielson, "Symmetric constrained optimal control: Theory, algorithms, and applications," PhD thesis, University of California, Berkeley, CA, 2014.

[32] ——, "An alternating direction method of multipliers algorithm for symmetric MPC," in *Proc. IFAC Nonlinear Model Predictive Contr. Conf. (NMPC)*, Madison, WI, Aug. 2018, pp. 319–324.

[33] P. J. Davis, *Circulant Matrices*, 1st ed. New York, NY: Wiley, 1979.

[34] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, Apr. 2000.

[35] M. C. de Oliveira, J. E. Camino, and R. E. Skelton, "A convexifying algorithm for the design of structured linear controllers," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Sidney, Australia, Dec. 2000, pp. 2781–2786.

[36] F. Deutsch and H. Hundal, "The rate of convergence of Dykstra's cyclic projections algorithm: The polyhedral case," *Numer. Functional Anal. Optim.*, vol. 15, no. 6, pp. 537–565, May 1994.

[37] S. R. Duncan, "The design of robust cross-directional control systems for paper making," in *Proc. Amer. Contr. Conf. (ACC)*, Seattle, WA, Jun. 1995, pp. 1800–1805.

[38] ——, "The design of a fast orbit beam stabilisation system for the Diamond synchrotron," University of Oxford, Oxford, UK, Tech. Rep. 2296/07, Aug. 2007.

[39] S. R. Duncan, J. M. Allwood, and S. S. Garimella, "The analysis and design of spatial control systems in strip metal rolling," *IEEE Trans. Contr. Syst. Technol.*, vol. 6, no. 2, pp. 220–232, 1998.

[40] S. R. Duncan and W. Heath, "The robustness of multi-array cross-directional control systems," in *Proc. Contr. Sys. Conf.*, Stockholm, Sweden, Sep. 2010, pp. 180–185.

[41] S. R. Duncan, W. Heath, and A. Taylor, "A new approach to multi-array cross-directional control," in *Proc. Contr. Sys. PanPacific Conf.*, Vancouver, BC, Jun. 2008, pp. 69–74.

[42] D. Einfeld, "ALBA synchrotron light source commissioning," in *Proc. Int. Part. Accel. Conf. (IPAC)*, San Sebastián, Spain, Jul. 2011, pp. 1–5.

[43] L. El Ghaoui *et al.*, *Advances in Linear Matrix Inequality Methods in Control*, 1st ed. Philadelphia, PA: SIAM, 2000.

[44] D. G. Feingold and R. S. Varga, "Block diagonally dominant matrices and generalizations of the Gerschgorin circle theorem," *Pac. J. Math.*, vol. 12, no. 4, pp. 1241–1250, Apr. 1962.

[45] C. E. Garcia and M. Morari, "Internal model control. 2. Design procedure for multivariable systems," *Ind. Eng. Chem. Process Des. Dev.*, vol. 24, no. 2, pp. 472–484, Apr. 1985.

[46] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice – a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989.

[47] S. Gayadeen, "Synchrotron electron beam control," DPhil Thesis, University of Oxford, Oxford, UK, 2014.

[48] S. Gayadeen and S. R. Duncan, "Uncertainty modeling and robust stability analysis of a synchrotron electron beam stabilisation control system," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Maui, HI, USA, Dec. 2012, pp. 931–936.

[49] ——, "Design of an electron beam stabilisation controller for a synchrotron," *Contr. Eng. Pract.*, vol. 26, pp. 201–210, May 2014.

[50] ——, "Discrete-time anti-windup compensation for synchrotron electron beam controllers with rate constrained actuators," *Automatica*, vol. 67, pp. 224–232, May 2016.

[51] S. Gayadeen, S. R. Duncan, and W. P. Heath, "Design of multi-array controllers for electron beam stabilisation on synchrotrons," in *Proc. Amer. Contr. Conf. (ACC)*, Washington, DC, Jun. 2013, pp. 1201–1206.

[52] S. Gayadeen, M. Furseman, and G. Rehm, "A procedure for the characterization of corrector magnets," in *Proc. Int. Beam Instrum. Conf. (IBIC)*, Barcelona, Spain, Sep. 2016, pp. 728–731.

[53] S. Gayadeen and W. Heath, "An internal model control approach to mid-ranging control," in *Proc. IFAC Symp. Adv. Contr. Chem. Process*, Istanbul, Turkey, Jul. 2009, pp. 542–547.

[54] S. Gayadeen, S. R. Duncan, and G. Rehm, "Optimal control of perturbed static systems for synchrotron electron beam stabilisation," *IFAC PapersOnLine*, vol. 50, no. 1, pp. 9967–9972, 2017, 20th IFAC World Congress.

[55] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. Baltimore, MD: The Johns Hopkins Univ. Press, 2013.

[56] G. A. Gravvanis, "Solving symmetric arrowhead and special tridiagonal linear systems by fast approximate inverse preconditioning," *J. Math. Model Algorithms*, vol. 1, no. 4, pp. 269–282, Dec. 2002.

[57] G. H. Hardy, J. E. Littlewood, and G. Pólya, *Inequalities*, 1st ed. Cambridge, UK: Cambridge Univ. Press, 1934.

[58] A. Harrison. (Jul. 2022). Diamond receives funding confirmation for the first phase of Diamond-II. Accessed on 11.01.2023, Diamond Light Source, https://www.diamond.ac.uk/Home/News/LatestNews/2022/15-06-22.html.

[59] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis," *UCLA Work. Pap. Phon.*, vol. 16, pp. 1–84, Dec. 1970.

[60] ——, "Parafac2: Mathematical and technical notes," *UCLA Work. Pap. Phon.*, vol. 22, pp. 30–44, Dec. 1972.

[61] W. Heath, "Orthogonal functions for cross-directional control of web forming processes," *Automatica*, vol. 32, no. 2, pp. 183–198, Feb. 1996.

[62] W. Heath and A. Wills, "Design of cross-directional controllers with optimal steady state performance," *Eur. J. Contr.*, vol. 10, no. 1, pp. 15–27, Dec. 2004.

[63] E. Heaven *et al.*, "Recent advances in cross machine profile control," *IEEE Contr. Syst. Mag.*, vol. 14, no. 5, pp. 35–46, Oct. 1994.

[64] M. Herceg *et al.*, "Multi-parametric toolbox 3.0," in *Proc. Eur. Contr. Conf. (ECC)*, Zürich, Switzerland, Jul. 2013, pp. 502–510. http://control.ee.ethz.ch/~mpt.

[65] M. T. Heron *et al.*, "Performance and future development of the Diamond fast orbit feedback system," in *Proc. Eur. Part. Accel. Conf. (EPAC)*, Genoa, Italy, Jun. 2008, pp. 3257–3259.

[66] M. T. Heron *et al.*, "Diamond Light Source electron beam position feedback: Design, realization and performance," in *Proc. Int. Conf. Accel. Large Exp. Phys. Contr. Syst. (ICALEPCS)*, Kobe, Japan, Oct. 2009, pp. 650–652.

[67] M. Heron *et al.*, "Feed-forward and feedback schemes applied to the Diamond Light Source storage ring," in *Proc. Int. Part. Accel. Conf. (IPAC)*, Dresden, Germany, Jul. 2014, pp. 1757–1759.

[68] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge, UK: Cambridge Univ. Press, 2013.

[69] N. Hubert *et al.*, "Global orbit feedback systems down to DC using fast and slow correctors," in *Proc. Eur. Workshop Beam Diagn. Instrum. Part. Accel. (DIPAC)*, Basel, Switzerland, Jan. 2009, pp. 27–31.

[70] S.-H. Hur, R. Katebi, and A. Taylor, "Modeling and control of a plastic film manufacturing web process," *IEEE Trans. Ind. Inform.*, vol. 7, no. 2, pp. 171–178, May 2011.

[71] S. Hyung and S. Lall, "Optimal decentralized control of linear systems via Groebner bases and variable elimination," in *Proc. Amer. Contr. Conf. (ACC)*, Baltimore, MD, Jun. 2010, pp. 5608–5613.

[72] J. L. Jerez *et al.*, "Embedded online optimization for model predictive control at Megahertz rates," *IEEE Trans. Automat. Contr.*, vol. 59, no. 12, pp. 3238–3251, Aug. 2014.

[73] B. Kågström, "The generalized singular value decomposition and the general $(A - \lambda B)$-problem," *BIT*, vol. 24, pp. 568–583, Dec. 1984.

[74] T. Kato, *Perturbation Theory for Linear Operators*, 2nd ed. Berlin, Germany: Springer, 1980.

[75] I. Kempf, Sep. 2019. https://github.com/kmpape/box_rate_projection.

[76] ——, Sep. 2019. https://github.com/kmpape/fast_gradient_method.

[77] ——, Oct. 2019. https://github.com/kmpape/Fast-IPC-TIC6678.

[78] ——, Jul. 2021. https://github.com/kmpape/fofb-amc540.

[79] ——, Jul. 2021. https://github.com/kmpape/fofb-amc540-codegen.

[80] ——, Nov. 2022. https://github.com/kmpape/fofb-amc540-startup.

[81] ——, Dec. 2022. https://github.com/kmpape/fofb-amc540-hil.

[82] ——, May 2022. https://github.com/kmpape/HO-GSVD.

[83] I. Kempf, P. J. Goulart, and S. R. Duncan, "Alternating direction method of multipliers for block circulant model predictive control," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Nice, France, Dec. 2019, pp. 4311–4316.

[84] ——, "Fast gradient method for model predictive control with input rate and amplitude constraints," in *Proc. IFAC World Congr.*, Berlin, Germany, Jul. 2020, pp. 6542–6547.

[85] ——, "A higher-order generalized singular value decomposition for rank deficient matrices," *SIAM J. Matrix Anal. Appl.*, 2023, to appear.

[86] I. Kempf, *Diamond-II fast orbit feedback: Controller design report*, Apr. 2023. https://github.com/kmpape/DII_controller_design.

[87] I. Kempf, P. Goulart, and S. Duncan, *Control of cross-directional systems with approximate symmetries*, Jun. 2023. arXiv: 2306.17565 [eess.SY].

[88] I. Kempf *et al.*, "Multi-array electron beam stabilization using block-circulant transformation and generalized singular value decomposition," in *Proc. IEEE Conf. Decis. Contr. (CDC)*, Jeju Island, Republic of Korea, Dec. 2020, pp. 3431–3436.

[89] I. Kempf *et al.*, "Model predictive control for electron beam stabilization in a synchrotron," in *Proc. Eur. Contr. Conf. (ECC)*, London, UK, Jul. 2022, pp. 814–819.

[90] I. Kempf *et al.*, "Symmetry exploitation in orbit feedback systems of synchrotrons for computational efficiency," *IEEE Trans. Nucl. Sci.*, vol. 68, no. 3, pp. 258–269, Mar. 2021.

[91] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Toronto, ON, Tech. Rep., Apr. 2009.

[92] K.Van Deun *et al.*, "Identifying common and distinctive processes underlying multiset data," *Chemom. Intell. Lab. Syst.*, vol. 129, pp. 40–51, Nov. 2013.

[93] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: An overview of methods, challenges, and prospects," *Proc. IEEE*, vol. 103, no. 9, pp. 1449–1477, Sep. 2015.

[94] S. Lang, *Algebra*, 3rd ed. New York, NY: Springer, 2002.

[95] P. D. Lax, *Linear Algebra and its Applications*, 2nd ed. New York, NY: Wiley, 2007.

[96] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1999.

[97] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognit.*, vol. 44, no. 7, pp. 1540–1551, Jul. 2011.

[98] J. M. Maciejowski, *Multivariable Feedback Design*, 1st ed. Reading, MA: Addison-Wesley, 1989.

[99] K. Margellos *et al.*, "Distributed constrained optimization and consensus in uncertain networks via proximal minimization," *IEEE Trans. Automat. Contr.*, vol. 63, no. 5, pp. 1372–1387, 2018.

[100] I. P. S. Martin *et al.*, "Active optics stabilisation measures at the Diamond storage ring," in *Proc. Int. Part. Accel. Conf. (IPAC)*, Dresden, Germany, Jul. 2014, pp. 1760–1762.

[101] ——, "Orbit stability studies for the Diamond-II storage ring," in *Proc. Int. Part. Accel. Conf. (IPAC)*, Bangkok, Thailand, Jul. 2022, pp. 2602–2605.

[102] I. P. S. Martin *et al.*, "Operating the Diamond storage ring with reduced vertical emittance," in *Proc. Int. Part. Accel. Conf. (IPAC)*, Shanghai, China, Jun. 2013, pp. 249–251.

[103] M. Martini, "An introduction to transverse beam dynamics in accelerators," CERN, Geneva, Switzerland, Tech. Rep. CERN-PS-96-11-PA, Mar. 1996.

[104] I. McIerney, E. C. Kerrigan, and G. A. Constantinides, "Horizon-independent preconditioner design for linear predictive control," *IEEE Trans. Automat. Contr.*, vol. 68, no. 1, pp. 580–587, Jan. 2023.

[105] S. H. Mirza *et al.*, "Closed orbit correction at synchrotrons for symmetric and near-symmetric lattices," *Phys. Rev. Accel. Beams*, vol. 22, no. 7, p. 072 804, Jul. 2019.

[106] S. H. Mirza *et al.*, "Performance of closed orbit feedback systems with spatial model mismatch," *Phys. Rev. Accel. Beams*, vol. 23, no. 7, p. 072 801, Jul. 2020.

[107] M. Morari and E. Zafiriou, *Robust Process Control*, 1st ed. New Jersey, NJ: Prentice-Hall, 1989.

[108] A. F. D. Morgan, M. G. Abbott, and G. Rehm, "First experiences with the longitudinal feedback system at Diamond Light Source," in *Proc. Int. Part. Accel. Conf. (IPAC)*, Copenhagen, Denmark, May 2017, pp. 1992–1995.

[109] Motorola, "MVME5500 series VME," January, Tech. Rep. MVME5500 Series, 2003.

[110] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, 1st ed. Berlin, Germany: Springer, 2003.

[111] N. S. Nise, *Control Systems Engineering*, 7th ed. New York, NY: Wiley, 2011.

[112] A. Olmos *et al.*, "Commissioning of the ALBA fast orbit feedback system," in *Proc. Int. Beam Instrum. Conf. (IBIC)*, Monterey, CA, Sep. 2014.

[113] C. C. Paige and M. A. Saunders, "Towards a generalized singular value decomposition," *SIAM J. Numer. Anal.*, vol. 18, no. 3, pp. 398–405, 1981.

[114] A. Patrascu and I. Necoara, "On the convergence of inexact projection primal first-order methods for convex minimization," *IEEE Trans. Automat. Contr.*, vol. 63, no. 10, pp. 3317–3329, Oct. 2018.

[115] E. Plouviez and F. Uberto, "The orbit correction scheme of the new EBS of the ESRF," in *Proc. Int. Beam Instrum. Conf. (IBIC)*, Barcelona, Spain, Sep. 2016, pp. 694–697.

[116] E. Plouviez *et al.*, "The new fast orbit correction system of the ESRF storage ring," in *Proc. Eur. Workshop Beam Diagn. Instrum. Part. Accel. (DIPAC)*, Hamburg, Germany, May 2011, pp. 215–217.

[117] E. Plouviez *et al.*, "Optimisation of the SVD treatment in the fast orbit correction of the ESRF storage ring," in *Proc. Int. Beam Instrum. Conf. (IBIC)*, Oxford, UK, Sep. 2013, pp. 215–217.

[118] S. P. Ponnapalli, "Higher-order generalized singular value decomposition: Comparative mathematical framework with applications to genomic signal processing," PhD thesis, University of Texas at Austin, Austin, TX, USA, 2010.

[119] S. P. Ponnapalli *et al.*, "A higher-order generalized singular value decomposition for comparison of global mRNA expression from multiple organisms," *PLOS ONE*, vol. 6, no. 12, pp. 1–11, Dec. 2011.

[120] G. Rehm, "Achieving and measuring sub-micrometer beam stability at 3rd generation light sources," *J. Phys. Conf. Ser.*, vol. 425, no. 4, p. 042 001, Mar. 2013.

[121] G. Rehm, "Characterisation of closed orbit feedback systems," in *Proc. Int. Beam Instrum. Conf. (IBIC)*, Malmö, Sweden, Sep. 2019, pp. 479–485.

[122] S. Richter, C. N. Jones, and M. Morari, "Computational complexity certification for real-time MPC with input constraints based on the fast gradient method," *IEEE Trans. Automat. Contr.*, vol. 57, no. 6, pp. 1391–1403, Jun. 2012.

[123] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control," *IEEE Trans. Automat. Contr.*, vol. 51, no. 2, pp. 274–286, Feb. 2006.

[124] J. Rowland *et al.*, "Status of the Diamond fast orbit feedback system," in *Proc. Int. Conf. Accel. Large Exp. Phys. Contr. Syst. (ICALEPCS)*, Knoxville, TN, Oct. 2007, pp. 535–537.

[125] A. K. Saibaba, J. Hart, and B. van Bloemen Waanders, "Randomized algorithms for generalized singular value decomposition with application to sensitivity analysis," *Numer. Linear Algebra Appl.*, vol. 28, no. 4, e2364, Feb. 2021.

[126] M. Sands, "The physics of electron storage rings: An introduction," Stanford Linear Accelerator Center, Menlo Park, CA, Tech. Rep. SLAC-121, May 1979.

[127] S. Shin, "New era of synchrotron radiation: Fourth-generation storage ring," *AAPPS Bull.*, vol. 31, no. 1, pp. 21–36, Aug. 2021.

[128] D. Simon, *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*, 1st ed. New York, NY: Wiley, 2006.

[129] S. Skogestad, M. Morari, and J. C. Doyle, "Robust control of ill-conditioned plants: High-purity distillation," *IEEE Trans. Automat. Contr.*, vol. 33, no. 12, pp. 1092–1105, Dec. 1988.

[130] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 2nd ed. New York, NY: Wiley, 2005.

[131] C. Steier *et al.*, "Operational experience integrating slow and fast orbit feedbacks at the ALS," in *Proc. Eur. Part. Accel. Conf. (EPAC)*, Lucerne, Switzerland, Jul. 2004, pp. 2786–2788.

[132] B. Stellato *et al.*, "OSQP: An operator splitting solver for quadratic programs," *Math. Program. Comput.*, vol. 12, no. 4, pp. 637–672, Feb. 2020.

[133] G. W. Stewart, "On the sensitivity of the eigenvalue problem $Ax = \lambda Bx$," *SIAM J. Numer. Anal.*, vol. 9, no. 4, pp. 669–686, Dec. 1972.

[134] B. Suksiri and M. Fukumoto, "An efficient framework for estimating the direction of multiple sound sources using higher-order generalized singular value decomposition," *Sensors*, vol. 19, no. 13, Jul. 2019.

[135] Y. R. Tan *et al.*, "Commissioning of the fast orbit feedback system at the Australian synchrotron," in *Proc. Int. Part. Accel. Conf. (IPAC)*, Copenhagen, Denmark, May 2017, pp. 1770–1773.

[136] Texas Instruments, *TMS320C6000 chip support library API reference guide*, SPRU401J, Aug. 2004.

[137] ——, *CPU and instruction set*, SPRUGH7, Nov. 2010.

[138] ——, *DSP cache*, SPRUGY8, Nov. 2010.

[139] ——, *General purpose input/output (GPIO)*, SPRUGV1, Nov. 2010.

[140] ——, *Optimizing loops on the C66x DSP*, SPRABG7, Nov. 2010.

[141] ——, *TMS320C6000 assembly language tools v8.2.x*, SPRUI03B, May 2010.

[142] ——, *SYS/BIOS inter-processor communication (IPC) 1.25*, SPRUGO6E, Sep. 2012.

[143] ——, *TMS320C6000 optimizing compiler v7.4*, SPRU187U, Jul. 2012.

[144] ——, *Peripheral component interconnect express (PCIe)*, SPRUGS6D, Sep. 2013.

[145] ——, *Multicore fixed and floating-point digital signal processor*, SPRS691E, Mar. 2014.

[146] ——, *Multicore navigator*, SPRUGR9H, Apr. 2015.

[147] ——, *Throughput performance guide for KeyStone II devices*, SPRABK5B, Dec. 2015.

[148] ——, *PROCESSOR-SDK-C667X*, v06.03.00.106, Apr. 2020.

[149] Y. Tian and L. H. Yu, "NSLS-II fast orbit feeback with individual eigenmode compensation," in *Proc. Part. Accel. Conf. (PAC)*, New York, NY, Apr. 2011, pp. 1488–1490.

[150] R. J. Tibshirani, "Dykstra's algorithm, ADMM, and coordinate descent: Connections, insights, and extensions," in *Adv. Neural Inf. Process. Syst. (NeurIPS 2017)*, vol. 30, 2017, pp. 517–528.

[151] I. S. Uzun and M. T. Heron, "Fast orbit feedback – communication controller specification and design," Diamond Light Source, Didcot, UK, Tech. Rep. CTRL-FOFB-CC-0001, Jan. 2009.

[152] VadaTech, *Xilinx Virtex-7 FPGA AMC with dual TI DSP (AMC540)*, 4FM737-12, 2019.

[153] J. G. Van Antwerp *et al.*, "Cross-directional control of sheet and film processes," *Automatica*, vol. 43, no. 2, pp. 191–211, Feb. 2007.

[154] P. Van den Hof, "Closed-loop issues in system identification," *Annu. Rev. Contr.*, vol. 22, pp. 173–186, 1998.

[155] C. F. Van Loan, "Generalizing the singular value decomposition," *SIAM J. Numer. Anal.*, vol. 13, no. 1, pp. 76–83, Mar. 1976.

[156] ——, "Computing the CS and the generalized singular value decompositions," *Numer. Math.*, vol. 49, no. 4, pp. 479–491, 1985.

[157] ——, (Jun. 2015). Lecture 6. The higher-order generalized singular value decomposition, CIME-EMS Summer School, http://www.dm.unibo.it/%5C%7Esimoncin/CIME/Vanloan_Lec6.pdf.

[158] J. G. VanAntwerp and R. D. Braatz, "A tutorial on linear and bilinear matrix inequalities," *J. Proc. Contr.*, vol. 10, no. 4, pp. 363–385, Aug. 2000.

[159] L. J. Van't Veer *et al.*, "Gene expression profiling predicts clinical outcome of breast cancer," *Nature*, vol. 415, no. 6871, pp. 530–536, Jan. 2002.

[160] N. Viswanadham and K. D. Minto, "Robust observer design with application to fault detection," in *Proc. Amer. Contr. Conf. (ACC)*, Atlanta, GA, Jun. 1988, pp. 1393–1399.

[161] J. Von Neumann, *Functional Operators: The Geometry of Orthogonal Spaces*, Bd. 2. Princeton Univ. Press, 1951.

[162] R. Walker, "Commissioning and status of the Diamond storage ring," in *Proc. Asian Part. Accel. Conf. (APAC)*, Indore, India, Jan. 2007.

[163] J. R. Weaver, "Centrosymmetric (cross-symmetric) matrices, their basic properties, eigenvalues, and eigenvectors," *Amer. Math. Mon.*, vol. 92, no. 10, pp. 711–717, Dec. 1985.

[164] H. Wiedemann, *Particle Accelerator Physics*, 4th ed. Berlin, Germany: Springer, 2007.

[165] E. J. N. Wilson, *An Introduction to Particle Accelerators*, 1st ed. Oxford, UK: Oxford Univ. Press, 2001.

[166] L. H. Yu *et al.*, "The performance of a fast closed orbit feedback system with combined fast and slow correctors," in *Proc. Eur. Part. Accel. Conf. (EPAC)*, Genoa, Italy, Jun. 2008.

[167] L. Yu *et al.*, "Real-time harmonic closed orbit correction," *Nucl. Instrum. Methods Phys. Res. A*, vol. 284, no. 2, pp. 268–285, Dec. 1989.

[168] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, 1st ed. New York, NY: Prentice-Hall, 1996.