UNIVERSITY OF CALIFORNIA, SAN DIEGO

Optimal Control Techniques for Resistive Wall Modes in Tokamaks

A Dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Engineering Sciences (Engineering Physics)

by

Mitchell Dobbs Pearson Clement

Committee in charge:

Professor George Tynan, Chair
Professor Scott Baden
Professor Farhat Beg
Professor Robert Bitmead
Professor Gerald Navratil
Professor Clifford Surko

2017

The Dissertation of Mitchell Dobbs Pearson Clement is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____

_____

Chair

University of California, San Diego

2017

DEDICATION

To those that allow the imagination free reign.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my advisors George Tynan, for helping me find this project, and Gerald Navratil for supporting my research. I would especially like to thank Jeremy Hanson for all of his help and insight, as well as the rest of the DIII-D team.

Chapters 1, 5, 6, 7 and 8 and Appendix B contain material which has been accepted for publication in *Control Engineering Practice*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. The dissertation author was the primary investigator and author of this paper.

Chapters 3, 6, 7 and 8 contain material which has been submitted for publication in *Nuclear Fusion*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Atomic Energy Agency, 2017. The dissertation author was the primary investigator and author of this paper.

DIII-D data shown in this dissertation can be obtained in digital format by following the links at https://fusion.gat.com/global/D3D_DMP.

This work was supported by the U.S. Department of Energy (DOE) under grants DE-FC02-04ER54698 and DE-FG02-04ER54761.

VITA

2006	B.S., Mechanical Engineering, United States Naval Academy
2006-2011	Submarine Officer, United States Navy
2011-2012	Engineer II, Control Systems, Honeywell Aerospace and Defense
2014	M.S., Engineering Sciences (Engineering Physics), University of California, San Diego
2014	Teaching Assistant, University of California, San Diego
2014-2017	Data Acquisition Technician, Columbia University
2017	Ph.D., Engineering Sciences (Engineering Physics), University of California, San Diego

PUBLICATIONS

M. D. Clement, J. M. Hanson, J. Bialek, and G. A. Navratil, "$\mathscr{H}_2$ optimal control techniques for resistive wall mode feedback in tokamaks", manuscript in preparation.

M. Clement, J. Hanson, J. Bialek, and G. Navratil, "GPU-based optimal control for RWM feedback in tokamaks", *Control Engineering Practice*, 68:15 - 22, 2017.

ABSTRACT OF THE DISSERTATION

Optimal Control Techniques for Resistive Wall Modes in Tokamaks

by

Mitchell Dobbs Pearson Clement

Doctor of Philosophy in Engineering Sciences (Engineering Physics)

University of California, San Diego, 2017

Professor George Tynan, Chair

Tokamaks can excite kink modes that can lock or nearly lock to the vacuum vessel wall, and whose rotation frequencies and growth rates vary in time but are generally inversely proportional to the magnetic flux diffusion time of the vacuum vessel wall. This magnetohydrodynamic (MHD) instability is pressure limiting in tokamaks and is called the Resistive Wall Mode (RWM). Future tokamaks that are expected to operate as fusion reactors will be required to maximize plasma pressure in order to maximize fusion performance. The DIII-D tokamak is equipped with electromagnetic control coils, both inside and outside of its vacuum vessel, which create magnetic fields that are small by

comparison to the machine's equilibrium field but are able to dynamically counteract the RWM. Presently for RWM feedback, DIII-D uses its interior control coils using a classical proportional gain only controller to achieve high plasma pressure. Future advanced tokamak designs will not likely have the luxury of interior control coils and a proportional gain algorithm is not expected to be effective with external control coils. The computer code VALEN was designed to calculate the performance of an MHD feedback control system in an arbitrary geometry. VALEN models the perturbed magnetic field from a single MHD instability and its interaction with surrounding conducting structures using a finite element approach. A linear quadratic gaussian (LQG) control, or $\mathcal{H}_2$ optimal control, algorithm based on the VALEN model for RWM feedback was developed for use with DIII-D's external control coil set. The algorithm is implemented on a platform that combines a graphics processing unit (GPU) for real-time control computation with low latency digital input/output control hardware and operates in parallel with the DIII-D Plasma Control System (PCS). Simulations and experiments showed that modern control techniques performed better, using 77% less current, than classical techniques when using coils external to the vacuum vessel for RWM feedback. RWM feedback based on VALEN outperformed a classical control algorithm using external coils to suppress the normalized plasma response to a rotating $n = 1$ perturbation applied by internal coils over a range of frequencies. This study describes the design, development and testing of the GPU based control hardware and algorithm along with its performance during experiment and simulation.

Chapter 1

Resistive Wall Modes

To maximize fusion power gain, tokamaks designed to operate as steady state reactors will need to operate at high plasma pressure. Unfortunately, high plasma pressure also brings with it magnetohydrodynamic (MHD) instabilities, most concerning of which is the external kink mode slowed down to a timescale commensurate with the flux diffusion time of a nearby conducting vacuum vessel wall. Normally the ideal kink mode has a growth rate on the order of an Alfvén time, but the electromagnetic interaction of the perturbed plasma field with the vessel wall can induce large eddy currents that in turn slow the mode's growth. This form of the external kink mode is known as the Resistive Wall Mode (RWM). The stability of a tokamak plasma in the presence of such modes may be studied using the equations of ideal MHD.

1.1 Ideal Magnetohydrodynamics

One of the simplest models for plasma treats it as a single species, infinitely conducting fluid and is known as the ideal MHD model. The full derivation from the

two-fluid equations can be found in a number of textbooks[1][3]. The ideal MHD model is expressed by the following coupled equations:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{v}) = 0 \tag{1.1a}$$

$$\rho(\frac{\partial}{\partial t} + \vec{v} \cdot \vec{\nabla})\vec{v} = \vec{j} \times \vec{B} - \vec{\nabla} p, \tag{1.1b}$$

which describe the mass and momentum conservation within a plasma, where ρ is the mass density, $\vec{v}$ is the bulk fluid velocity, p is the pressure, $\vec{j}$ is the current density and $\vec{B}$ is the magnetic field. From electrodynamics, Ohm's law relates the current density to the fluid velocity, magnetic field and electric field $\vec{E}$, by[4]:

$$(\vec{E} + \vec{v} \times \vec{B})\sigma = \vec{j}$$

where σ is the plasma conductivity, and may also be defined as the inverse of resistivity, $\eta = \frac{1}{\sigma}$. Ideal MHD is concerned with the limit when the plasma is infinitely conductive, and Ohm's law may be written,

$$\vec{E} + \vec{v} \times \vec{B} = 0. \tag{1.2}$$

The spatial and time evolution of the electric and magnetic fields are given by Faraday's and Ampere's laws from Maxwell's equations[4],

$$\vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{1.3a}$$

$$\vec{\nabla} \times \vec{B} = \mu_0 \vec{j} \tag{1.3b}$$

where the low-frequency limit in 1.3b has been assumed in order to drop the displacement current that normally appears in Ampere's law. The last equation used to couple the model is the equation of state; assuming the ratio of specific heats to be $\gamma = \frac{5}{3}$, the equation of state is given as

$$\frac{p}{\rho^{\gamma}} = const. \tag{1.4}$$

The ideal MHD model is primarily concerned with the low-frequency, long-wavelength magnetic behavior of the plasma and can be used to analyze the stability of various plasma equilibria so long as some assumptions are valid. These assumptions include: i) the plasma is charge-neutral; ii) the fluid velocity is much less then the speed of light; $|\vec{v}| \ll c$; iii) the plasma is collisional, resulting in parallel density and pressure gradients that allows the assumption $T_e = T_i$ to hold; and iv) the time scale is long compared to the electron and ion cyclotron periods. Even though tokamak plasmas are essentially collisionless, the MHD model is still adequate to describe the plasma's equilibrium.

1.1.1 MHD Parameters

An important relationship to describe a plasma in the MHD context is the ratio of hydrodynamic pressure to magnetic pressure. This property is called β and is defined as

$$\beta = \frac{2\mu_0 p}{B^2}. \tag{1.5}$$

In tokamaks, a more often used variant of β is normalized β[5], β_N, and is defined as:

$$\beta_N = \frac{\beta}{I_p/(r_0 B_T)}, \tag{1.6}$$

where I_p is the plasma current in MAmps, r_0 is the plasma minor radius in meters and B_T is the toroidal magnetic field in Tesla. To maximize fusion power gain in tokamaks, high-β plasma is desirable. The scaling law for volumetric fusion power density (P_f/V_P) in tokamaks is given as[6]:

$$\frac{P_f}{V_P} \propto \beta^2 B_T^4, \tag{1.7}$$

where B_T is the toroidal magnetic field.

1.1.2 Ideal MHD Equilibrium in Tokamaks

In the steady state limit when fluid velocity and time derivatives are zero, $\frac{\partial}{\partial t} = \vec{v} = 0$, 1.1a-1.4 reduce to

$$\vec{j} \times \vec{B} = \vec{\nabla} p \tag{1.8a}$$

$$\vec{\nabla} \times \vec{B} = \mu_0 \vec{j} \tag{1.8b}$$

$$\vec{\nabla} \cdot \vec{B} = 0. \tag{1.8c}$$

Taking the dot product of $\vec{B}$ with 1.8a gives $\vec{B} \cdot \vec{\nabla} p = 0$. Similarly, taking the dot product of $\vec{j}$ with 1.8a gives $\vec{j} \cdot \vec{\nabla} p = 0$. These relationships imply that $\vec{B}$ and $\vec{j}$ are each orthogonal to $\vec{\nabla} p$ and the lines representing these vectors reside on contours of constant pressure, as shown in figure 1.1. The overall field becomes helical because of the poloidal magnetic field generated by the plasma current and the toroidal field created by the toroidal field coils. A measure of the "twistedness" of the field is known as the safety factor, q, and is the ratio of poloidal mode number and toroidal mode number, $q = m/n$. Edge safety factor relates the number of times a field line transits the toroidal direction to the number of times it transits the poloidal direction on a given flux surface. When m and n are integer numbers and q is

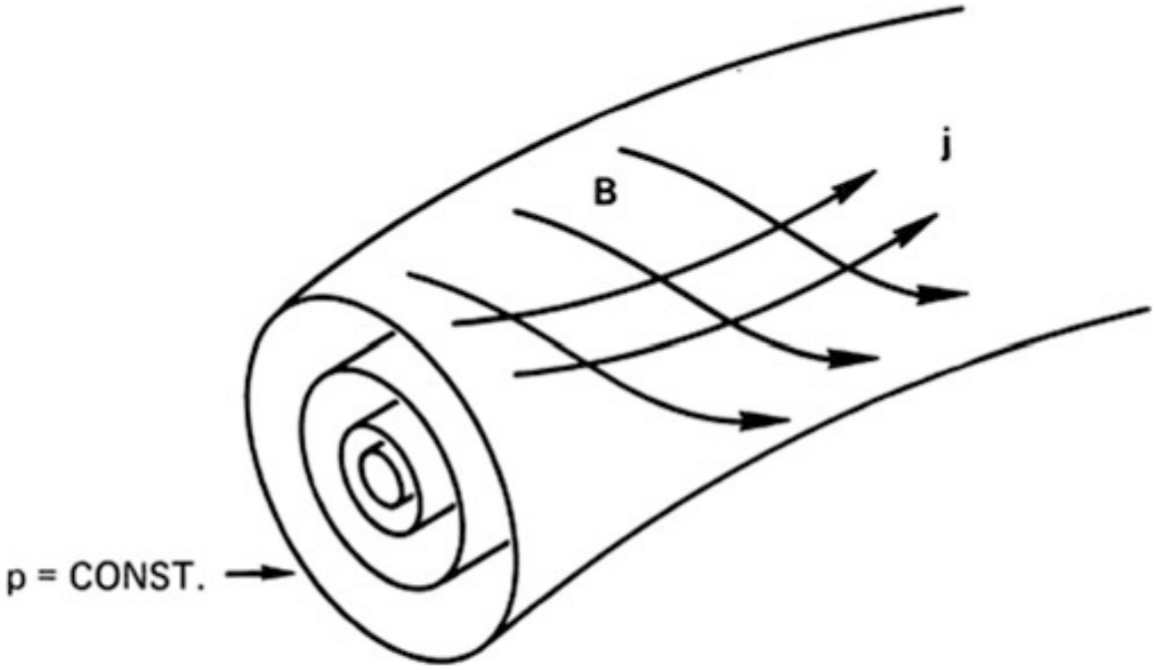

Figure 1.1: In a toroidal configuration, both $\vec{j}$ and $\vec{B}$ lie on constant-pressure, nested flux surfaces[1].

rational, the safety factor at these surfaces becomes important for stability properties of that equilibrium. Safety factors below 1 are not stable configurations for tokamaks.

1.1.3 Equilibrium Stability

Stability is an important issue when analyzing a system's equilibrium, where the important question being whether or not the system will remain at equilibrium in the presence of small perturbations. The MHD energy principle was first investigated by Bernstein[7], using a linearized version of the ideal MHD model (1.1a-1.4) and analyzing small displacements away from equilibrium.

1.1.3.1 Initial Value Formulation

It is assumed that all parameters have an equilibrium value (0 subscript) and a perturbed value (1 subscript). All quantities are linearized about the background state: $X(\vec{r},t) = X_0(\vec{r}) + \bar{X}_1(\vec{r},t)$ with $\bar{X}_1/X_0 \ll 1$, where $\vec{r}$ is a position vector for the coordinate system . All quantities are expressed in terms of a small displacement vector about the position $\vec{r}$, $\vec{\xi}(\vec{r})$, where $\vec{v}_1 = \frac{\partial\vec{\xi}}{\partial t}$ or $\vec{\xi} = \int \vec{v}_1 dt$. The aim of this analysis is to express all perturbed quantities in terms of $\vec{\xi}$ and then obtain a single equation describing the time evolution of $\vec{\xi}$[3]. Initial conditions for this analysis are:

$$\vec{\xi}(\vec{r},0) = \vec{B}_1(\vec{r},0) = \bar{\rho}_1(\vec{r},0) = \bar{p}_1(\vec{r},0) = 0$$
$$\frac{\partial\vec{\xi}(\vec{r},0)}{\partial t} \equiv \vec{v}_1(\vec{r},0) \neq 0.$$

Ohm's law (equation 1.2) and Faraday's law (equation 1.3a) may be combined to give the induction equation:

$$\frac{\partial\vec{B}}{\partial t} = \nabla \times (\vec{v} \times \vec{B}). \tag{1.10}$$

The adiabatic relation given in equation 1.4 implies the spatial and temporal derivatives of density and pressure:

$$\frac{\vec{\nabla}P}{P} = \gamma\frac{\vec{\nabla}\rho}{\rho}, \frac{1}{P}\frac{\partial P}{\partial t} = \frac{\gamma}{\rho}\frac{\partial\rho}{\partial t}, \tag{1.11}$$

which may be combined with the continuity equation (equation 1.1a) to give an equation to describe the pressure evolution in time:

$$\frac{\partial P}{\partial t} + \vec{v}\cdot\vec{\nabla}P + \gamma P\vec{\nabla}\cdot\vec{v} = 0. \tag{1.12}$$

Equations 1.1a, 1.12 and 1.10 can be integrated with respect to time to give perturbed quantities as functions of the equilibrium quantity and the displacement, $\vec{\xi}$:

$$\bar{\rho}_1 = -\vec{\nabla}\cdot(\rho_0\vec{\xi}) \tag{1.13a}$$

$$\bar{p}_1 = -\vec{\xi}\cdot\vec{\nabla}p_0 - \gamma p_0\vec{\nabla}\cdot\vec{\xi} \tag{1.13b}$$

$$\vec{\bar{Q}} \equiv \vec{\bar{B}}_1 = \vec{\nabla}\times(\vec{\xi}\times\vec{B}_0). \tag{1.13c}$$

These perturbed quantities are then inserted into equation 1.8a to give the MHD force operator:

$$\mathbf{F}(\vec{\xi}) = \rho\frac{\partial^2\vec{\xi}}{\partial t^2} = \vec{j}\times\vec{\bar{B}}_1 + \vec{\bar{j}}_1\times\vec{B} - \vec{\nabla}\bar{p}_1 \tag{1.14}$$

where the 0 subscripts have been dropped but still denote equilibrium quantities. Using this formulation for a given perturbation, one may compute the time evolution of a specific perturbation. However, if the only concern is whether or not a given configuration is stable or not to perturbations, a better formulation exists and is described in the next two sections.

1.1.3.2 Variational Formulation

Equation 1.14 may be more explicitly written as:

$$\mathbf{F}(\vec{\xi}) = \frac{1}{\mu_0}(\nabla\times\vec{B})\times\vec{\bar{Q}} + \frac{1}{\mu_0}(\nabla\times\vec{\bar{Q}})\times\vec{B} + \nabla(\vec{\xi}\cdot\nabla p + \gamma p\nabla\cdot\vec{\xi}). \tag{1.15}$$

Next, it is assumed that all perturbed quantities vary as $X_1 = \Re(X_1 e^{-i\omega t})$, so that time derivatives may be converted as $\frac{\partial}{\partial t} \rightarrow \omega$ and equation 1.14 can be written:

$$-\omega^2\rho\vec{\xi} = \mathbf{F}(\vec{\xi}). \tag{1.16}$$

1.1.3.3 MHD Energy Principle

Taking equation 1.16 and multiplying by $\dot{\vec{\xi}}$, then integrating over the plasma volume and finally integrating with respect to time, gives the change in system potential energy associated with an arbitrary displacement from $\vec{r}$, to $\vec{r}+\vec{\xi}(\vec{r})$:

$$\delta W = -\frac{1}{2}\int \vec{\xi}\cdot\mathbf{F}(\vec{\xi})d\vec{r}. \tag{1.17}$$

An equilibrium will be stable if $\delta W \geq 0$, otherwise it is unstable to a perturbation. The calculation of equation 1.17 may be done either with a vacuum surrounding the plasma, with a conducting wall surrounding the plasma or some combination of the two, as is the case in tokamaks. More generally, the δW of equation 1.17 may be decomposed like so:

$$\delta W = \delta W_F + \delta W_S + \delta W_V, \tag{1.18}$$

where the subscripts F, S and V correspond to the fluid, surface and vacuum contributions. These components are expressed as:

$$\delta W_F = \frac{1}{2}\int_F \left[\frac{\vec{\bar{Q}}}{\mu_0} - \vec{\xi}_\perp\cdot(\vec{j}\times\vec{B}) + \gamma p|\vec{\nabla}\cdot\vec{\xi}|^2 + (\vec{\xi}_\perp\cdot\vec{\nabla}p)\nabla\cdot\vec{\xi}_\perp\right]d\vec{r} \tag{1.19a}$$

$$\delta W_S = \frac{1}{2}\int_S |\hat{n}\cdot\vec{\xi}_\perp|^2\hat{n}\cdot\left\|\nabla\left(p+\frac{B^2}{2\mu_0}\right)\right\|dS \tag{1.19b}$$

$$\delta W_V = \frac{1}{2}\int_V \frac{|\hat{B}_1|^2}{\mu_0}d\vec{r}, \tag{1.19c}$$

where $\vec{\xi}_\perp = \vec{\xi} - \vec{\xi}\cdot\hat{b}_0$ is the displacement in the direction perpendicular to the unit vector of the equilibrium magnetic field, $\hat{b}_0$, $\hat{n}$ is the normal vector to the plasma surface and $\|\|$ denotes the jump from vacuum to plasma. A measure of the local curvature of the

equilibrium magnetic field, $\vec{\kappa}$, is defined as

$$\vec{\kappa} = \hat{b}_0 \cdot \nabla \hat{b}_0 \tag{1.20}$$

and it can be shown that situations in which $\vec{\kappa}$ is parallel to the gradient in equilibrium pressure, $\vec{\nabla} p_0$, that is when $\vec{\kappa} \cdot \vec{\nabla} p_0 > 0$, leads to negative values of δW_F, which is destabilizing. These modes are called pressure-driven instabilities and are of concern in tokamaks in which β is to be maximized.

1.2 External Kink Modes

An external mode is one in which the equilibrium position of the plasma-vacuum interface moves, and is sometimes called a free-boundary mode. Also, these perturbations normally have an associated rational flux surface (see section 1.1.2) in the vacuum region between the plasma and vessel wall. As shown in figure 1.2, what is called the "kink" instability can develop in a cylindrical column of plasma with a toroidal surface current and toroidal and poloidal magnetic fields. Qualitatively, any perturbation to the plasma equilibrium will deform the helical equilibrium field lines, causing the field to become stronger in the concave region and weaker in the convex region. Normally, the strong toroidal field is enough to stabilize this instability, but if the pressure profile gradient is steep enough (high β) or the current profile severe enough, the kink will become unstable to perturbations and grow. The external kink mode is often the limiting instability in tokamak plasma and the time scale for its growth is on the order of an Alfvén time, $\tau_A = r_0 \sqrt{\rho \mu_0} / B_T$, where r_0 is the machine minor radius, ρ is the plasma mass density, and B_T the toroidal magnetic field. In tokamaks like DIII-D, τ_A is on the order of microseconds.

It has been shown that in tokamaks and similar toroidal devices, a safety factor greater than unity is required for kink stability, $q = \frac{m}{n} = \frac{aB_P}{RB_T} > 1$[8][9].

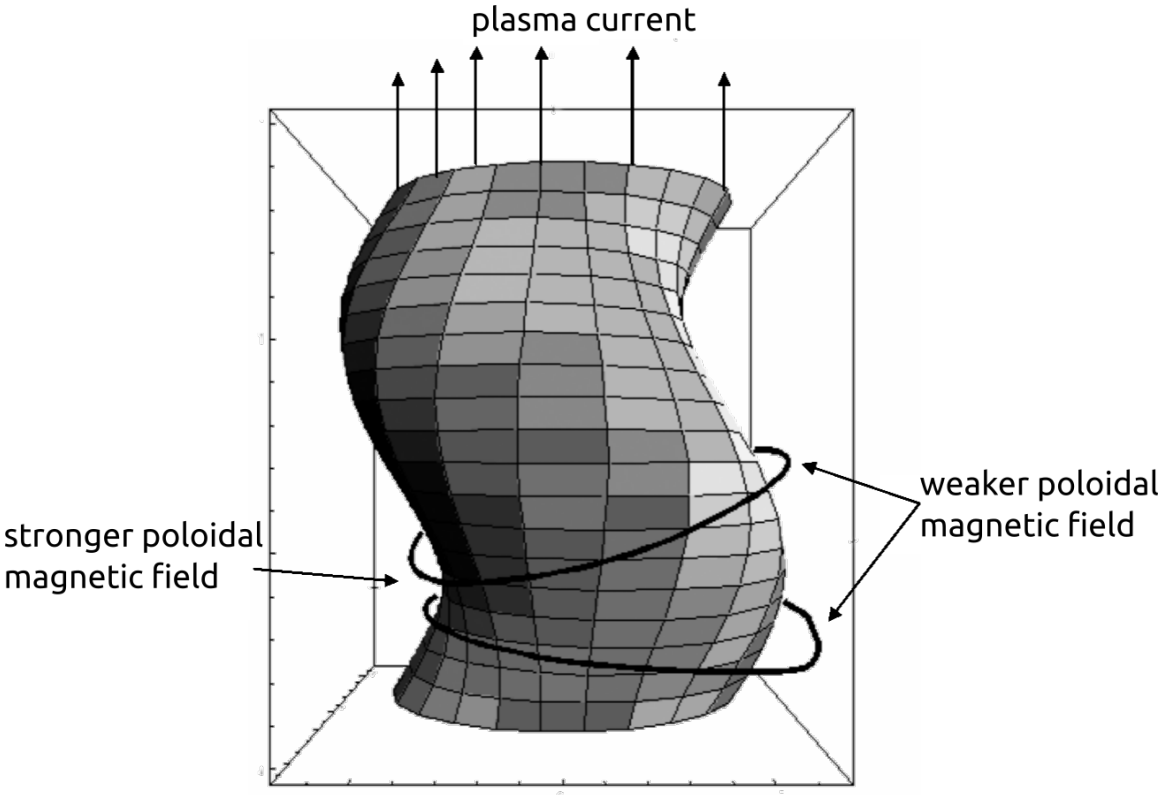

Figure 1.2: External kink instability can occur in helical equilibrium magnetic fields[2].

1.3 The Resistive Wall Mode

In most fusion experiments, the plasma is contained in a somewhat close fitting vacuum vessel with some finite conductivity. Any external mode that perturbs the plasma-vacuum interface will cause a perturbation in magnetic flux through the nearby conducting wall, and in the wall, currents will be generated by Lenz's law[4] to oppose this flux. These wall eddy currents have a stabilizing effect on the instability but owing to the wall's resistivity, the currents will decay on the order of the flux diffusion time of the wall,

$\tau_w = L_w/R_w$. Here L_w and R_w are the inductance and resistance of the wall, respectively, and τ_w is often referred to simply as the "wall time." This temporary stabilization reduces the growth time scale of the external kink mode from microseconds (τ_A) to milliseconds (τ_w). This special case of the external kink mode is called the Resistive Wall Mode (RWM). Only walls that are impenetrable to changes in flux, i.e. superconducting walls, can stabilize the RWM[10]. This phenomenon implies a maximum theoretical plasma pressure that may be reached by a tokamak ($\beta_{\text{ideal-wall}}$), which is higher than if the plasma is surrounded purely by vacuum ($\beta_{\text{no-wall}}$).

Developing a superconducting vacuum vessel is not very practical, but fortunately other methods exist for stabilizing the RWM. One method is via plasma rotation, where the electromagnetic torque generated between the wall and the plasma stabilizes the mode[11]. Simulations show that the RWM may or may not be fully suppressed by rotation alone in future tokamak devices operating with high performance plasma conditions[12][13][14][15]. The other method is by providing a stabilizing flux around the plasma using electromagnetic coils positioned near the plasma-vacuum interface. These control coils may be either internal or external to the vacuum vessel wall. Coils placed outside of the vessel, though protected from the intense heat of the plasma and nuclear environment, will see any imparted flux on the RWM delayed by the vessel wall. Future Advanced Tokamaks (AT) built to demonstrate burning plasma conditions will likely not have control coils internal to the vacuum vessel. Simulations and modeling predict that classical control techniques such as PID control, used with external coils on the proposed ITER tokamak may be ineffective at stabilizing the RWM or may use excessive current in doing so[16]. For active feedback, a variety of control schemes have been investigated to counteract the RWM, including classical control and state-space methods[17]. A Lin-

ear Quadratic optimal controller for RWM feedback on ITER using the in-vessel control coils has also been proposed[18]. Other fusion experiments have had success in controlling the RWM using largely classical algorithms[19][20], but it is worth noting that these machines have very long flux diffusion times for their walls compared to other large scale tokamaks and are not able to reach plasma parameters relevant to ATs. Two Reverse Field Pinch (RFP) experiments, which are similar in configuration to tokamaks, have reported using system identification techniques to develop state space models for RWM feedback[21]. Other tokamaks have reported success in stabilizing the RWM via active control methods[22]. Simulation of controllers based on lowerer dimensional RWM models have yielded promising resluts for applications on the DIII-D tokamak[23][24]. Numerical simulations of feedback using Linear Quadratic Gaussian (LQG) control with external coils and the VALEN RWM model of the ITER tokamak have also yielded promising results[16]. The second method of stabilizing the RWM will be extensively explored in this dissertation since it allows the problem of stability to be solved by feedback control, or, more specifically, model-based optimal control.

This chapter contains material which has been accepted for publication in *Control Engineering Practice*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. The dissertation author was the primary investigator and author of this paper.

Chapter 2

Linear Systems: State-Space Methods and Model Reduction

This chapter is intended to provide the minimum basic knowledge of state-space linear systems needed to comprehend the subsequent chapters. This chapter also covers model reduction techniques that reduce the dimensions of linear systems while nominally maintaining the transfer function of the overall system. Classical control typically involves the analysis of dynamic linear systems through some transform from the time domain to the frequency domain. Such methods will not be covered here.

2.1 State-space Linear Systems

A system governed by linear dynamics in continuous-time (CT) may be characterized by the following matrix-vector ordinary differential equation and matrix-vector

equation:

$$\frac{d}{dt}\vec{x}(t) = \mathbf{A}(t)\vec{x}(t) + \mathbf{B}(t)\vec{u}(t) \tag{2.1a}$$

$$\vec{y}(t) = \mathbf{C}(t)\vec{x}(t) + \mathbf{D}(t)\vec{u}(t), \tag{2.1b}$$

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are the system, input, output and feed-forward matrices, respectively, and $\vec{y}$, $\vec{u}$ and $\vec{x}$ are the measurement, control and state vectors, respectively. For cases covered in the sections to follow, the feed-forward matrix $\mathbf{D}$ will always be zero and all other matrices and vectors will be real valued. When all the matrices of equations 2.1a and 2.1b do not vary with time, the system is said to be linear time-invariant (LTI). When time is, however, discrete (as it is with digital control systems), this system has the discrete-time (DT) representation characterized by matrix-vector difference equations:

$$\vec{x}_{k+1} = \mathbf{F}\vec{x}_k + \mathbf{G}\vec{u}_k \tag{2.2a}$$

$$\vec{y}_k = \mathbf{C}\vec{x}_k, \tag{2.2b}$$

where k is the sample number and the matrices $\mathbf{F}$ and $\mathbf{G}$ are derived from the CT matrices $\mathbf{A}$ and $\mathbf{B}$ like so:

$$\mathbf{F} = e^{\mathbf{A}\Delta t} \tag{2.3a}$$

$$\mathbf{G} = \mathbf{F}(\mathbf{I} - e^{-\mathbf{A}\Delta t})\mathbf{A}^{-1}\mathbf{B}, \tag{2.3b}$$

where Δt is the sample time and $\mathbf{I}$ is the identity matrix.

2.1.1 System Stability

In the absence of a control input, the stability of the system described by equations 2.1a and 2.2a will be governed by the eigenvalues of the system matrix $\mathbf{A}$ (CT) or $\mathbf{F}$ (DT). For the CT case, if any eigenvalue of $\mathbf{A}$ has a positive real part, or in the DT case, if any eigenvalue of $\mathbf{F}$ is outside the unit circle in the complex plane, the system is unstable and will grow unbounded in time.

2.1.2 Controllability and Observability

The solution to equation 2.1a is given by[25]:

$$\vec{x}(t) = \Phi(t,t_0)\vec{x_0} + \int_{t_0}^{t} \Phi(t,\tau)\mathbf{B}(\tau)u(\tau)d\tau \tag{2.4}$$

where Φ is the system's state transition matrix and is defined as $\Phi(t,t_0) = e^{\mathbf{A}(t-t_0)}$. For discrete time cases, the solution to 2.2a is

$$\vec{x}(t) = \Phi(t,t_0)\vec{x_0} + \sum_{\tau=t_0}^{t-1} \Phi(t,\tau+1)\mathbf{G}(\tau)u(\tau) \tag{2.5}$$

A system, or more specifically the pair ($\mathbf{A}(t)$, $\mathbf{B}(t)$), is said to be *controllable* if there exists some control function $u(t)$, depending on t_0 and $\vec{x_0}$, and defined on some finite closed interval $[t_0, t_1]$, that is able to transfer the state from some arbitrary initial state, $\vec{x_0}$, to the origin ($\vec{x_1} = 0$)[25]. A test for controllability is given by whether or not the controllability gramian, $\mathbf{W}_c$,

$$\mathbf{W}_c(t_0,t_1) = \int_{t_0}^{t_1} \Phi(t_0,\tau)\mathbf{B}(\tau)\mathbf{B}^T(\tau)\Phi^T(t_0,\tau)d\tau \tag{2.6}$$

is positive definite for some $t_1 > t_0$[25]. For discrete time systems $\mathbf{W}_c$ is calculated as

$$\mathbf{W}_c(t_0,t_1) = \sum_{\tau=t_0}^{t_1-1} \Phi(t_0,\tau+1)\mathbf{G}(\tau)\mathbf{G}^T(\tau)\Phi^T(t_0,\tau+1). \tag{2.7}$$

A system or the pair $(\mathbf{A}(t), \mathbf{C}(t))$ is said to be *observable* if the initial-state vector $\vec{x_0} = \vec{x}(t_0)$, can be found from $u(t)$ and $y(t)$ measured over some finite closed interval $[t_0,t_1]$[26]. Controllability and observability are related concepts and similarly a test for observability includes determining the positive definiteness of the observability gramian $\mathbf{W}_o$,

$$\mathbf{W}_o(t_0,t_1) = \int_{t_0}^{t_1} \Phi^T(\tau,t_0)\mathbf{C}^T(\tau)\mathbf{C}(\tau)\Phi(\tau,t_0)d\tau. \tag{2.8}$$

For discrete time systems, $\mathbf{W}_o$ is calculated as

$$\mathbf{W}_o(t_0,t_1) = \sum_{\tau=t_0}^{t_1-1} \Phi^T(\tau,t_0)\mathbf{C}^T(\tau)\mathbf{C}(\tau)\Phi(\tau,t_0). \tag{2.9}$$

The gramians for controllability and observability are not defined for systems with unstable eigenvalues.

2.2 Model Reduction

Often large dimension linear systems will have many states that have poor controllability and observability. For the sake of real-time computation, these states need not be retained. Model reduction methods are the means by which higher dimensional state-space models are reduced in dimension, while still preserving the overall input/output relationship of the system.

2.2.1 Balanced Realization

State-space realizations given in 2.1a and 2.2a are not unique. Consider the invertible matrix $\mathbf{T}$ which can be used to transform a system's state $\vec{x}$ by $\vec{x_T} = \mathbf{T}^{-1}\vec{x}$. This transformation can be applied to any state-space system expressed by 2.1a and 2.1b as

$$\frac{d}{dt}\vec{x_T} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}\vec{x_T}(t) + \mathbf{T}^{-1}\mathbf{B}\vec{u}(t) \tag{2.10a}$$

$$\vec{y}(t) = \mathbf{C}\mathbf{T}\vec{x_T}(t). \tag{2.10b}$$

An important realization of such system is one in which the states are reordered by decreasing controllability and observability, and the system's observability and controllability gramians are equal and diagonal[27][28]. A transformation matrix must be found that will realize this reordering. To do so, first the Cholesky factorization of both gramians are computed, $\mathbf{W}_c = \mathbf{G}_c\mathbf{G}_c^H$ and $\mathbf{W}_o = \mathbf{G}_o\mathbf{G}_o^H$. Next, a new matrix is defined as $\mathbf{H} = \mathbf{G}_c^H\mathbf{G}_o$, and taking its Singular Value Decomposition gives $\mathbf{H} = \mathbf{U}\Sigma\mathbf{V}^H$. The matrix Σ will be diagonal and contain the Hankel Singular Values (HSVs) of the system, $\sigma_1, .., \sigma_n$, with $\sigma_1 \geq ... \geq \sigma_n \geq 0$. The required transformation matrix $\mathbf{T}$ is then given as $\mathbf{T} = \mathbf{G}_c\mathbf{U}\Sigma^{-\frac{1}{2}}$. This new balanced system now has the relation, $\mathbf{W}_c = \mathbf{W}_o = \Sigma$, and the system can be divided into subsystems as

$$\frac{d}{dt}\begin{bmatrix}\vec{x_1}\\ \vec{x_2}\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{bmatrix}\begin{bmatrix}\vec{x_1}\\ \vec{x_2}\end{bmatrix} + \begin{bmatrix}\mathbf{B}_1\\ \mathbf{B}_2\end{bmatrix}\vec{u},\ \ \Sigma = \begin{bmatrix}\Sigma_1 & 0\\ 0 & \Sigma_2\end{bmatrix} \tag{2.11a}$$

$$\vec{y} = \begin{bmatrix}\mathbf{C}_1 & \mathbf{C}_2\end{bmatrix}\begin{bmatrix}\vec{x_1}\\ \vec{x_2}\end{bmatrix} + \mathbf{D}\vec{u}, \tag{2.11b}$$

where $\mathbf{A}_{11}$ and Σ_1 are $r \times r \ \ (r < n)$ matrices.

2.2.2 Singular Perturbation Approximation

With the balanced realization given in 2.11a, states beyond the dimension r may be disregarded, as they have limited controllability and observability and contribute little to the overall input-output relationship of the system. A simple method would be to truncate the system at r states, which implies that $\sigma_r \gg \sigma_{r+1}$. The Singular Perturbation Approximation[29] is used to better approximate the low frequency input-output response of the full system. The first step of the approximation requires setting $\frac{d\vec{x_2}}{dt} = 0$ in 2.11a and then solving for $\vec{x_2}$ and reinserting that into the $\vec{x_1}$ subsystem. The resulting reduced order model is given by

$$\begin{aligned} \frac{d}{dt}\vec{x_1} &= \bar{\mathbf{A}}\vec{x_1} + \bar{\mathbf{B}}\vec{u} \\ \vec{y} &= \bar{\mathbf{C}}\vec{x_1} + \bar{\mathbf{D}}\vec{u}, \end{aligned} \tag{2.12}$$

where

$$\begin{aligned} \bar{\mathbf{A}} &= \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21} \\ \bar{\mathbf{B}} &= \mathbf{B}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{B}_2 \\ \bar{\mathbf{C}} &= \mathbf{C}_1 - \mathbf{C}_2\mathbf{A}_{22}^{-1}\mathbf{A}_{21} \\ \bar{\mathbf{D}} &= \mathbf{D} - \mathbf{C}_2\mathbf{A}_{22}^{-1}\mathbf{B}_2. \end{aligned} \tag{2.13}$$

2.2.3 Model Reduction of Unstable Systems

The methods described previously may only be used on stable systems. For unstable systems, the unstable subsystem must be separated via a real Schur decomposition[30].

A reduction method is then applied to the stable subsystem and the unstable subsystem is reinserted into the model[31]. Using the Schur decomposition of $\mathbf{A}$, the matrices of 2.1a and 2.1b may be transformed as follows

$$\begin{aligned}\mathbf{A}_s &= \mathbf{V}^T\mathbf{A}\mathbf{V} = \begin{bmatrix}\mathbf{A}_- & \mathbf{A}_c\\ 0 & \mathbf{A}_+\end{bmatrix}\\ \mathbf{B}_s &= \mathbf{V}^T\mathbf{B} = \begin{bmatrix}\mathbf{B}_-\\ \mathbf{B}_+\end{bmatrix}\\ \mathbf{C}_s &= \mathbf{C}\mathbf{V} = \begin{bmatrix}\mathbf{C}_- & \mathbf{C}_+\end{bmatrix}.\end{aligned} \tag{2.14}$$

The transformation matrix $\mathbf{V}$ can be chosen so that $\mathbf{A}_- \in \mathbb{R}^{l\times l}$ has stable eigenvalues and $\mathbf{A}_+ \in \mathbb{R}^{(n-l)\times(n-l)}$ has unstable eigenvalues, where l is the number of stable eigenvalues of the full system. A decomposed state-space representation of the system may now be written as

$$\begin{aligned}\frac{d}{dt}\vec{x}_- &= \mathbf{A}_-\vec{x}_- + \mathbf{A}_c\vec{x}_+ - \mathbf{B}_-\vec{u}\\ \frac{d}{dt}\vec{x}_+ &= \mathbf{A}_+\vec{x}_+ + \mathbf{B}_+\vec{u}\\ \vec{y} &= \mathbf{C}_-\vec{x}_- + \mathbf{C}_+\vec{x}_+.\end{aligned} \tag{2.15}$$

The contribution of the unstable subsystem to the stable subsystem may be treated by including it with the input:

$$\begin{aligned}\frac{d}{dt}\vec{x}_- &= \mathbf{A}_-\vec{x}_- + \tilde{\mathbf{B}}\tilde{u}\\ \vec{y} &= \mathbf{C}_-\vec{x}_- + \tilde{\mathbf{D}}\tilde{u}\end{aligned} \tag{2.16}$$

with

$$\tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{B}_- & \mathbf{A}_c \end{bmatrix}, \quad \tilde{u} = \begin{bmatrix} \vec{u} \\ \vec{x}_+ \end{bmatrix}, \quad \tilde{\mathbf{D}} = \begin{bmatrix} 0 & \mathbf{C}_+ \end{bmatrix}.$$

The truncation techniques of section 2.2.2 may be applied to the system given by equation 2.16 to give a reduced order subsystem

$$\begin{aligned} \frac{d}{dt}\vec{x}_{-r} &= \mathbf{A}_{-r}\vec{x}_{-r} + \tilde{\mathbf{B}}_r\tilde{u}, \quad \tilde{\mathbf{B}}_r = \begin{bmatrix} \mathbf{B}_{-r} & \mathbf{A}_{cr} \end{bmatrix} \\ \vec{y} &= \mathbf{C}_{-r}\vec{x}_{-r} + \tilde{\mathbf{D}}\tilde{u}. \end{aligned} \tag{2.17}$$

Now the unstable subsystem is added back into the system to give the final reduced order model

$$\begin{aligned} \frac{d}{dt}\vec{x}_r &= \mathbf{A}_r\vec{x}_r + \mathbf{B}_r\vec{u} \\ \vec{y} &= \mathbf{C}_r\vec{x}_r, \end{aligned} \tag{2.18}$$

where

$$\mathbf{A}_r = \begin{bmatrix} \mathbf{A}_{-r} & \mathbf{A}_{cr} \\ 0 & \mathbf{A}_+ \end{bmatrix}, \qquad \vec{x}_r = \begin{bmatrix} \vec{x}_{-r} \\ \vec{x}_+ \end{bmatrix},$$

$$\mathbf{C}_r = \begin{bmatrix} \mathbf{C}_{-r} & \mathbf{C}_+ \end{bmatrix}, \qquad \mathbf{B}_r = \begin{bmatrix} \mathbf{B}_{-r} \\ \mathbf{B}_+ \end{bmatrix}.$$

The state-space model methods discussed in section 2.1 will be used in the next chapter to describe the RWM and its interaction with surrounding conducting structures, such as in a tokamak, as a LTI state-space system. The model reduction techniques described in section 2.2 will also be used to considerably reduce the dimension of this system.

Chapter 3

VALEN

This chapter gives a detailed explanation of the VALEN RWM model. In essence, VALEN models the RWM and the surrounding geometry of arbitrary conducting structures as a group of coupled circuit equations. This is accomplished by representing the perturbation of flux caused by the MHD mode as a current on the unperturbed plasma surface. VALEN requires an input file to define the geometry of conducting structures, a kinetic equilibrium fit (kinetic EFIT)[32] of the plasma conditions when the mode is active and two user-defined parameters, s and α, where s is the stability parameter and α is the torque parameter; values for these affect the growth and rotation rate of the modeled mode.

3.1 The VALEN Model

VALEN is a finite element computer code, written in FORTRAN, that models the RWM and its interaction with surrounding conducting structures[33]. The following sections describe how VALEN formulates the RWM as a current and how this current interacts

with other nearby currents either in the vacuum vessel wall, or in an electromagnetic control coil; this is the crux of VALEN.

3.1.1 Plasma Circuit Formulation of MHD Perturbations

Once the RWM is expressed as a plasma surface current, the interaction of that current with surrounding conducting structures is a matter of basic electrodynamics. The idea is to leverage the definition of magnetic flux,

$$\Phi_i \equiv \int f_i(\theta,\phi)\vec{B}\cdot d\vec{a} \equiv L_i I_i, \tag{3.1}$$

to correlate the total perturbed magnetic field $\vec{B}$ caused by the perturbation to a current $\vec{I_i}$ for the ith mode or perturbation modeled. The eigenfunctions $f_i(\theta,\phi)$ are defined later. To fully describe the external properties of this perturbation, the normal component of the perturbation at the surface of the unperturbed plasma must be defined. Using standard toroidal coordinates (r,θ,ϕ) and defining the surface of the plasma as $r = const.$, the normal component is then equal to $\mathscr{J}\vec{B}\cdot\vec{\nabla}r$, where $\mathscr{J}$ denotes the Jacobian of the coordinate system, $\mathscr{J} \equiv \frac{\partial\vec{x}}{\partial r}\cdot(\frac{\partial\vec{x}}{\partial\theta}\times\frac{\partial\vec{x}}{\partial\phi})$. In toroidal coordinates, $\mathscr{J}$ has units of length squared, $\vec{B}$ has units of field and $\vec{\nabla}r$ is a dimensionless unit vector pointing normal to the unperturbed plasma surface. $\mathscr{J}\vec{B}\cdot\vec{\nabla}r$ therefore has units of magnetic flux. This normal component can then be expressed as the linear sum of eigenmodes:

$$\mathscr{J}\vec{B}\cdot\vec{\nabla}r = \sum_{i=0}^{\infty}\Phi_i(t)f_i(\theta,\phi), \tag{3.2}$$

where the eigenfunctions $f_i(\theta,\phi)$ are a set of linearly independent basis functions on the plasma surface. The eigenfunctions are orthogonal and normalized such that $\int_0^{2\pi}\int_{-\pi}^{\pi} f_i f_j d\theta d\phi =$

δ_{ij}. The eigenfunctions can also be taken to form the weighted sum of an arbitrary displacement $\vec{\xi}$,

$$\vec{\xi}(\theta,\phi) = \sum_{i=0}^{\infty} a_i f_i(\theta,\phi), \tag{3.3}$$

and from Bernstein's[7] formulation for the MHD energy principle (section 1.1.3.3), δW may then be written as

$$\delta W = -\frac{1}{2}\sum_{i=0}^{\infty}\sum_{j=0}^{\infty} a_i a_j \int_0^a \int_0^{2\pi} \int_{-\pi}^{\pi} \vec{\xi}_i \cdot \mathbf{F}(\vec{\xi}_j) d\theta d\phi dr \tag{3.4a}$$

$$\delta W = -\frac{1}{2}\sum_{i=0}^{\infty} a_i^2 \omega_i^2, \tag{3.4b}$$

where $\mathbf{F}$ is the force operator and the equation of motion is written as an eigenvalue problem of the form,

$$-\omega_i^2 \rho \vec{\xi}_i = \mathbf{F}(\vec{\xi}_i). \tag{3.5}$$

δW gives the change in potential energy in the plasma volume and the surrounding vacuum. Using equation 3.2 and the orthogonality of the eigenfunctions, the flux Φ_i may be written as

$$\Phi_i = \int_0^{2\pi}\int_{-\pi}^{\pi} f_i(\theta,\phi)\vec{B}\cdot d\vec{a} = \int_0^{2\pi}\int_{-\pi}^{\pi} f_i(\theta,\phi)\vec{B}\cdot \mathscr{J}\vec{\nabla} r d\theta d\phi. \tag{3.6}$$

The next pillar of the VALEN model is that the perturbed magnetic field $\vec{B}$ outside of the plasma may be represented by a surface current distribution on a control surface just

outside the unperturbed plasma. This surface current $\vec{K}(\theta,\phi)$ is defined by the equations

$$\vec{B}\cdot\vec{\nabla}r = 0$$
$$\vec{\nabla}r\times\vec{B} = \mu_0\vec{K}.$$

This surface current is solenoidal ($\vec{\nabla}\cdot\vec{K}=0$), tangential to the control surface and is expressed as

$$\vec{K} = -\vec{\nabla}r\times\vec{\nabla}\kappa, \tag{3.7}$$

with

$$\kappa(\theta,\phi,t) = \sum_{i=0}^{\infty} I_i(t)g_i(\theta,\phi). \tag{3.8}$$

The weighting functions $g_i(\theta,\phi)$ are chosen to produce the field given by equation 3.2 in the absence of any other currents, and the functions are normalized by

$$\int_0^{2\pi}\int_{-\pi}^{\pi} f_i g_i d\theta d\phi = 1 \tag{3.9}$$

so that I_i will have units of Amperes. Now the perturbed flux and surface current given by the ith mode are related by an inductance, L_i, defined as

$$\Phi_i = L_i I_i. \tag{3.10}$$

The set of currents defined by the plasma perturbation, I_i^p, may be defined like so,

$$L_i I_i^p = \sum_j^{\infty}(\delta_{ij} + s_i\lambda_{ij})\Phi_j, \tag{3.11}$$

where s_i is the stability constant of the ith mode and is defined as $s_i = -\omega_i^2 L_i$ and ω_i^2 is from the δW calculation of equation 3.4b. λ_{ij} is obtained by normalizing the basis functions

$$\lambda_{ij} = (\int_0^{2\pi} \int_{-\pi}^{\pi} f_i g_j d\theta d\phi)^{-1}$$

and, based on the normalization given in equation 3.9, $\lambda_{ij} = 1$ for $i = j$.

MHD analysis of plasma and plasma modes states that the potential energy released by an instability is transferred to plasma kinetic energy. That is to say, plasma kinetic energy is a sink for the mode's potential energy. To represent this relationship, the current I_i associated with the ith mode is split into two terms: I_i^p, introduced in the previous section, and which may be thought of as the part of the current associated with the plasma's kinetic energy, and I_i^d, a current lying on a dissipative shell existing on the surface of the plasma and is associated with the potential energy of the instability. The total current is then expressed as the sum of the plasma perturbation current and the current on the dissipative shell,

$$I_i = I_i^p + I_i^d. \tag{3.12}$$

I^d is determined from Ohm's law and the flux rule[4] as

$$\frac{d\Phi_i}{dt} = -R_i I_i^d, \tag{3.13}$$

where R_i is the resistance associated with the dissipative shell. R_i is determined to give R_i/L_i a value that is commensurate with the growth rate of the ideal mode, and in the case of the ideal kink mode in a tokamak, this is on the order of μs.

Image currents induced in the vacuum vessel wall and currents in feedback control

coils play an important role in the feedback stabilization of MHD instabilities. The ith component of flux associated with an external field is expressed on the plasma surface as

$$\Phi_i^{\text{ext}} = \int_0^{2\pi} \int_{-\pi}^{\pi} f_i(\theta,\phi)\vec{B}_{\text{ext}} \cdot \mathscr{J}\vec{\nabla} r d\theta d\phi = \sum_{j=0}^{N} M_{ij} I_j^w, \tag{3.14}$$

where I_j^w are the currents in the finite elements of the wall and control coils and M_{ij} is a mutual inductance between the ith component and jth finite element, of which there are N total elements. The normal field on the plasma surface for each mode is then given by

$$\Phi_i = L_i I_i + \Phi_i^{\text{ext}} = L_i I_i + \sum_{j=0}^{N} M_{ij} I_j^w. \tag{3.15}$$

The flux through a finite element k of the wall may be written as

$$\Phi_k = \sum_{i=0}^{\infty} L_{ki} I_i + \sum_{j=0}^{N} M_{kj} I_j^w, \tag{3.16}$$

and the time evolution of flux through this finite element is given by Ohm's law as

$$\frac{d\Phi_k}{dt} = -\sum_{j=0}^{N} R_{kj} I_j^w + V_k(t), \tag{3.17}$$

where R_{kj} will only be finite if $k = j$, and $V_k(t)$ is the potential applied to the finite element k, meaning that $V_k(t)$ will only be finite if the element k models a control coil connected to a power supply.

3.1.2 Instantaneous Plasma Response Model

The linear response of a plasma maybe represented by a permeability matrix $\mathscr{P}_{ss'}(t)$. What's more, the plasma response may be considered instantaneous if the

permeability can be approximated as $\mathscr{P}_{ss'}(t) \propto \delta(t)$[34]. The flux, Φ_s, associated with a plasma surface current, I_s, can be represented as

$$\vec{\Phi}_s(t) = \sum_{s'} \mathscr{P}_{ss'} \vec{\Phi}^x_{s'}(t). \tag{3.18}$$

This flux caused by external currents may be represented by the current potential (stream function for the current in the dissipative shell[35]), $\kappa_d(\theta, \phi, t)$, defined in equation 3.8. The current potential is then used to express the energy required to perturb the plasma as

$$\delta W = \frac{1}{2} \int_0^{2\pi} \int_{-\pi}^{\pi} \kappa_d \vec{B} \cdot \mathscr{J} \vec{\nabla} r d\theta d\phi \tag{3.19}$$

and the toroidal torque exerted on the plasma as

$$\tau_\phi = \int_0^{2\pi} \int_{-\pi}^{\pi} \frac{\partial \kappa_d}{\partial \phi} \vec{B} \cdot \mathscr{J} \vec{\nabla} r d\theta d\phi. \tag{3.20}$$

In toroidally symmetric plasmas (as is normally the case in tokamaks) the eigenmodes of δW (equations 3.4b and 3.19) are twofold degenerate, where one is like a sine and the other a cosine, and displaced by one quarter of a toroidal period from one another[34]. The flux vector of equation 3.18 may be more explicitly expressed as $\vec{\Phi}_s(t) = [\Phi_s^{(\cos)}(t) \;\; \Phi_s^{(\sin)}(t)]^T$. It can be shown[34] that the permeability matrix may be expressed as

$$\mathscr{P}_{ss'} = -\frac{1}{s_s^2 + \alpha_s^2} \begin{bmatrix} s_s & \alpha_s \\ -\alpha_s & s_s \end{bmatrix} \frac{L_s}{L_{s'}} \int_0^{2\pi} \int_{-\pi}^{\pi} f_s(\theta, \phi) g_{s'}(\theta, \phi) d\theta d\phi, \tag{3.21}$$

where s_s is the stability constant defined in equation 3.11, L_s and $L_{s'}$ are the inductances defined by $L_s \equiv \frac{\Phi_s}{I_s}$, the basis function $f_s(\theta, \phi)$ and $g_{s'}(\theta, \phi)$ are defined in equations 3.2 and

3.8, respectively, and the new dimensionless parameter, α_s, is the torque parameter that quantifies the toroidal torque applied by a stationary eigenmode. The torque parameter has the form:

$$\alpha_s = \frac{\Omega_p - \Omega_s}{\gamma_v}, \tag{3.22}$$

where Ω_p is the toroidal rotation rate of the plasma, Ω_s is the toroidal rotation rate of the mode, and γ_v is an effective viscosity coefficient. The ratio of the torque to the energy associated with an eigenmode is $-\alpha_s s/s_s$. Calculations of the torque are possible but the optimal method for determining α_s may be from experimental observations[34].

3.1.3 Conducting Structures Circuit Model

With the plasma mode now modeled as a circuit, its interaction with the surrounding conducting structures of the tokamak may be studied. To do this, VALEN models the induced currents in 3-dimensional conducting structures as a series of coupled R-L circuit equations[33]. Fields and currents in the model are simplified to be quasi-static, meaning that magnetostatic rules apply and coupled circuit equations are valid and no displacement currents exist. Within the 3-dimensional structure, current density is approximated as

$$\vec{J}(\vec{r},t) = \sum_{j=0}^{N} I_j(t)\vec{w}_j(\vec{r}). \tag{3.23}$$

The weight functions $\vec{w}_j(\vec{r})$ are solenoidal, $\vec{\nabla} \cdot \vec{w}_j(\vec{r}) = 0$, and correspond to macroscopic loops of circulating current in each “mesh” element. The weight functions have units of inverse area and define a closed vector path. $I_j(t)$ has units of Amperes and is the current circulating within the closed path. VALEN assumes a thin shell approximation of conducting structures, meaning that fields and currents do not vary through the thickness

of the element.

VALEN's circuit equations are derived by substitution into the equation for magnetic and electric field expressed as potentials[4],

$$\vec{E} + \dot{\vec{A}} + \vec{\nabla}\phi = 0, \tag{3.24}$$

where $\vec{\nabla} \times \vec{A} = \vec{B}$. Each term in equation 3.24 may be expressed in terms of mesh current and weight functions from equation 3.23. Using Ohm's law, the electric field, $\vec{E}$, is expressed by

$$\vec{E} = \eta \vec{J} = \eta \sum_{j=0}^{N} I_j(t) \vec{w}_j(\vec{r}) \tag{3.25}$$

and the magnetic vector potential $\vec{A}$ is written using the Coulomb gauge, $\vec{\nabla} \cdot \vec{A} = 0$, as

$$\vec{A}(\vec{r}) = \frac{\mu_0}{4\pi} \int_{V'} \frac{\vec{J}(\vec{r}')}{|\vec{r} - \vec{r}'|} d^3 r' = \frac{\mu_0}{4\pi} \int_{V'} \frac{\sum_{j=0}^{N} I_j(t) \vec{w}_j(\vec{r}')}{|\vec{r} - \vec{r}'|} d^3 r'. \tag{3.26}$$

Multiplying equation 3.24 by $\vec{w}_i(\vec{r})$ and integrating over the volume of all elements gives

$$\int_V \vec{w}_i(\vec{r}) \cdot \left(\eta \sum_{j=0}^{N} I_j(t) \vec{w}_j(\vec{r}) + \frac{\mu_0}{4\pi} \int_{V'} \frac{\sum_{j=0}^{N} \dot{I}_j(t) \vec{w}_j(\vec{r}')}{|\vec{r} - \vec{r}'|} d^3 r' + \vec{\nabla}\phi \right) dr = 0. \tag{3.27}$$

To define the $\mathbf{R}$ and $\mathbf{L}$ matrices, $I(t)$ and $\dot{I}(t)$ are not dependent on the position vector $\vec{r}$ and so may be taken out of their respective integrands to give,

$$L_{ij} = \frac{\mu_0}{4\pi} \int_V \int_{V'} \frac{\vec{w}_i(\vec{r}) \cdot \vec{w}_j(\vec{r}')}{|\vec{r} - \vec{r}'|} d^3 r' d^3 r \tag{3.28a}$$

$$R_{ij} = \int_V \eta \vec{w}_i(\vec{r}) \cdot \vec{w}_j(\vec{r}) d^3 r, \tag{3.28b}$$

where equation 3.28a has a similar form to Nuemann's formula for inductance[4] and equation 3.28b is the definition of lumped resistance. If element i is connected to a power supply, then

$$V_i(t) = -\int_V \vec{w}_i(\vec{r}) \cdot \nabla\phi(t) d^3r. \tag{3.29}$$

Otherwise $V_i(t) = 0$. The coupled equations of 3.27 can be cast into a block matrix-vector equation of the form

$$\begin{bmatrix} L_{ww} & L_{wf} \\ L_{fw} & L_{ff} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} I_w \\ I_f \end{bmatrix} = -\begin{bmatrix} R_w & 0 \\ 0 & R_f \end{bmatrix} \begin{bmatrix} I_w \\ I_f \end{bmatrix} + \begin{bmatrix} 0 \\ V_f \end{bmatrix}, \tag{3.30a}$$

$$\vec{\Phi} = \begin{bmatrix} M_w & M_f \end{bmatrix} \begin{bmatrix} I_w \\ I_f \end{bmatrix}, \tag{3.30b}$$

where the subscript w denotes a wall element and f denotes a feedback coil element. The vector $\vec{\Phi}$ represents the flux measured by magnetic sensors near the wall and coil elements. A simple schematic showing DIII-D's wall and control coils divided into conducting finite elements is shown in figure 3.1.

3.1.4 Single Mode VALEN

At present, VALEN is equipped to handle a single plasma mode only. In this single mode limit, equations 3.11, 3.15 and 3.16 have the form

$$\Phi_w = L_w I^W + M_{wp} I^d + M_{wp} I^p \tag{3.31}$$

$$\Phi = M_{pw} I_w + L I^d + L I^p \tag{3.32}$$

$$\Phi = L_p I^p \frac{1}{1+s}. \tag{3.33}$$

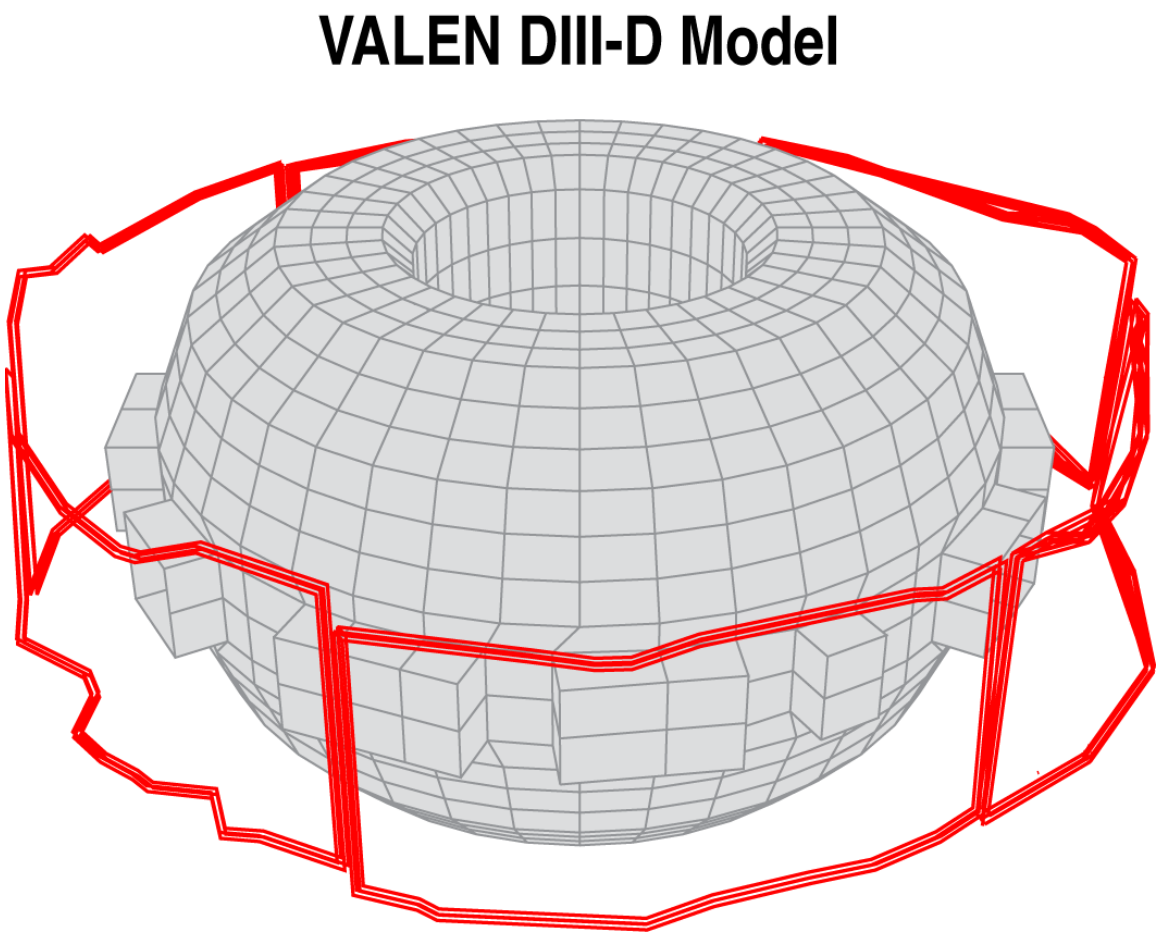

Figure 3.1: The DIII-D tokamak's wall and external C-coils divided into conducting finite elements.

Using these equations, current on the plasma surface, I^p, may be defined as

$$I^p = -\frac{1+s}{sL_p}(M_{pw}I_w + L_pI^d). \tag{3.34}$$

This substitution allows I^p to be removed from the coupled equations when new inductance matrices are defined like so

$$\mathscr{L}_{ww} = L_w - \frac{1+s}{sL_p} M_{wp} M_{pw} \tag{3.35a}$$

$$\mathscr{L}_{wp} = M_{wp} - \frac{1+s}{s} M_{wp} \tag{3.35b}$$

$$\mathscr{L}_{pw} = -\frac{1}{s} M_{pw} \tag{3.35c}$$

$$\mathscr{L}_{p} = -\frac{1}{s} L_p \tag{3.35d}$$

and coupled with the time evolution of flux given by equations 3.13 and 3.17. VALEN is now given in block matrix form as:

$$\begin{bmatrix} \mathscr{L}_{ww} & \mathscr{L}_{wp} \\ \mathscr{L}_{pw} & \mathscr{L}_{p} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} I_w \\ I_d \end{bmatrix} = - \begin{bmatrix} R_w & 0 \\ 0 & R_p \end{bmatrix} \begin{bmatrix} I_w \\ I_d \end{bmatrix} + \begin{bmatrix} V_w \\ 0 \end{bmatrix}, \tag{3.36a}$$

$$\vec{\Phi} = \begin{bmatrix} M_w & M_p \end{bmatrix} \begin{bmatrix} I_w \\ I_d \end{bmatrix}. \tag{3.36b}$$

3.1.5 VALEN Incorporates DCON

The Direct Criterion of Newcomb, or DCON[36], is an MHD stability code that analyzes a given axisymmetric toroidal equilibrium, and by a δW calculation (section 1.1.3.3), determines the stability of the equilibrium. DCON builds on the techniques of Newcomb[37], which analyzes the stability of cylindrical plasma, and extends them to axisymmetric toroidal plasma. VALEN uses a DCON mode pattern plot, which provides the 3D structure of the mode on the plasma surface by defining the perturbed magnetic field normal to the plasma surface, δB_n. The mode pattern plot also defines the eigenvalue basis functions

of equations 3.1 and 3.2. The displacement vector, $\vec{\xi}$, of equation 3.3 is now represented as a Fourier series:

$$\begin{pmatrix} \xi_s \\ \xi_r \end{pmatrix} = (r,\theta,\phi) \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \left. \begin{pmatrix} \xi_s \\ \xi_r \end{pmatrix} \right|_{m,n} (r) \times e^{i(m\theta - n\phi)}, \tag{3.37}$$

where ξ_s and ξ_r represent the surface and normal displacements, respectively. In practice, the infinite sums in equation 3.37 are truncated at some finite number M of components. An example mode pattern plot is shown in figure 3.2, which shows the cosine component of the mode (section 3.1.2). The single mode in VALEN is represented as two currents in VALEN as $\vec{I}_d = [I^c \;\; I^s]^T$, where I^c and I^s are the cosine and sine components, respectively.

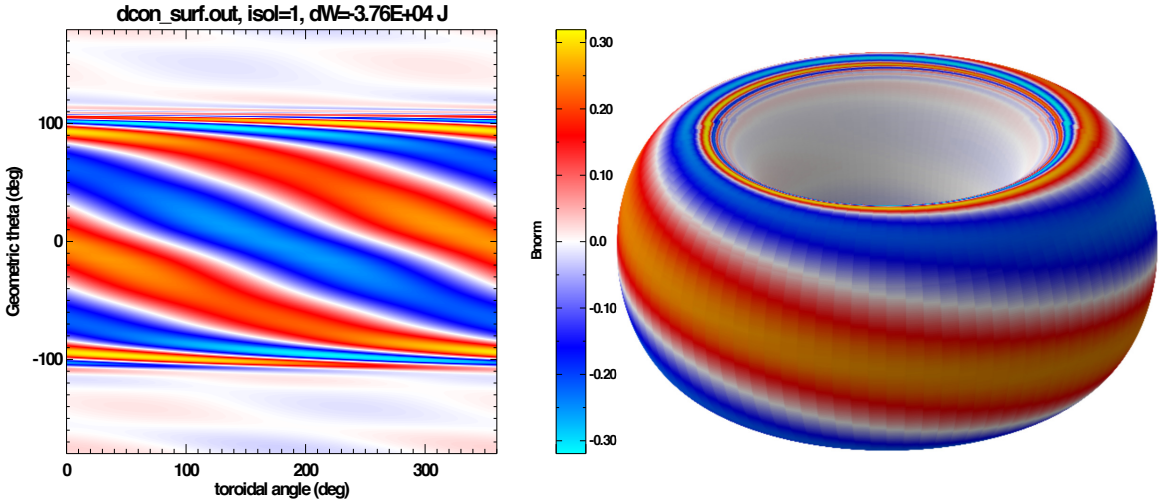

Figure 3.2: Normal magnetic field, δB_n, on the plasma surface of the cosine component of the most unstable plasma mode as computed by DCON.

The largest plasma physics approximation used in coupling an ideal MHD stability code like DCON to the lumped element circuit model of VALEN is to assume that the δW eigenmode geometry computed by DCON, which has no conducting structure around the plasma equilibrium, is not significantly altered by the presence of the surrounding passive conducting structure and active control coil fields. In other words, the no-wall kink

eigenmode structure is “rigid” when used in the VALEN state-space model. For tokamaks like DIII-D, with relatively sparse kink-mode spectra, this assumption has held up very well in experiments[38][39] where the internal kink mode radial structure was found to be in good agreement with the no-wall MHD code calculated eigenmode structure. This means turning on a control coil does not significantly alter the kink-mode geometry and therefore dynamically change the mutual inductive coupling between the mode and the control coil. If this were not the case, RWM feedback control would be a very complex non-linear problem to model and likely an intractable one.

3.2 State-Space VALEN

When more explicitly expressed as in equation 3.36a, the VALEN model with plasma mode in block matrix form is given by

$$\begin{bmatrix} \mathscr{L}_{ww} & \mathscr{L}_{wf} & \mathscr{L}_{wp} \\ \mathscr{L}_{fw} & \mathscr{L}_{ff} & \mathscr{L}_{fp} \\ \mathscr{L}_{pw} & \mathscr{L}_{pf} & \mathscr{L}_{p} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} I_w \\ I_f \\ I_d \end{bmatrix} = - \begin{bmatrix} R_w & 0 & 0 \\ 0 & R_f & 0 \\ 0 & 0 & R_p \end{bmatrix} \begin{bmatrix} I_w \\ I_f \\ I_d \end{bmatrix} + \begin{bmatrix} 0 \\ V_f \\ 0 \end{bmatrix}, \tag{3.38a}$$

$$\vec{\Phi} = \begin{bmatrix} M_w & M_f & M_p \end{bmatrix} \begin{bmatrix} I_w \\ I_f \\ I_d \end{bmatrix}. \tag{3.38b}$$

To get VALEN into standard state space control form (described in section 2.1),

$$\frac{d}{dt}\vec{x} = \mathbf{A}\vec{x} + \mathbf{B}\vec{u} \tag{3.39}$$

the inductance matrix is inverted and multiplied through the right side of the equation

$$\frac{d}{dt}\vec{I} = -L^{-1}R\vec{I} + L^{-1}\vec{V}_f, \tag{3.40}$$

where

$$\mathbf{A} = -L^{-1}R = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \mathbf{B} = L^{-1} = \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \end{bmatrix}. \tag{3.41}$$

The output equation given by 3.38b does not require any transformation because it already has the form of equation 2.2b. The input to the system is the applied potential on the control coil set being used for feedback. In single mode VALEN, the stability of the mode, and therefore the system matrix, $\mathbf{A}$, is a function of VALEN's parameters s and α. The eigenvalue of $\mathbf{A}$ with the largest real part determines the mode's growth rate, γ, and that eigenvalue's imaginary part determines the rotation rate, ω. Figures 3.3 and 3.4 show the growth rate and rotation rate as functions of s and α, respectively, for a given equilibrium. The equilibrium used is from DIII-D discharge 133103 at 3330ms.

3.2.1 Voltage Control VALEN

The feedback control problem is straightforward when the Audio Amplifiers (AAs, section 5.2.2) are used. The AAs amplify the input signal and the voltage calculated by the control law is multiplied by the inverse of the AA gain. The AAs then in turn amplify the signal by a gain and deliver the required voltage to the control coils.

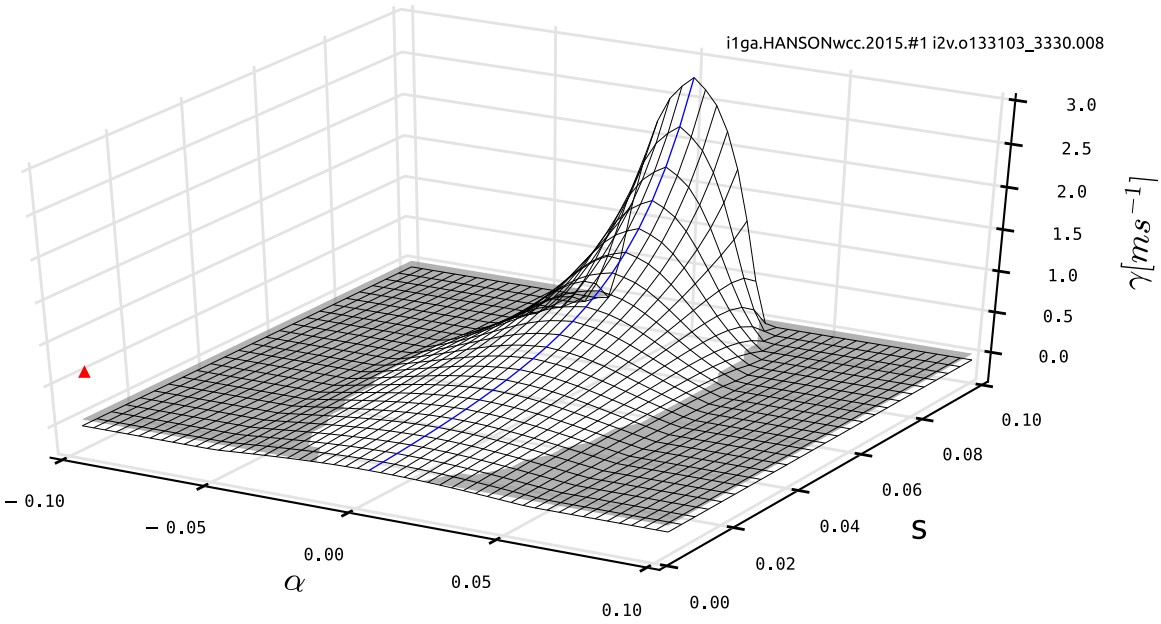

Figure 3.3: Mode growth rate, γ, as a function of s and α. The growth rate is determined as $\gamma = \Re(\lambda_{\max})$, where $\lambda_{\max}$ is the eigenvalue of $\mathbf{A}$ from equation 3.39 with the largest real part.

3.2.2 Current Control VALEN

Use of the VALEN model becomes more complex when used with the Switching Power Amplifiers (SPAs, section 5.2.1). The SPAs attempt to emulate a current source. Figure 3.5 shows the block diagram of the SPA internal dynamics. The gains G_p, G_i and G_d are set by potentiometer positions located physically on the SPA itself. The transfer function governing this system is linear and a state space model approximating the SPA's dynamics may be easily constructed from input/output data from the SPA and control coils. From figure 3.5, a generic state space model for the SPAs can be expressed as a two state system

$$\frac{d}{dt}\begin{bmatrix} I \\ V \end{bmatrix} = \mathbf{A}\begin{bmatrix} I \\ V \end{bmatrix} + \mathbf{B}I_{com}. \tag{3.42}$$

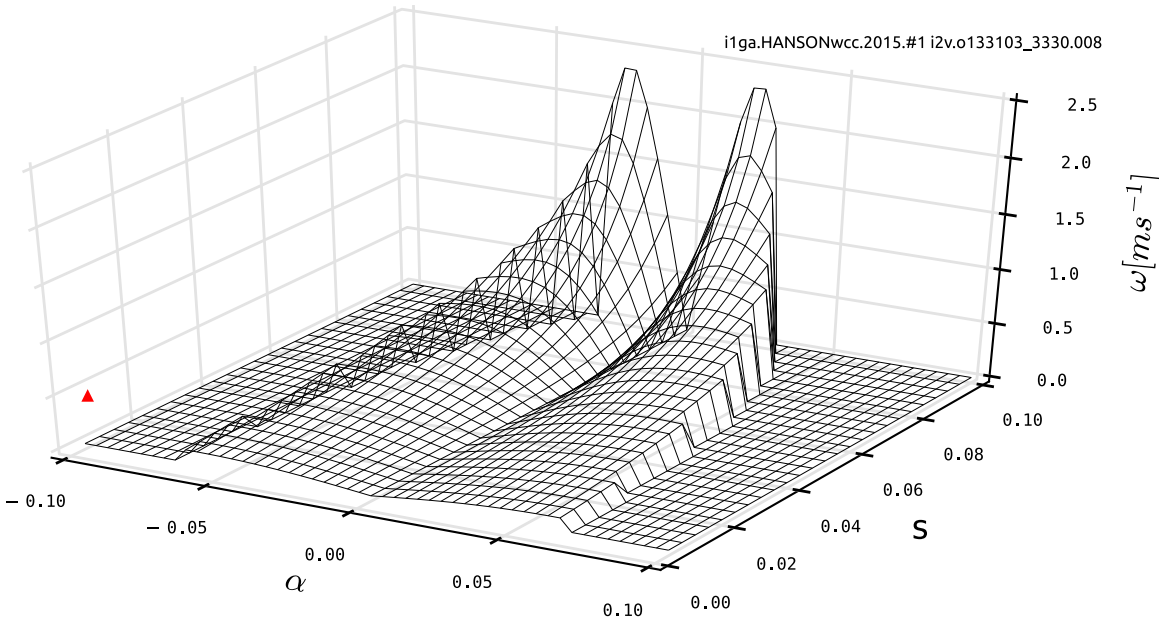

Figure 3.4: Mode rotation rate, ω, as a function of s and α. The rotation rate is determined as $\omega = \Im(\lambda_{\max})$, where $\lambda_{\max}$ is the eigenvalue of $\mathbf{A}$ from equation 3.39 with the largest real part. The blue line shows growth rates for α equal to 0.

To estimate the system and input matrices, $\mathbf{A}$ and $\mathbf{B}$, a least squares fit of experimental data is done

$$\frac{d}{dt}\begin{bmatrix} I \\ V \end{bmatrix}\begin{bmatrix} I \\ V \\ I_{com} \end{bmatrix}^{+} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \end{bmatrix}, \tag{3.43}$$

where in this case I, V and I_{com} represent a time series of data, + indicates the Moore-Penrose pseudo-inverse[30] and $\mathbf{A}$ and $\mathbf{B}$ are composed of elements

$$\mathbf{A} = \begin{bmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \beta_{11} \\ \beta_{21} \end{bmatrix}. \tag{3.44}$$

AC waveform testing of SPA1 connected to C79/C259 (C-coil pairs) from shot 153236 is shown in figure 3.6 and is the experimental data set used for identifying the SPA1 model. Similar data sets were used for SPA2 and SPA3. This new model for the SPAs is now

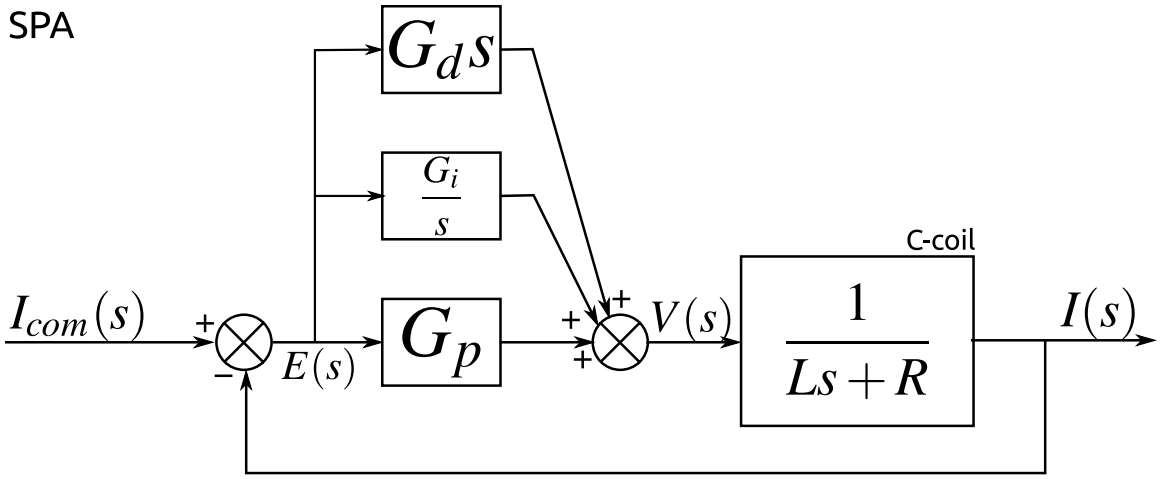

Figure 3.5: Laplace transformed block diagram of a Switching Power Amplifier (SPA).

inserted into the existing state-space VALEN model. VALEN already includes the current on feedback coils. Because the internal controller of the SPA will now regulate current on the control coil, the row associated with current on that coil is zeroed out and necessary components from the SPA model are then inserted. Also, the state of VALEN is augmented to include potential on the control coil. This modified VALEN model now has the block matrix form

$$\frac{d}{dt}\begin{bmatrix} I_w \\ I_f \\ I_d \\ V_f \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_{11} \\ 0 & \alpha_{11} & 0 & \alpha_{12} \\ a_{31} & a_{32} & a_{33} & b_{31} \\ 0 & \alpha_{21} & 0 & \alpha_{22} \end{bmatrix} \begin{bmatrix} I_w \\ I_f \\ I_d \\ V_f \end{bmatrix} + \begin{bmatrix} 0 \\ \beta_{11} \\ 0 \\ \beta_{21} \end{bmatrix} I_{com}, \tag{3.45}$$

and the output equation for VALEN is now

$$\vec{\Phi} = \begin{bmatrix} M_w & M_f & M_p & 0 \end{bmatrix} \begin{bmatrix} I_w \\ I_f \\ I_d \\ V_f \end{bmatrix}. \tag{3.46}$$

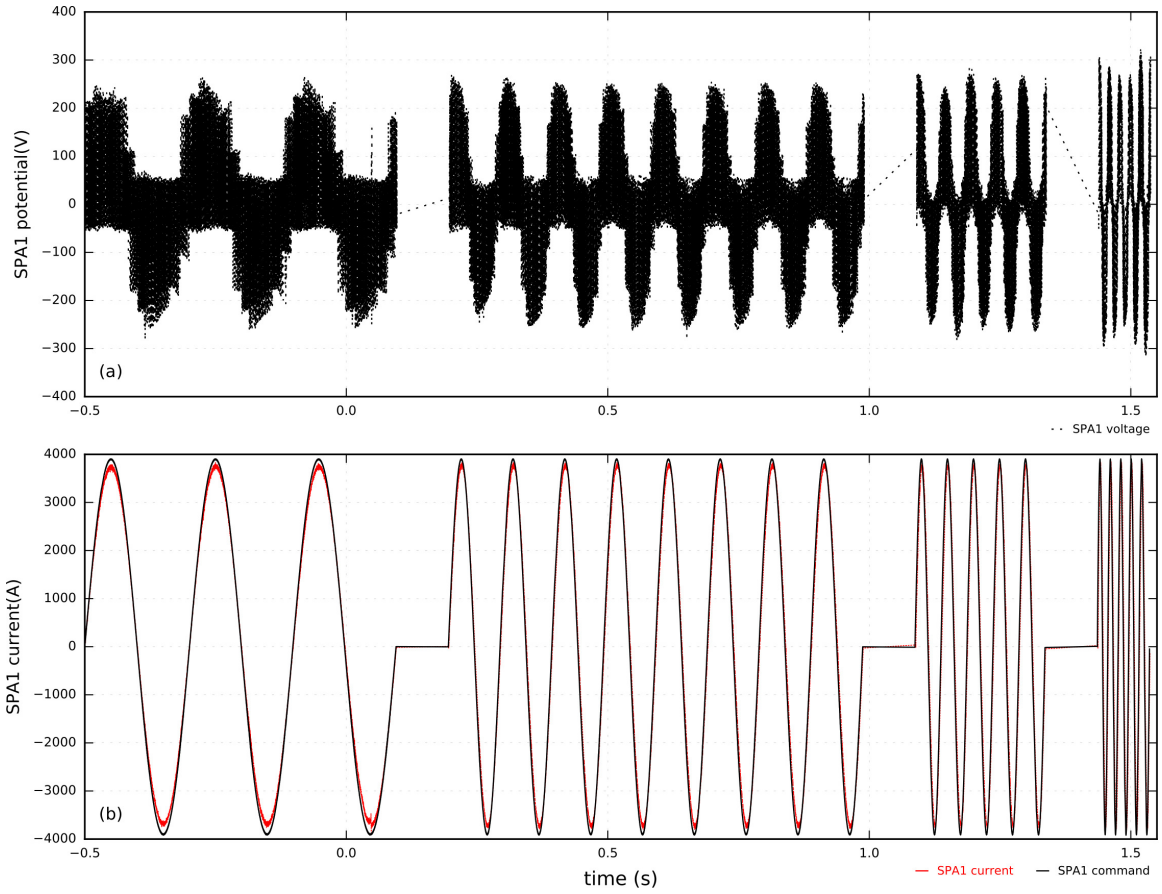

Figure 3.6: AC waveform testing for SPA1->C79/C259. (a) shows the Pulse Width Modulated (PWM) signal of potential (V). (b) shows the SPA command in red with the actual SPA current (A) in black. Frequencies for the commanded sinusoidal waveforms are 5, 10, 20 and 50 Hz.

Similarly to voltage control, the current calculated by the control law is multiplied by the inverse of the SPA gain. The SPAs in turn amplify the signal by their gain and the internal PID controller delivers the required voltage to the control coils.

VALEN requires four items from the user to generate the ordinary differential equation and output equation given by 3.38a and 3.38b, respectively. These items are: the geometry of conducting structures, which does not change for DIII-D, a δB_n distribution on the plasma surface for the δW eigenmode computed from an equilibrium for the target discharge by an MHD stability code such as `DCON`, and the parameters s and α. The model

reduction techniques described in section 2.2 are used to reduce the state dimension of VALEN from approximately 1400 states to 32 (section 6.2). The selection of parameters s and α from experimental data is described in section 7.2.1. The next chapter described the techniques that are used to design a feedback control system using a state-space linear system such as VALEN in state-space form.

This chapter contain material which has been submitted for publication in *Nuclear Fusion*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Atomic Energy Agency, 2017. The dissertation author was the primary investigator and author of this paper.

Chapter 4

Optimal Control

Optimal control theory is concerned with the stabilization of linear systems as described in chapter 2. The objective of the optimal control problem is to regulate a system's output to some desired value (zero is perfectly acceptable). The output or measurement of this system may also be corrupted by noise. To achieve this, optimal control theory seeks to optimize, that is to say minimize, a cost function quadratic in state and control, as well as error between actual system state and an estimated state using measurements corrupted by zero-mean Gaussian noise. This is often called Linear Quadratic Gaussian (LQG) control as the system dynamics are Linear, the cost function is Quadratic and the noise is Gaussian. LQG control will also minimize the $\mathcal{H}_2$-norm of the system's transfer function. For this reason optimal control, specifically $\mathcal{H}_2$ optimal control, is often used interchangeably with LQG control. These terms will be used synonymously herein. This chapter outlines the fundamentals of optimal control required to design controllers and state estimators necessary for VALEN-based feedback.

4.1 Linear Quadratic Regulator

4.1.1 The Regulation Problem

In control theory the regulation problem involves regulating a system's output, $\vec{y}(t)$ (equation 2.1b), to some desired output, $\vec{r}(t)$. From these two vectors, an error may be defined as

$$\vec{e}(t) = \vec{r}(t) - \vec{y}(t). \tag{4.1}$$

Thus, the control objective is to determine a control $\vec{u}(t)$ such that the error defined in 4.1 is as close to $\vec{0}$ as possible[40]. A perfectly acceptable desired output is $\vec{0}$, which from 2.1b implies that the desired state is either $\vec{0}$ or in the nullspace of the system's output matrix $\mathbf{C}$ (equation 2.1b). The nullspace of $\mathbf{C}$ is any vector $\vec{z}$ that satisfies $\mathbf{C}\vec{z} = \vec{0}$, where $\vec{z} \neq \vec{0}$[30]. In the case when the desired state is $\vec{0}$, it can be seen from equation 2.1a that in the absence of control $\vec{u}(t)$, $\vec{0}$ is an equilibrium of the system. This means that, starting from the initial condition $\vec{x}_0 = \vec{0}$, the system will not depart from this state. Consequently, the regulator problem may be stated as: given any state, $\vec{x}_0$ of the plant 2.1a at t_0, form a control input, $\vec{u}(t)$ for $t \geq t_0$, based on $\vec{x}(t)$, which is able to move the state $\vec{x}_0$ to $\vec{0}$[25]. In most cases, the state of a system is not directly measured, but rather linear combinations or projections of it. This means that $\vec{u}(t_1)$ must be determined from the knowledge of $\vec{y}(t)$ for $t \leq t_1$. The problem is broken into two steps to accomplish this: (1) computation of the "best approximation" $\hat{x}(t_1)$ of the state $\vec{x}(t_1)$ from the knowledge of $\vec{y}(t)$ for $t \leq t_1$, and (2) computation of $\vec{u}(t_1)$ given $\hat{x}(t_1)$[25].

4.1.2 State Variable Feedback

Given the state-space equations of 2.1a and 2.1b, a stabilizing control law is given by:

$$u(t) = -\mathbf{K}\vec{x}(t), \tag{4.2}$$

which results in equation 2.1a taking the form:

$$\frac{d}{dt}\vec{x}(t) = (\mathbf{A} - \mathbf{BK})x(t). \tag{4.3}$$

The matrix $\mathbf{K}$ may be constructed to place the eigenvalues of the square matrix $(\mathbf{A} - \mathbf{BK})$ in desirable locations, meaning that all the eigenvalues have real parts less than 0. Now, the remaining challenge is how to construct $\mathbf{K}$. To do this in the discrete time case, first a cost function quadratic in state and control input is chosen having the form:

$$\mathcal{J} = \frac{1}{2}\sum_{k=0}^{N}[\vec{x}_k^T\mathbf{Q}\vec{x}_k + \vec{u}_k^T\mathbf{R}\vec{u}_k]. \tag{4.4}$$

This cost function is for a discrete-time system (section 2.1), where k is the sample number and N is the number of samples. The infinite time horizon regulation problem deals with the limit when $N \to \infty$ and will be assumed for the discussion that follows. Equation 4.2, subject to the constraint that equation 4.4 is minimized, is called the Linear Quadratic Regulator (LQR). A complete derivation of the LQR is available in [41]. The control input $\vec{u}_k$ will be chosen so that equation 4.4 is minimized[41]. The matrices $\mathbf{Q}$ and $\mathbf{R}$ are selected based on the relative importance of states and controls and must be symmetric and nonnegative definite. The selection of "penalty" matrices $\mathbf{Q}$ and $\mathbf{R}$ is done by the designer and is covered in detail in section 6.4. Another way of stating the optimal control problem

given by the discrete time plant of 2.2a and the cost given by 4.4, is to minimize 4.4 subject to the constraint,

$$-\vec{x}_{k+1} + \mathbf{F}\vec{x}_k + \mathbf{G}\vec{u}_k = 0, \;\; k = 0, 1 \ldots N. \tag{4.5}$$

Lagrange multipliers may be used to solve this constrained-minima problem. This Lagrange multiplier will be called $\vec{\lambda}_{k+1}$ for each k and is often times referred to as the costate or adjoint. Now the cost is rewritten as

$$\mathcal{J}' = \sum_{k=0}^{N} \left[\frac{1}{2}\vec{x}_k^T \mathbf{Q}\vec{x}_k + \frac{1}{2}\vec{u}_k^T \mathbf{R}\vec{u}_k + \vec{\lambda}_{k+1}^T(-\vec{x}_{k+1} + \mathbf{F}\vec{x}_k + \mathbf{G}\vec{u}_k) \right] \tag{4.6}$$

and the minimum of $\mathcal{J}'$ is found with respect to $\vec{x}_k$, $\vec{u}_k$, and $\vec{\lambda}_k$ by differentiation with respect to each vector and leads to the following three equations:

$$\frac{\partial \mathcal{J}'}{\partial \vec{u}_k} = \vec{u}_k^T \mathbf{R} + \vec{\lambda}_{k+1}^T \mathbf{G} = 0 \tag{4.7a}$$

$$\frac{\partial \mathcal{J}'}{\partial \vec{\lambda}_{k+1}} = -\vec{x}_{k+1} + \mathbf{F}\vec{x}_k + \mathbf{G}\vec{u}_k = 0 \tag{4.7b}$$

$$\frac{\partial \mathcal{J}'}{\partial \vec{x}_k} = \vec{x}_k^T \mathbf{Q} - \vec{\lambda}_k^T + \vec{\lambda}_{k+1}^T \mathbf{F} = 0. \tag{4.7c}$$

The adjoint equation of 4.7c may be rearranged as a backward difference equation,

$$\vec{\lambda}_k = \mathbf{F}^T \vec{\lambda}_{k+1} + \mathbf{Q}\vec{x}_k. \tag{4.8}$$

The control input is given by rearranging 4.7a, which yields,

$$\vec{u}_k = -\mathbf{R}^{-1}\mathbf{G}^T \vec{\lambda}_{k+1}. \tag{4.9}$$

Equations 4.7b, 4.8 and 4.9 form a set of coupled difference equations that may be used to solve for $\vec{x}_k$, $\vec{u}_k$, and $\vec{\lambda}_k$ so long as suitable boundary conditions are specified. From equation 4.4 it can be inferred that $\vec{u}_N = 0$ in order to minimize $\mathcal{J}$. With $\vec{u}_N = 0$ inserted into 4.7a we have $\vec{\lambda}_{N+1} = 0$, that with equation 4.8 gives $\vec{\lambda}_N = \mathbf{Q}\vec{x}_N$. The initial state, $\vec{x}_0$, must be specified by the problem statement as an initial condition. It is helpful to assume the following relationship between the state and adjoint (costate)

$$\vec{\lambda}_k = \mathbf{X}_k \vec{x}_k \tag{4.10}$$

which transforms the two-point boundary-value problem in $\vec{x}$ and $\vec{\lambda}$ into a single-point boundary-value problem in $\mathbf{X}$. In the infinite time horizon, as $k \to \infty$, the solution of $\mathbf{X}$ is constant. By using the definition of equation 4.10 and combining it with equations 4.7a, 4.8 and 2.2a, the following matrix equation results

$$\mathbf{X} = \mathbf{F}^T\mathbf{X}\mathbf{F} - \mathbf{F}^T\mathbf{X}\mathbf{G}(\mathbf{G}^T\mathbf{X}\mathbf{G} + \mathbf{R})^{-1}\mathbf{G}^T\mathbf{X}\mathbf{F} + \mathbf{Q} \tag{4.11}$$

and is called the Discrete Time Algebraic Riccati equation. Many computational routines exist to solve 4.11 for $\mathbf{X}$ and the time-invariant optimal control gain $\mathbf{K}$ from equation 4.2 is given by

$$\mathbf{K} = (\mathbf{R} + \mathbf{G}^T\mathbf{X}\mathbf{G})^{-1}\mathbf{G}^T\mathbf{X}\mathbf{F}. \tag{4.12}$$

4.2 Optimal State Estimation

The previous section 4.1.2 was concerned with determining a stabilizing control law, assuming perfect knowledge of the system state, $\vec{x}$. This situation is hardly the case,

and the next step of the optimal control problem is determining an optimal state estimate, $\hat{x}$, based on what is measured from the system, $\vec{y}$. In reality the equations that define the discrete-time plant and output equations, 2.2a and 2.2b, are corrupted by noise and maybe written as

$$\vec{x}_{k+1} = \mathbf{F}\vec{x}_k + \mathbf{G}\vec{u}_k + \vec{w}_k \tag{4.13a}$$

$$\vec{y}_k = \mathbf{C}\vec{x}_k + \vec{v}_k, \tag{4.13b}$$

where $\vec{w}_k$ and $\vec{v}_k$ are the process and measurement noise, respectively, are zero-mean,

$$\mathscr{E}[\vec{w}_k] = \mathscr{E}[\vec{v}_k] = 0,$$

are "white" or uncorrelated in time, meaning

$$\mathscr{E}[\vec{w}_i\vec{w}_j^T] = \mathscr{E}[\vec{v}_i\vec{v}_j^T] = 0 \text{ if } i \neq j,$$

and have covariances

$$\mathscr{E}[\vec{w}_k\vec{w}_k^T] = \mathbf{Q}_\mathrm{o}$$

$$\mathscr{E}[\vec{v}_k\vec{v}_k^T] = \mathbf{R}_\mathrm{o}.$$

In the preceding equations, $\mathscr{E}[\cdot]$ denotes the expected value of a random variable, which, if explicitly stated, is $\mathscr{E}[\vec{x}] = \frac{1}{N}\sum_{i=1}^{N} x_i p_i$, where p_i is the probability of occurrence of x_i. An optimal state estimate would be one that minimizes the error between the state estimate,

$\hat{x}$, and the actual state, $\vec{x}$[41]. This may be quantified as the estimation error covariance,

$$\mathbf{P}_{\mathrm{e}} = \mathscr{E}[(\hat{x}-\vec{x})(\hat{x}-\vec{x})^T]. \tag{4.14}$$

4.2.1 The Kalman Filter

State estimation using discrete measurements is a dynamic process and the state estimate may be updated in time based on new measurements and known system dynamics[42]. All the measurements up to and including time k may be used to estimate the system state, and will be denoted $\vec{x}_k^+$. However, if only the measurements before, but not including, time k are available, then the estimate is written $\vec{x}_k^-$. The estimation error covariance of 4.14 can be similarly denoted

$$\mathbf{P}_k^- = \mathscr{E}[(\hat{x}_k^- - \vec{x}_k)(\hat{x}_k^- - \vec{x}_k)^T]$$
$$\mathbf{P}_k^+ = \mathscr{E}[(\hat{x}_k^+ - \vec{x}_k)(\hat{x}_k^+ - \vec{x}_k)^T].$$

The following equations are used to update the state estimate and estimation error covariance in time

$$\hat{x}_k^- = \mathbf{F}\hat{x}_{k-1}^+ + \mathbf{G}\vec{u}_{k-1} \tag{4.15a}$$
$$\mathbf{P}_k^- = \mathbf{F}\mathbf{P}_{k-1}^+\mathbf{F}^T + \mathbf{Q}_{\mathrm{o}} \tag{4.15b}$$

and are called the time update equations. Similarly, the state estimate and estimation error covariance may be updated by using the measurement, $\vec{y}_k$ at time k. The estimation

error covariance is updated like so

$$\mathbf{P}_k^+ = ((\mathbf{P}_k^-)^{-1} + \mathbf{C}^T\mathbf{R}_\mathrm{o}\mathbf{C})^{-1} \tag{4.16}$$

and the state estimate is updated as

$$\mathbf{K}_k = \mathbf{P}_k^+\mathbf{H}^T\mathbf{R}_\mathrm{o}^{-1} \tag{4.17a}$$

$$\hat{x}_k^+ = \hat{x}_k^- + \mathbf{K}_k(\vec{y}_k - \mathbf{C}\vec{x_k^-}), \tag{4.17b}$$

where $\mathbf{K}$ is the estimator gain but commonly referred to as the Kalman gain. These are called the measurement update equations. By increasing the time index by one in 4.15a and inserting 4.16, a single equation can be used for the state estimate:

$$\hat{x}_{k+1}^- = \mathbf{F}(\hat{x}_k^- + \mathbf{K}_k(\vec{y}_k - \mathbf{C}\hat{x}_k^-)) + \mathbf{G}\vec{u}_k. \tag{4.18}$$

The same technique may be applied to the estimation error covariance by combining 4.15b and 4.16 to give

$$\mathbf{P}_{k+1}^- = \mathbf{F}((\mathbf{P}_k^-)^{-1} + \mathbf{C}^T\mathbf{R}_\mathrm{o}\mathbf{C})^{-1}\mathbf{F}^T + \mathbf{Q}_\mathrm{o}. \tag{4.19}$$

Usually, on the infinite time horizon, the estimation error covariance will stabilize to some steady-state value, that is to say, as k becomes large enough, $\mathbf{P}_{k+1} = \mathbf{P}_k$. The estimation error covariance equation now becomes

$$\mathbf{P} = \mathbf{F}\mathbf{P}\mathbf{F}^T - \mathbf{F}\mathbf{P}\mathbf{C}^T(\mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{R}_\mathrm{o})^{-1}\mathbf{C}\mathbf{P}\mathbf{F}^T + \mathbf{Q}_\mathrm{o} \tag{4.20}$$

which has almost identical form to equation 4.11. This steady-state estimation error covariance may be used to calculate the Kalman gain of equation 4.17a. This steady-state formulation saves computing new gains and covariances at every time step, which may be impossible if the system state size is sufficiently large. The full derivation of the Kalman Filter (KF) can be found in [42]. The near identical forms of 4.11 and 4.20 are why the optimal control and optimal estimation problems are duals and may be solved using similar pre-packaged computer routines.

A LQR coupled with the state estimate provided by a KF completes the LQG control problem. In the LQG control problem: the dynamics are *linear*, the cost function is *quadratic* and the noise is *gaussian*. The techniques described here are used to design the VALEN-based feedback algorithm that is described further in chapter 6.

Chapter 5

Control Hardware on DIII-D

This chapter gives an overview of systems on the DIII-D tokamak that are used for feedback control of MHD modes.

5.1 Magnetics and 3D Field Coils

DIII-D can measure low frequency (<20kHz) toroidal modes with toroidal mode number, $n > 0$, with 34 poloidal can-type probe difference pairs arranged in five toroidal arrays, and radial magnetic field with 38 saddle loop difference pairs arranged in six toroidal arrays[43]. Both B_p and B_r arrays span 360 degrees of toroidal angle. A simple schematic of DIII-D with coil and sensor configurations is shown in figure 5.1. Sensors located at, above and below the low field side midplane are used and their respective locations are shown in figure 5.1. DIII-D is equipped with 2 sets of electromagnetic control coils for creating 3D magnetic fields. The I-coils, which are interior to the vacuum vessel, are arranged in two rows of six, above and below the midplane. The C-coils, which are external to the vacuum vessel, are arranged in a single row of six coils located on

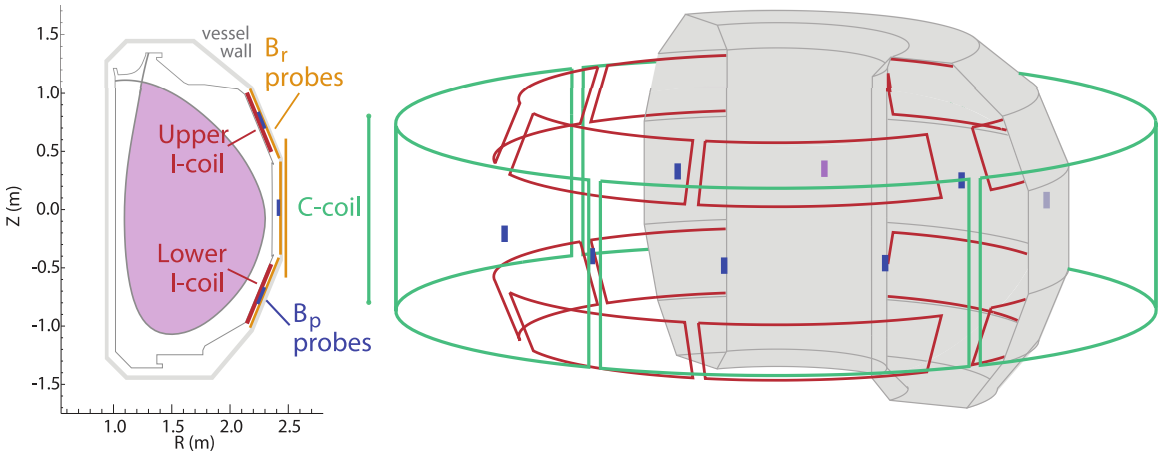

Figure 5.1: Simple cross sectional view and schematic of DIII-D sensors and 3D control coils.

the midplane. Not shown in figure 5.1 are the axisymmetric coils responsible for the equilibrium magnetic field that is much larger than the fields produced by the control coils. These 3D coils are used primarily for Error Field Correction (EFC)[44] and other 3D field control[45]. Error fields arise from small errors in the symmetry of a tokamak's magnetic field, usually as a result of imperfect coil and vessel fabrication. The I and C-coils are typically configured in $n = 1$ quartets or pairs, respectively, for feedback on the $n = 1$ component of the RWM, which is expected to be dominant. This translates to three control commands to amplifiers connected to the lead coil in each quartet or pair. Vacuum coupling between sensors and coils, i.e. poloidal field coils, needs to be accounted for and eliminated prior to using sensor measurements for feedback. System latency should be kept well below the lowest expected RWM growth time, 2.5ms. The I-coils are typically used for RWM feedback due to their proximity to the plasma edge and because they are not encumbered by the relatively lengthy magnetic flux diffusion time through the vacuum vessel. The present algorithm for RWM feedback uses Proportional Integral Derivative (PID) control, using only some of the available poloidal field sensors, to determine voltages or currents to be applied to the coils[45].

Within the vacuum vessel of future tokamaks that are expected to operate as reactors will be a nuclear environment, an environment in which feedback control coils may not survive for the lifetime of the machine. For this reason, feedback techniques using control coils located outside of the vacuum vessel will be essential and due to the delay on flux from external coils, more advanced control algorithms than PID will be needed.

5.2 Power Supplies

At present, two power supplies are available to drive DIII-D's 3D field coils. The Switching Power Amplifiers (SPAs) and Audio Amplifiers (AAs). The main difference between these two supplies is the input signal. The SPAs attempt to emulate a current source and take a desired current as input. The AAs simply amplify a voltage signal and were designed originally to drive audio equipment. Specifications for each follow.

5.2.1 Switching Power Amplifiers (SPAs)

The Robicon Corporation System Line-up 5000A Power Supplies, or more commonly referred to as the SPAs, are the primary power supplies for 3D fields. There are four sets of SPAs, SPA1 through SPA4. Additionally, they may be subdivided into sub SPAs, e.g. SPA1a, SPA1b and SPA1c. Because the SPAs take an input of desired current, and current is proportional to magnetic field, they are primarily used for EFC with either the I or C-coils. To act as a current source, the SPAs have an internal control circuit that measures current on the control coil and then creates an error signal for use by a Proportional Integral Derivative (PID) controller to regulate voltage on the coil. Gains for the PID controller are set by potentiometer positions physically located on the SPA itself. This potential is

Pulse Width Modulated (PWM) between -300 and +300VDC by a switching circuit. PWM is a modulation technique used to get the results of an analog signal by pulsing a signal between on and off. An example of the SPA's modulated voltage signal is shown in the upper plot of figure 3.6. When operated in full groups, the SPAs are rated to 2800A from .5 to 15Hz and for 3733A from 15 to 100Hz. When operated as sub SPAs, the ratings for each sub SPA are effectively one third of each full SPA. The gain for the input signal is 500A/V[46].

5.2.2 Audio Amplifiers (AAs)

The Techron 7700 Series Power Supply Amplifiers, or more commonly known as the Audio Amplifiers, are single channel industrial amplifiers that provide precision amplification from DC to 40kHz. Although the AAs may be used in controlled current mode, at DIII-D they are used in controlled voltage mode. This means no internal control loop is used and the amplifiers merely amplify the input signal received with fixed, flat gain over a specified bandwidth. Three groups of eight amplifiers are operated in parallel to power a pair or quartet lead coil. The AAs are rated to approximately 1500A per group. The gain for the input signal is roughly 30V/V[47].

5.3 Tesibius

Tesibius is a feedback control system built to perform advanced RWM control using a Graphical Processing Unit (GPU). The system described herein is largely based on the system installed at the HBT-EP tokamak[48], as it demonstrated the concept of using a GPU for fast real-time control computation in the microsecond regime, a task that is non-traditional in the relatively young field of GPU computing[49][50]. GPUs are

programmed in conventional programming languages like C, and therefore can leverage all the benefits of digital control such as decision-making or logic flow and extremely flexible control programs[41]. Major differences between DIII-D's and HBT-EP's system are host computer architecture, 64bit (DIII-D) versus 32bit (HBT-EP) host, a superior GPU and the use of an LQG algorithm for MHD mode control. Tesibius integrates the following components into a low latency, high performance system:

- NVIDIA Tesla K20c GPU, 5 GB RAM GPU.
- D-TACQ Solutions ACQ196 96 channel, 16bit digitizer.
- D-TACQ Solutions AO32 32 channel, 16bit analog output module.
- National Instruments PXI-PCIe8362, MXI-Express, 2 Port PCIe host bus adapter.
- Supermicro X9DAI-0 Motherboard running 64bit CentOS 6.5 with kernel 2.6.32.

The GPU resides in a PCIe-x16 slot on the motherboard in the same root complex as the National Instruments Host Bus Adapters (HBAs). The HBAs each connect via cable to a Rear Transition Module (RTM-T) attached to either the ACQ196 or AO32. The ACQ196, AO32 and both RTM-Ts are housed in a 2U Compact Peripheral Component Interconnect (CPCI) chassis. When memory buffers on the GPU are allocated for input and output, their physical addresses on the system are found with NVIDIA's GPUDirect Remote Direct Memory Access (RDMA) framework[51]. These physical addresses are passed to the RTM-T device drivers in the operating system's kernel for real-time streaming of data to the GPU's onboard memory. On every sample, new data are streamed from the digitizer, the algorithm operates on that data and writes its command to the output buffer. The output buffer is read by the output module and does a zero order hold, i.e. the value in the output

buffer is maintained throughout the sample interval. A diagram of the control system is shown in figure 5.2.

Table 5.1: Pointnames of signals teed from the DIII-D Plasma Control System to Tesibius

Magnetics Probes	Coil Currents	Plasma Current
ESLD079	F6A	IP_PROBES
ESLD139	F7A	
ESLD199	F6B	
ISLD079U	F7B	
ISLD139U	IU30	
ISLD199U	IU90	
MPID067U	IU150	
MPID097U	BCOIL	
MPID127U		
MPID157U		
USILD079		
UISLD139		
UISLD199		
LISLD079		
LISLD139		
LISLD199		
UMPID037		
UMPID097		
UMPID157		
LMPID037		
LMPID097		
LMPID157		
MPID66M020		
MPID66M200		

Due to space constraints, Tesibius only temporarily has access to 24 of the 72 total difference pairs of magnetics probes and saddle loops. Magnetics signals are the measurements used by the VALEN model described in chapter 3. In addition to magnetics, Tesibius gets signals for selected poloidal field coils, lead I-coils ($n = 1$ quartet configuration), toroidal field coil current and plasma current. The toroidal field current and plasma current can be used for EFC. Table 5.1 lists the pointnames that Tesibius tees from the DIII-D Plasma Control System (PCS).

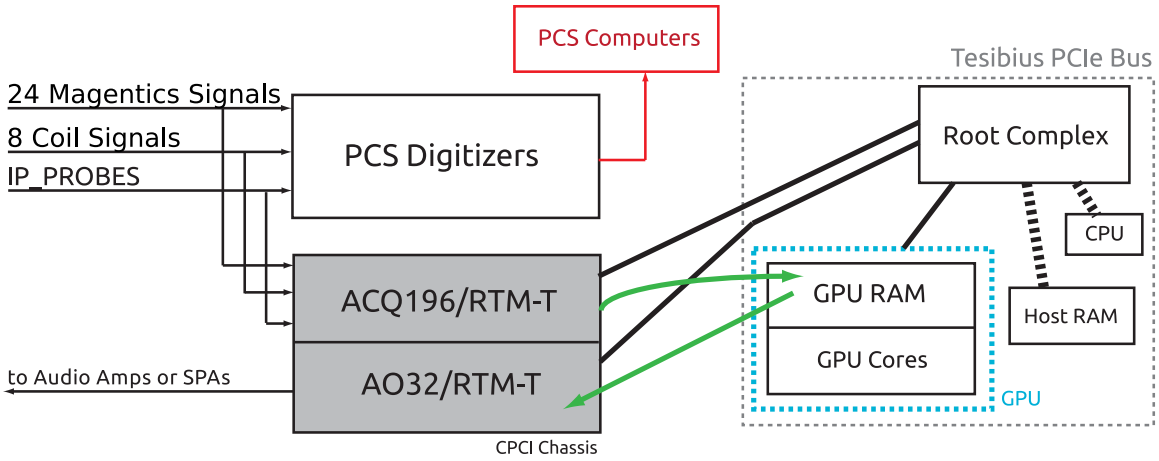

Figure 5.2: Control system (Tesibius) diagram. Tesibius operates in parallel with DIII-D's Plasma Control System (PCS).

5.4 Input/Output Latency

To test Tesibius' input/output latency, a 10kHz sine wave was teed to an oscilloscope and the input of the ACQ196 digitizer. The output of the AO32 was then sent to the same oscilloscope. A simple program that uses the output buffer as the input buffer was run to directly feedback the sine wave to the oscilloscope. This way, the cross-phase can be calculated by Fourier transforming the signals and then, based on the frequency, the time-delay between sine waves can be used to estimate the time required for analog to digital (A/D) and digital to analog (D/A) conversion. Because the host uses the same buffer for input and output, there should be no time wasted on the host for extraneous memory transfers. This was repeated for sample times from 2μs to 11μs. A sample time of 3μs yielded the shortest latency of approximately 12μs, as can be seen in figure 5.3. Results of all the runs are shown in figure 5.4. A linear fit of latency for sample times greater than or equal to 5μs is also shown in figure 5.4. The non-linearity below 5μs indicates that the AO32 is pulling data from the memory buffer before the digitizer has sent new data

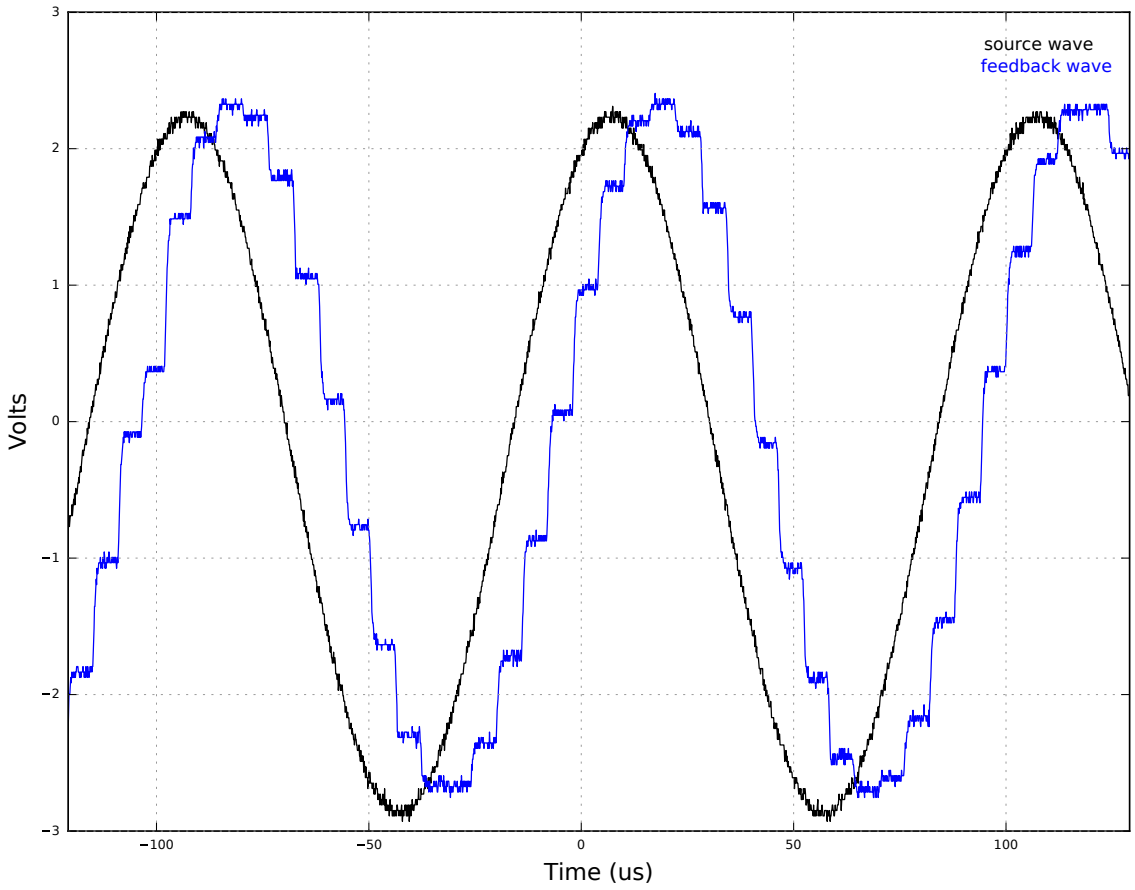

Figure 5.3: Illustration of input/output latency measurement. The control input is a 10kHz sine wave (black trace), which is passed unchanged as the control output (blue trace). Fourier analysis is used to determine the latency between traces. Sample time used for this plot is 3μs.

to the buffer and so the AO32 lags behind by a sample period. The offset of the linear fit, approximately 5μs, indicates the time required for the AO32 to pull data from the buffer and for the D/A conversion because input and output data are pushed and pulled at the same time. For the 13μs sample time presently used for the LQG algorithm (section B.2.2), the latency is approximately 25μs. This is acceptable as 25μs $\lll$ 2.5ms, which is the expected growth time of the RWM, and is comparable with the latency observed on the control system at HBT-EP[49][50].

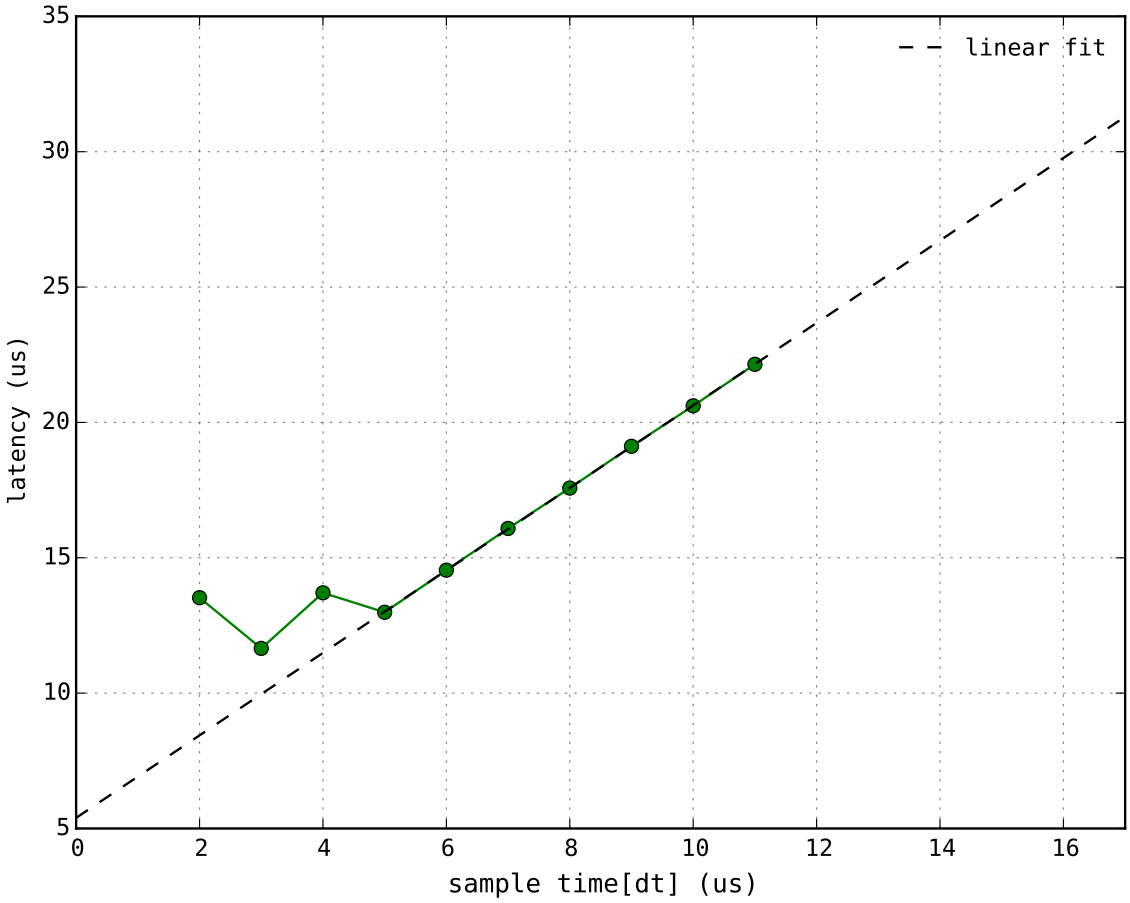

Figure 5.4: Sample times from 2μs to 11μs are plotted against their associated latency. The dashed line shows a linear fit of the data for sample times of 5μs and greater.

This chapter contains material which has been accepted for publication in *Control Engineering Practice*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. The dissertation author was the primary investigator and author of this paper.

Chapter 6

LQG Design for Real-Time Feedback

6.1 Parameter Selection

As discussed in chapter 3, s and α determine the state-space stability of VALEN's system matrix, $\mathbf{A}$. This means that the unstable mode modeled by VALEN will grow in time as $b(t) = b_0 e^{t(\gamma + i\omega)}$, where the growth rate is $\gamma = \Re(\lambda_{\max})$, ω is the rotation rate $\omega = \Im(\lambda_{\max})$ and $\lambda_{\max}$ is the largest eigenvalue of $\mathbf{A}$. Figure 6.1 shows the mode's growth rate as a function of s, while $\alpha = 0$. The curve in blue of figure 6.1 is the same as the blue line shown in figure 3.3 and using this figure, values of s can be determined to give realistic mode growth rates. All controllers used in simulations and experiments described in chapters 7 and 8 were designed with $\alpha = 0$. Routines in the software package `SLICOT`[52], exposed through the `Python-Control` module[53], were used to perform the balanced realization and model reduction of VALEN and for computation of the controller and observer gains.

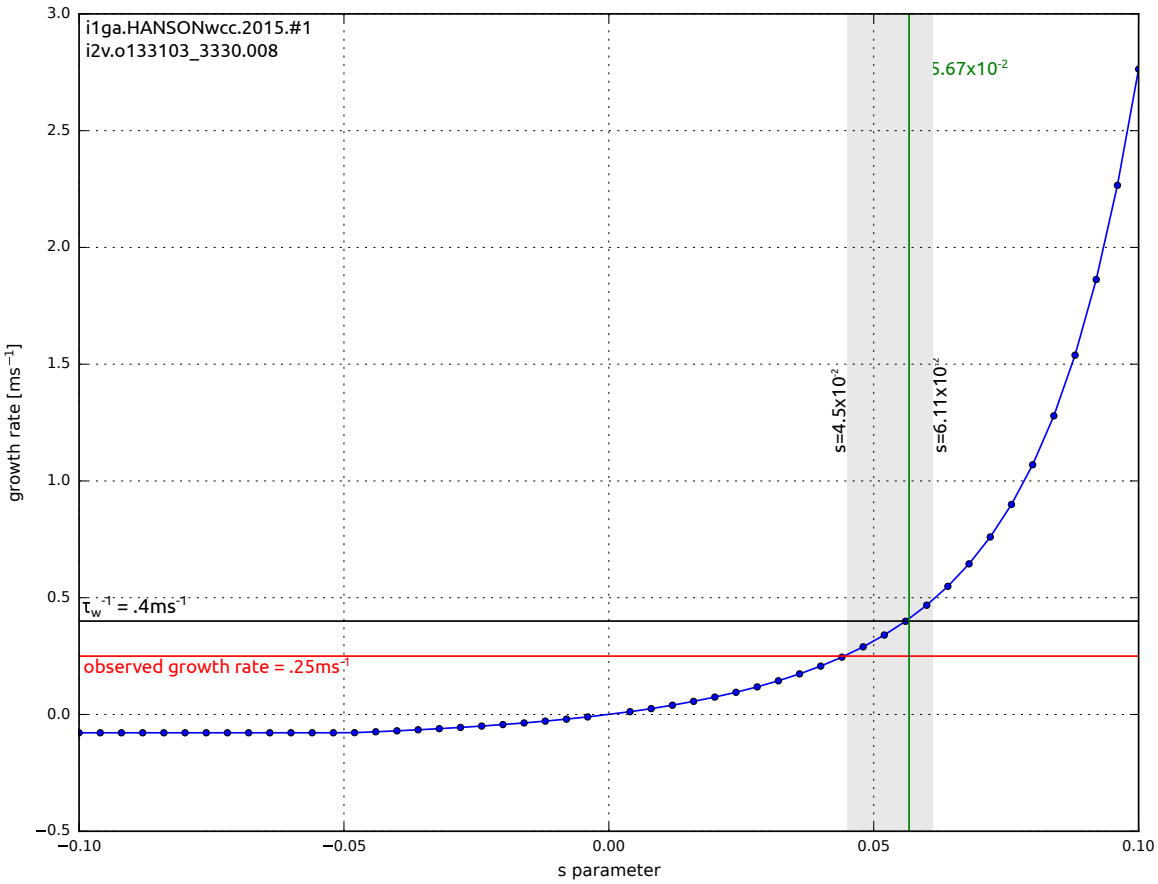

Figure 6.1: The growth rate, γ, of the mode modeled by VALEN as a function of s with $\alpha = 0$. The observed growth rate is determined in section 6.3. The curve shown in blue is the blue line shown in figure 3.3. The region highlighted in gray shows the range of s values used in the experiment described in section 7.1.

6.2 Model Reduction

Because VALEN is a large model (approximately 1400 states), it is not conducive to real-time control. Therefore, a singular perturbation approximation balanced realization (section 2.2.2) of VALEN is used for the LQG algorithm that runs on Tesibius (described in section 5.3). After a VALEN model has been produced by the FORTRAN executables and converted into state space form, the model is balanced and reduced[16][54]. Figure 6.2 shows the Hankel Singular Values (HSVs, defined in section 2.2.1) of a balanced realization

of a VALEN model up to the first 32 transformed states. When given code to execute by

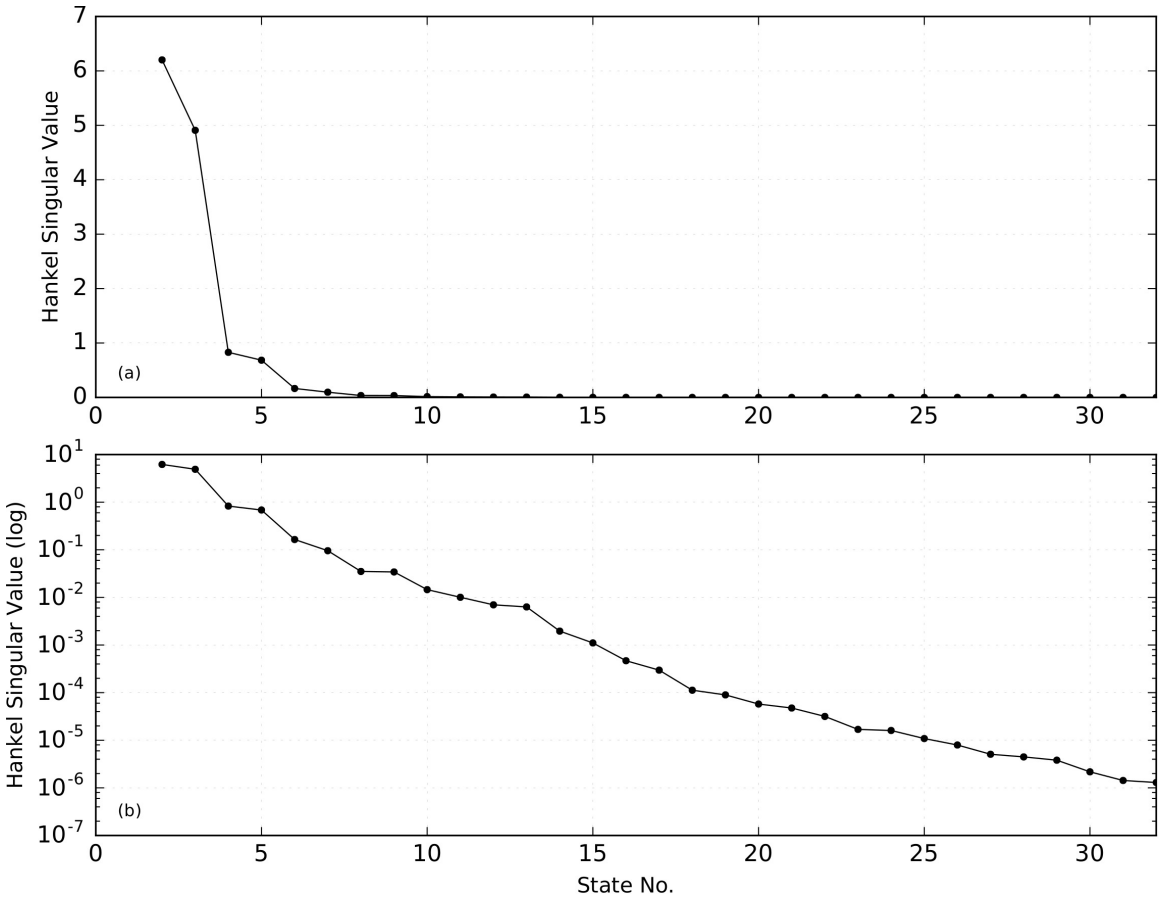

Figure 6.2: (a) plot of Hankel Singular Values (HSVs) for the first 32 states of the balanced realization version of the VALEN DIII-D model. (b) log of HSVs. Truncating the model at 32 states allows for HSVs up to 1×10^{-6} to be accounted for. This ensures a good approximation of the full VALEN model is retained in much fewer states.

many threads, the GPU partitions threads into groups of 32 threads, called warps, which are then scheduled for execution[55]. The balanced VALEN DIII-D model is truncated at 32 in order to align nicely with warps and thread scheduling and doing so allows for HSVs up to 1×10^{-6} to be accounted for. Truncation at a warp boundary allows for threads to diverge with little penalty on computation speed[55]. With the model reduced to a more computationally manageable size, observer and controller gains may be designed by the methods described in chapter 4.

6.3 Observer Design

The separation principle allows for separate design of the observer gain $\mathbf{K}_{\mathrm{o}}$ and controller gain $\mathbf{K}_{\mathrm{c}}$[41]. Estimates of the measurement noise, $\vec{v}$, and the process noise, $\vec{w}$, are required to design an optimal observer. Analysis of the observed dynamics of an RWM in DIII-D were used as the basis for estimating measurement noise. The dynamics of the $n = 1$ fit (appendix A) of the midplane poloidal field probes (MPID66M) for shot 154941, which terminated abruptly and prematurely due to an RWM, are shown in figure 6.3. The time series shown is during the early stages of the RWM and growth is assumed to be exponential. After taking the log of the $n = 1$ fit of MPID66M and doing a linear fit of the points, one standard deviation from the linear fit is taken as the standard deviation of the Gaussian measurement noise vector. One standard deviation for measurement noise is estimated at 4.86×10^{-5} Tesla. This value is then used to build a measurement noise covariance matrix and is also used for the added measurement noise for the simulations described in section 7.3. Tuning of the process noise standard deviation was done in simulation by gradually and incrementally increasing the scalar value, q_{o}, to make a covariance matrix, $\mathbf{Q}_{\mathrm{o}} = q_{\mathrm{o}}\mathbf{I}$ and identifying a value which makes the simulation unstable: that is to say, a point at which there is too much uncertainty in the system model to adequately do feedback. A value of 1×10^{-6} Amps is used for q_{o}. The process and measurement noise covariance matrices along with the system and output matrices, $\mathbf{F}$ and $\mathbf{C}$, are used to solve the Discrete Time Algebraic Riccati Equation (DARE) needed for calculating the observer gain, $\mathbf{K}_{\mathrm{o}}$[41].

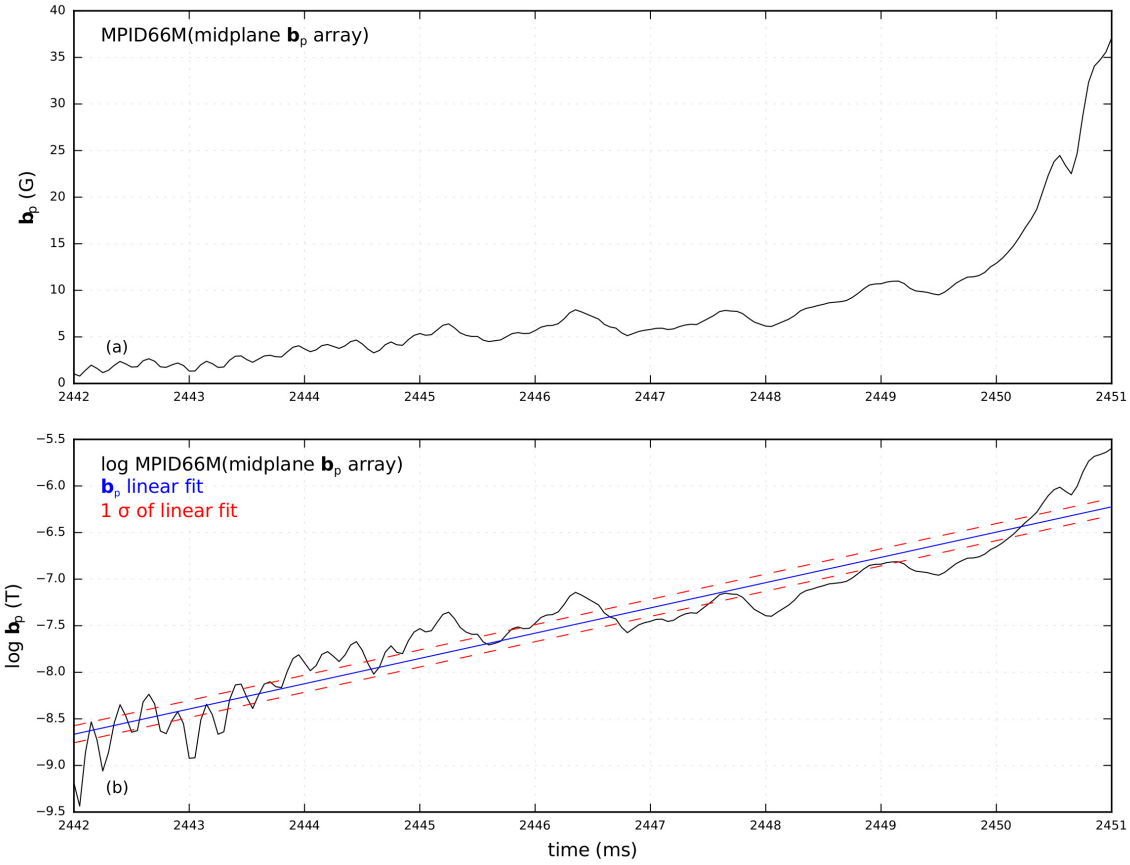

Figure 6.3: (a) $n = 1$ fit of MPID66M array (B_p toroidal midplane array) during shot 154941. (b) log of $n = 1$ fit (black), linear fit (blue), and one standard deviation from linear fit (red). One standard deviation of linear fit was used for covariance of measurement noise in Kalman filter and for noise in the simulations of section 7.3.

6.4 Controller Design

Design of the controller gain, $\mathbf{K}_c$, is determined by parameters in the LQG cost function,

$$\mathcal{J} = \frac{1}{2}\sum_{k=0}^{N}[\vec{x}_k^T\mathbf{Q}_c\vec{x}_k + \vec{u}_k^T\mathbf{R}_c\vec{u}_k]. \tag{6.1}$$

The selection of state and control penalty matrices, $\mathbf{Q}_c$ and $\mathbf{R}_c$, respectively, is a matter of trial and error[41]. Bryson's rule is used as a first step to identify reasonable values for

$\mathbf{Q}_\mathrm{c}$ and $\mathbf{R}_\mathrm{c}$[41]. Both penalties are required to be positive definite matrices, and thus they will be diagonal with positive elements. For $\mathbf{R}_\mathrm{c}$, the diagonal is selected to have elements like so,

$$R_{\mathrm{c}_{1,1}} = \frac{1}{u^2_{1_{\max}}}, R_{\mathrm{c}_{2,2}} = \frac{1}{u^2_{2_{\max}}}, ...R_{\mathrm{c}_{m,m}} = \frac{1}{u^2_{m_{\max}}}, \tag{6.2}$$

where $u_{m_{\max}}$ is the maximum control input that u_m may reasonably actuate. In the case of the amplifiers that command voltage to control coils, $u_{m_{\max}}$ is chosen to be 100 Volts. For $\mathbf{Q}_\mathrm{c}$, a similar approach is taken, where a new penalty is defined as, $\bar{\mathbf{Q}}_\mathrm{c}$, that is related to $\mathbf{Q}_\mathrm{c}$ as,

$$\mathbf{Q}_\mathrm{c} = \mathbf{H}^T \bar{\mathbf{Q}}_\mathrm{c} \mathbf{H}, \tag{6.3}$$

where $\mathbf{H}$ has the same dimensions as $\mathbf{C}$. This form is taken so that states enter the cost via the important outputs[41]. Therefore $\mathbf{H}$ is defined as,

$$H_{1,1} = 1, ...H_{n,n} = 1 \tag{6.4}$$

and all other elements are equal 0. This parameter n is changeable up to the maximum number of outputs. The diagonal of $\bar{\mathbf{Q}}_\mathrm{c}$ is chosen as,

$$\bar{Q}_{\mathrm{c}_{1,1}} = \frac{1}{y^2_{1_{\max}}}, \bar{Q}_{\mathrm{c}_{2,2}} = \frac{1}{y^2_{2_{\max}}}, ...\bar{Q}_{\mathrm{c}_{p,p}} = \frac{1}{y^2_{p_{\max}}}, \tag{6.5}$$

where $y_{p_{\max}}$ is the maximum allowable measurement deviation from measurement y_p and $y_{p_{\max}}$ is taken to be 1×10^{-3} Tesla (10 Gauss). The cost defined in equation 6.1 may be rewritten as,

$$\mathcal{J} = \frac{1}{2}\sum_{k=0}^{N} [\rho \vec{x}_k^T \mathbf{H}^T \bar{\mathbf{Q}}_\mathrm{c} \mathbf{H} \vec{x}_k + \vec{u}_k^T \mathbf{R}_\mathrm{c} \vec{u}_k], \tag{6.6}$$

where ρ is a scalar multiplier between state and control terms that may serve as a free parameter to be scanned in control development experiments. From the selection of $\mathbf{Q}_c$ and $\mathbf{R}_c$ along with the system and input matrices, $\mathbf{F}$ and $\mathbf{G}$, the DARE may be solved to give the controller gain, $\mathbf{K}_c$[41].

This chapter contains material which has been accepted for publication in *Control Engineering Practice* and material which has been submitted for publication in *Nuclear Fusion*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Atomic Energy Agency, 2017. The dissertation author was the primary investigator and author of these papers.

Chapter 7

System Identification and Simulations with VALEN

VALEN may used to simulate the effectiveness of RWM feedback schemes and algorithms. As discussed in chapter 3, VALEN has two free parameters, s and α, which determine the stability of the RWM. The growth rate of the mode is determined by s, the stability parameter, and the rotation rate by α, the torque parameter. In order to run simulation of control experiments, determination of VALEN's free parameters is necessary. These perturbation experiments are expected to yield a combination of s, and α, that gives a stable, damped RWM. This chapter summarizes the results of an evening experiment session that lasted approximately two hours in January 2016, the techniques used to determine VALEN's free parameters from these experiments and the simulation of experiments using the identified response model.

7.1 Controller Development

Experiments were conducted to develop and evaluate the performance of controller parameters used in the VALEN-based control algorithm. Because the RWM is nearly impossible to provoke on demand with a known growth and rotation rate, for this work it is necessary to actively stimulate the RWM using the internal I-coil set. The upper and lower I-coil set were paired with 240° phase difference to apply an n=1 field rotating at 20Hz and lasting 200ms. The goal of these perturbations is to drive an n=1 plasma response that has the same structure and dynamics of the RWM, in a plasma that is otherwise stable to the RWM. This perturbation is then measured by the control algorithm which then does feedback on the plasma response by driving the external C-coils. The intent of the phase difference in the upper and lower I-coils is to drive the plasma response so that it has the same "barber pole" style pitch as the normal magnetic field of a RWM as shown in figure 3.2. During a period of nearly constant plasma parameters, lasting approximately 1.4s, four such perturbations were applied. To establish a reference for comparison, a few shots were conducted without any feedback from the C-coils. A plot of raw signals from one such discharge is shown in figure 7.1 and shows the current in a single I-coil, a single B_p sensor and a single B_r sensor. The RWM is assumed to have a growth rate that grows on the order of the inverse of a wall time, τ_w, where τ_w is the inductance divided by the resistance, L/R, of DIII-D's wall. A generally accepted value of τ_w is 2.5ms, which corresponds to an RWM growth rate, γ_w, of 400s^{-1}. By keeping $\alpha = 0$, VALEN's stability parameter, s, can now be chosen to give an RWM with the desired growth rate. Four different values of s were used: 4.5×10^{-2}, 5.11×10^{-2}, 5.67×10^{-2} and 6.11×10^{-2}, which correspond to approximate growth rates of 250s^{-1}, 333s^{-1}, 400s^{-1} and 490s^{-1}, respectively.

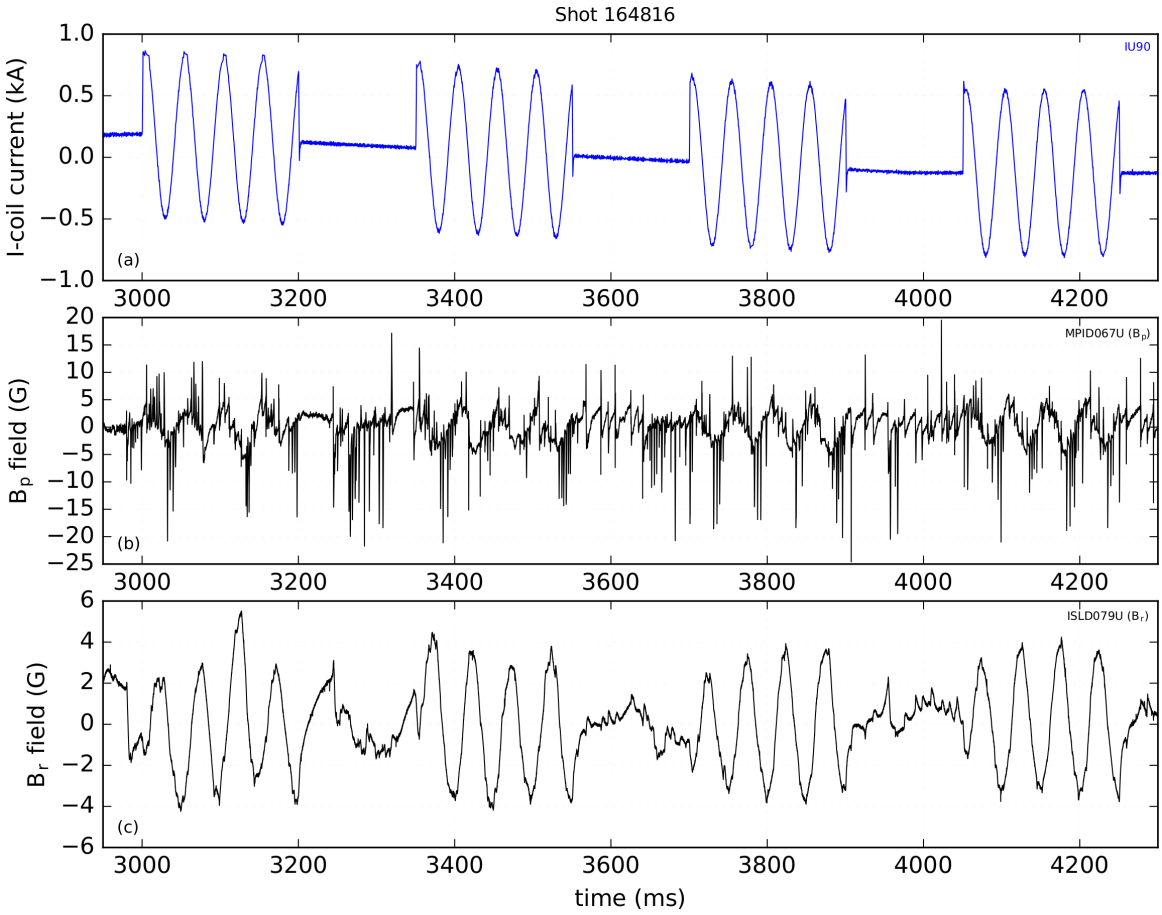

Figure 7.1: Raw signals from open loop controller development experiments with n=1 I-coil perturbation at 20Hz. These experiments are described in section 7.1. (a) Current in a single lead I-coil (IU90). (b) Poloidal (B_P) field in a single B_p sensor (MPID067U). (c) Radial (B_r) field in a single B_r sensor (ISLD079U). Perturbations by the I-coils cause perturbations to the plasma equilibrium in both the poloidal and radial directions.

This range of s is shown in gray in figure 6.1. This assumed growth rate is manifested as the real part of the most unstable eigenvalue of state-space VALEN's system matrix, $\mathbf{A}$. During each perturbation, a new controller based on one of the listed growth rates was used. In this manner the robustness of the VALEN model was investigated.

Figure 7.2 shows the fast Fourier transforms (FFTs) of the $n = 1$ fit (appendix A) of midplane poloidal field probes and the $n = 1$ fit of I-coil currents. The FFT of the midplane poloidal field probes shows that the dominant frequency of the plasma response is at 20Hz, the applied perturbation frequency. A meaningful figure of merit for these

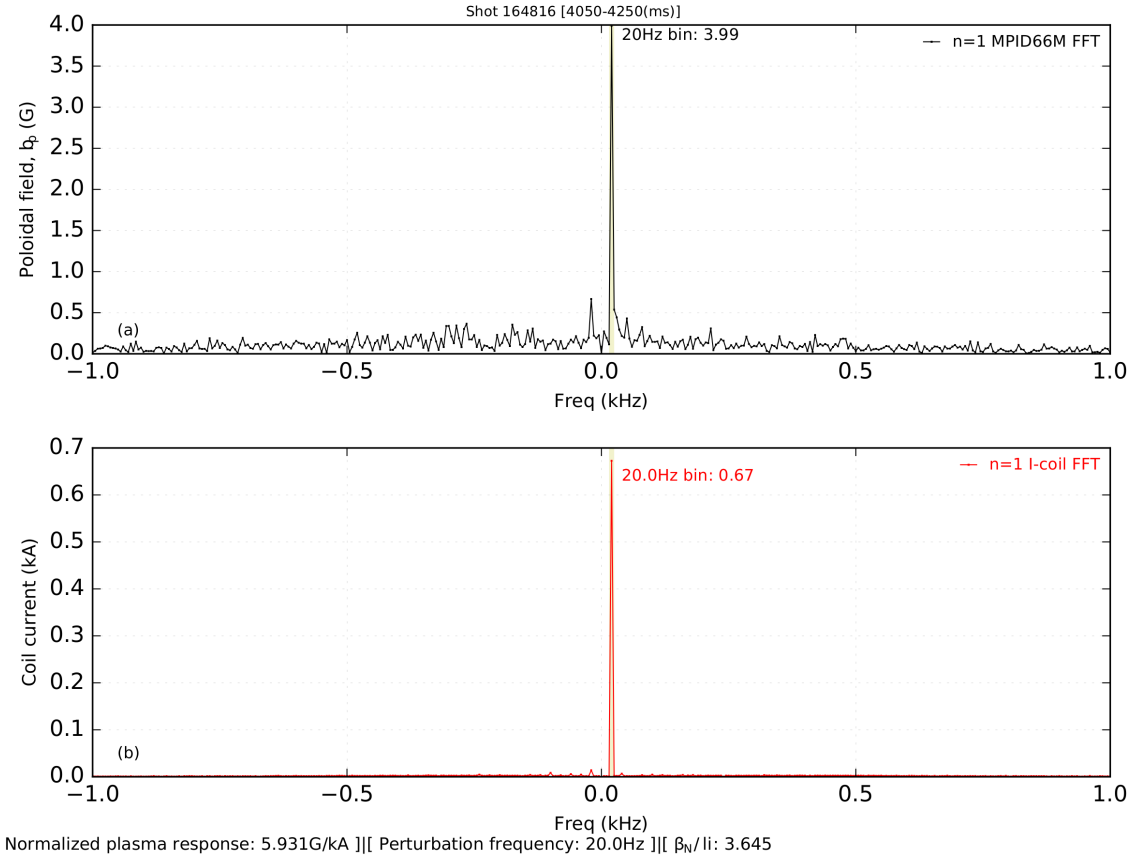

Figure 7.2: (a) Fast Fourier transforms of n=1 plasma response of MPID66M array (B_p poloidal midplane array); (b) n=1 I-coil perturbation. The highlighted region corresponds to 20Hz.

perturbation experiments and simulation is the normalized plasma response: that is to

say, the magnetics response at the perturbation frequency, divided by the I-coil current. For example, in figure 7.2 the normalized plasma response is calculated by dividing the 3.99G of poloidal field response at 20Hz by the .67kA of I-coil current at 20Hz, which gives a normalized plasma response of 5.9G/kA. The normalized plasma response was calculated for every I-coil perturbation cycle along with time averaged β_N over time averaged plasma inductance, li, for feedback-off (reference) shots and feedback-on shots. These normalized plasma responses versus β_N/li are plotted and shown in figure 7.6 for all open and closed loop discharges. Feedback reduced the average figure of merit between cases by approximately 32%. This reduction in figure of merit indicates that feedback is effective at dampening the plasma response to the applied perturbation. This method can be used to compare feedback algorithms controlling the external coils at various frequencies of the applied perturbation.

7.2 Determination of VALEN Parameters from Experiments

Magnetic measurements of the plasma response can be directly simulated using VALEN. A similar technique has been used to determine the s and α parameters in a single-mode response model on NSTX[56]. The python package `mpfit`[57], which implements the Levenberg-Marquardt[58][59] technique of least-squares minimization, was used to determine these parameters from experimental measurements of the driven, stable plasma response to applied n=1 perturbations.

7.2.1 Frequency Domain VALEN

The n=1 fits (appendix A) of the poloidal and radial field probes and the lead I-coils were Fourier transformed during the four 200ms perturbations of open loop shots (164816

and 164817, from January 2016 experiments),

$$\begin{bmatrix} B_p^c(t) \\ B_p^s(t) \\ B_r^c(t) \\ B_r^s(t) \end{bmatrix} = \mathbf{F}_{\text{VALEN}}\vec{y}(t), \quad \begin{aligned} B_p(\omega) &= \mathcal{F}\left(B_p(t) = B_p^c(t) + iB_p^s(t)\right) \\ B_r(\omega) &= \mathcal{F}\left(B_r(t) = B_r^c(t) + iB_r^s(t)\right) \end{aligned} \tag{7.1}$$

$$\begin{bmatrix} I^c(t) \\ I^s(t) \end{bmatrix} = \mathbf{F}_{ic} \begin{bmatrix} I_{\text{IU30}}(t) \\ I_{\text{IU90}}(t) \\ I_{\text{IU150}}(t) \end{bmatrix}, \quad I(\omega) = \mathcal{F}\left(I(t) = I^c(t) + iI^s(t)\right). \tag{7.2}$$

Where $\vec{y}(t)$ is a vector of poloidal and radial field magnetics measurements, $I_{\text{IUXX}}(t)$ is the time signal of currents in a particular lead I-coil and $\mathbf{F}_{\text{VALEN}}$ and $\mathbf{F}_{ic}$ are fitting matrices, defined in appendix A, used to determine the n=1 sine and cosine components of magnetics signals or I-coil currents. The perturbation frequency, 20Hz for these shots, was used to identify the applicable bins for the Fast Fourier Transforms (FFTs) of magnetics and lead I-coil currents. The ratio of these complex numbers is the normalized plasma response expressed in G/kA,

$$\begin{aligned} \tilde{B}_p(\omega^*) &= B_p(\omega^*)/I(\omega^*) \\ \tilde{B}_r(\omega^*) &= B_r(\omega^*)/I(\omega^*) \end{aligned}, \quad \begin{bmatrix} \tilde{B}_p^c(\omega^*) \\ \tilde{B}_p^s(\omega^*) \\ \tilde{B}_r^c(\omega^*) \\ \tilde{B}_r^s(\omega^*) \end{bmatrix} = \begin{bmatrix} \Re(\tilde{B}_p(\omega^*)) \\ \Im(\tilde{B}_p(\omega^*)) \\ \Re(\tilde{B}_r(\omega^*)) \\ \Im(\tilde{B}_r(\omega^*)) \end{bmatrix} = \tilde{y}_{\text{observed}}(\omega^*), \tag{7.3}$$

where ω^* indicates the perturbation frequency. The sine and cosine components of $\tilde{y}_{\text{observed}}$ in equation 7.3 are averaged for the eight perturbation cycles. For plots of normalized

plasma response, $|\tilde{B}_p(\omega^*)|$ or $|\tilde{B}_r(\omega^*)|$ is used.

The VALEN state-space equations 3.39 are Fourier transformed as follows:

$$\mathcal{F}\left(\frac{d}{dt}\vec{I}(t) = \mathbf{A}\vec{I}(t) + \mathbf{B}\vec{V}(t)\right) \tag{7.4a}$$

$$i\omega\vec{I}(\omega) = \mathbf{A}\vec{I}(\omega) + \mathbf{B}\vec{V}(\omega) \tag{7.4b}$$

$$\begin{bmatrix}\vec{I}_w(\omega)\\ \vec{I}_f(\omega)\\ \vec{I}_d(\omega)\end{bmatrix} = \vec{I}(\omega) = (\mathbf{I}i\omega - \mathbf{A})^{-1}\mathbf{B}\vec{V}(\omega), \tag{7.4c}$$

where $\vec{V}(\omega)$ and $\vec{I}(\omega)$ are now phasors and functions of the perturbation frequency, 20Hz, and $\mathbf{I}$ is the identity matrix. During experiments, the SPAs were used to command 675A at 20Hz through the lead coils and phase shifted by 60°in adjacent coils. Because VALEN's inputs are voltages on each control coil and not current on the coil, in order to simulate the I-coil perturbation, an appropriate amplitude for the voltage phasor must be found to give the correct amplitude for the current phasor. A routine using the Levenberg-Marquardt technique of least-squares minimization was used to find this voltage amplitude using the form of equation 7.4c, except that the matrices $\mathbf{A}$ and $\mathbf{B}$ are constructed from the vacuum matrices $\mathbf{L}$ and $\mathbf{R}$ generated by the VALEN pre-processor CUPR13. This is essentially the finite element model of conducting structures in vacuum, i.e., with no plasma. In equation 7.4c, $\vec{V}(\omega)$ takes the form

$$\vec{V}(\omega) = -V\begin{bmatrix}e^{-i\frac{\pi}{6}}\\ e^{-i\frac{\pi}{2}}\\ e^{-i\frac{5\pi}{6}}\end{bmatrix}, \tag{7.5}$$

where V is the amplitude from least squares fitting. The output measurement of Fourier

transformed state-space VALEN is now

$$\begin{bmatrix} B_p(\omega) \\ B_r(\omega) \end{bmatrix} = \mathbf{F}_{\text{VALEN}}\vec{y}(\omega) = \mathbf{F}_{\text{VALEN}}\mathbf{C}\vec{I}(\omega), \tag{7.6}$$

where $\vec{I}(\omega)$ is found from equation 7.4c and $\mathbf{F}_{\text{VALEN}}$ is the complex version of the fitting matrix from equation 7.1. The I-coil currents may be extracted from the phasor $\vec{I}(\omega)$ like so:

$$I_{ic}(\omega) = \mathbf{F}_{ic}\vec{I_f}(\omega), \tag{7.7}$$

$$\begin{aligned} \tilde{B}_p(\omega) &= B_p(\omega)/I_{ic}(\omega) \\ \tilde{B}_r(\omega) &= B_r(\omega)/I_{ic}(\omega), \end{aligned} \tag{7.8}$$

where $\mathbf{F}_{ic}$ is the complex version of the fitting matrix from equation 7.2 and $\vec{I_f}(\omega)$ is determined from equation 7.4c. Now the normalized plasma response for Fourier transformed state-space VALEN is calculated as:

$$\begin{bmatrix} \tilde{B}_p^c(\omega^*) \\ \tilde{B}_p^s(\omega^*) \\ \tilde{B}_r^c(\omega^*) \\ \tilde{B}_r^s(\omega^*) \end{bmatrix} = \begin{bmatrix} \Re(\tilde{B}_p(\omega)) \\ \Im(\tilde{B}_p(\omega)) \\ \Re(\tilde{B}_r(\omega)) \\ \Im(\tilde{B}_r(\omega)) \end{bmatrix} = \tilde{y}_{\text{VALEN}}(\omega^*). \tag{7.9}$$

With observed measurements from experiments (equation 7.3), measurements from Fourier transformed VALEN (equation 7.9) and an estimate for measurement error, a chi-squared distribution can be used to test the goodness of fit between observed data

and a VALEN model composed of a given s and α. The chi-squared is given as:

$$\chi^2 = \sum_{i=1}^{m} \frac{(\tilde{y}_{\text{observed}}(\omega)_i - \tilde{y}_{\text{VALEN}}(\omega)_i)^2}{\sigma^2}, \tag{7.10}$$

where σ^2 is the variance in the measurement error. Here, the variance is estimated by averaging the quantities in the Fourier bins adjacent to the perturbation frequency. The Levenberg-Marquardt algorithm will iterate over s and α in order to minimize the χ^2 value. In theory, the routine converges and terminates when no changes in the model parameters can decrease χ^2 any further. The i2 file used to generate VALEN during this iterative process is based on a `DCON` mode pattern plot (section 3.1.5) from a similar plasma equilibrium. Least squares fitting converged on values of s and α of 2.433×10^{-2} and -1.255×10^{-1}, respectively. These values correspond to the red triangles shown in figures 3.3 and 3.4.

7.3 Simulation

Using the converged upon values of s and α, a VALEN model can be built to simulate experiments. This model will henceforth be referred to as “the plant”, i.e., the system to be controlled, as is the standard in control engineering terminology. This model is converted from continuous to discrete time and the plant is run for 200ms with a sample time of 13μs, as is done in practice.

7.3.1 Open Loop Simulation

First, the open loop case, i.e., with no feedback, is described. The simulation is run as a for loop using the discrete time model:

$$\begin{aligned} \vec{x}_f &= \mathbf{A}'\vec{x}_p + \mathbf{B}'\vec{i}_k \\ \vec{y}_k &= \mathbf{M}\vec{x}_f + \vec{\nu} \\ \vec{x}_p &= \vec{x}_f, \end{aligned} \tag{7.11}$$

where the variable $\vec{\nu}$ is zero-mean, white, Gaussian noise as determined in section 6.3. The matrices $\mathbf{A}'$ and $\mathbf{B}'$ are the 13μs discrete time versions of matrices $\mathbf{A} = \mathbf{L}_{\text{eff}}^{-1}\mathbf{R}$ and $\mathbf{B} = \mathbf{L}_{\text{eff}}^{-1}$ where $\mathbf{L}_{\text{eff}}$ and $\mathbf{R}$ are from equation 3.38a. $\mathbf{B}'$ is the input matrix for the I-coils. $\mathbf{M}$ is from equation 3.38b. Variables $\vec{x}_f$ and $\vec{x}_p$ are the future and past states of the plant and are initially zero at the start of simulations. The subscript k indicates the for loop index, and for 200ms discretized at 13μs, there are 15385 indexes in the loop. The variable $\vec{i}_k$ is the sinusoidal voltage waveform for the I-coil perturbation and is a 3 by 15385 matrix array. The variable $\vec{y}_k$ is a 24 by 15385 array used to store measurements at each time step for later plotting. Open loop simulation results are shown in figure 7.3. Normalized plasma response is calculated in the same manner as described in section 7.1 using the measurements stored in $\vec{y}_k$. Plots of the Fourier transformed signals for the open loop simulation are shown in figure 7.4.

7.3.2 Closed Loop Simulation

Closed loop simulations were also done to evaluate the effectiveness of feedback schemes employing the C-coils. In the simulation, the outputs of the plant are used

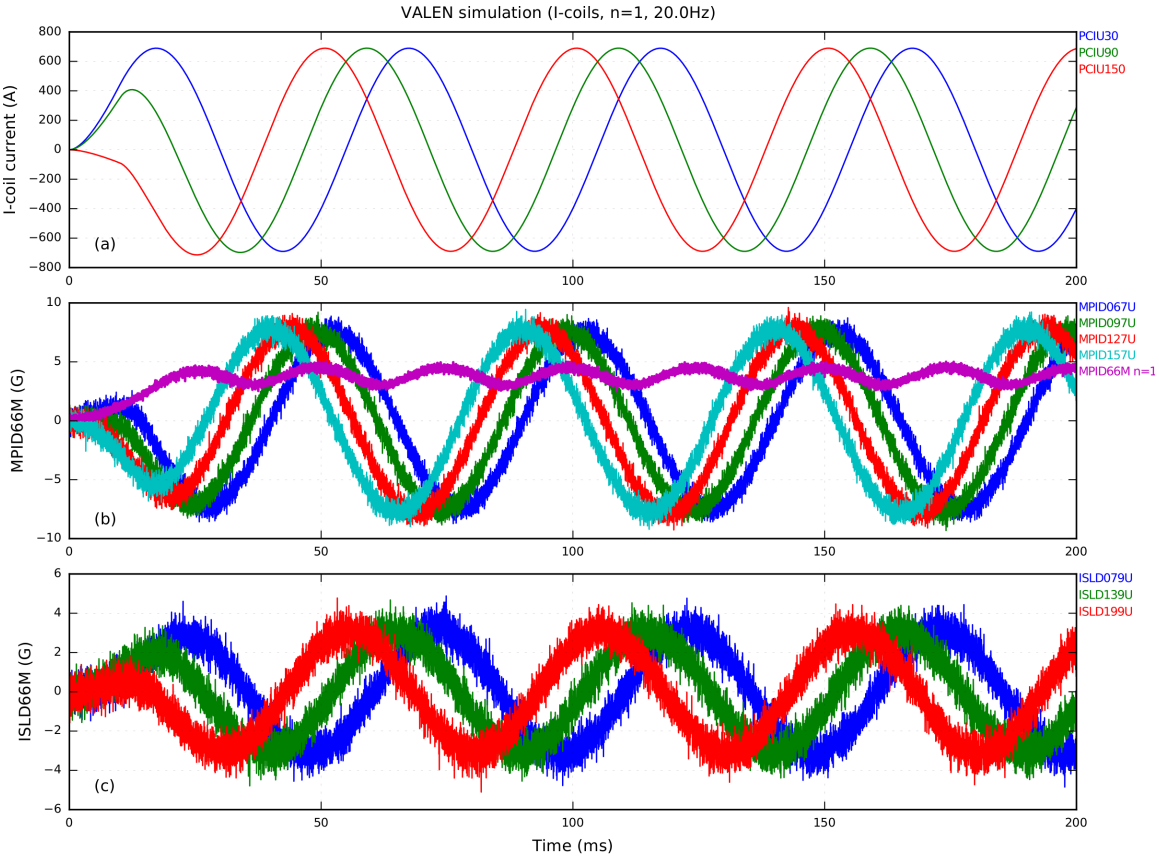

Figure 7.3: Open loop simulation of controller development experiments with n=1 I-coil perturbation at 20Hz. (a) lead I-coil currents (A). (b) midplane B_p sensors. (c) midplane B_r sensors. These experiments are described in section 7.1. This figure compares open loop simulation results to experimental results shown in figure 7.1.

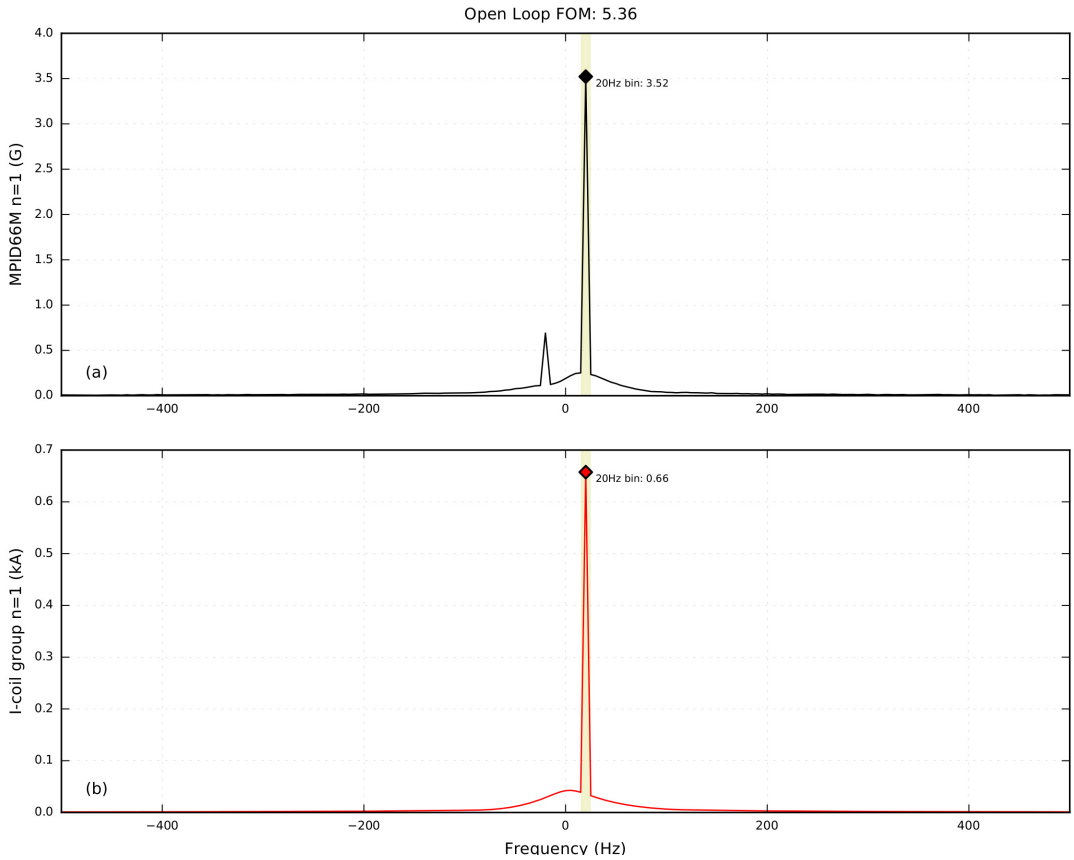

Figure 7.4: (a) Fourier transformed signals of the n=1 fits of the midplane B_p (MPID66M) array. (b) FFT of $n = 1$ fit of upper I-coils. The normalized plasma response is the value of the black diamond divided by the red diamond.

to inform the feedback signal to the C-coils. Two types of feedback algorithms were used for simulations. The current method for RWM feedback in use by the DIII-D PCS is proportional gain control. The algorithm is a variation of the widely used Proportional Integral Derivative (PID) controller, with some smoothing filters incorporated. The transfer function for this PID controller is written as,

$$\vec{u}(s) = \frac{1}{1+s\tau_p}\left(G_p + G_i\frac{1}{1+s\tau_i} + G_d\frac{s\tau_d}{1+s\tau_d}\right)\mathbf{F}\vec{y}(s), \tag{7.12}$$

where G_p, G_i and G_d are the proportional, integral and derivative gains, respectively, and τ_p, τ_i and τ_d are their respective filter time constants[60]. $\vec{y}$ is a vector of magnetics signals and $\mathbf{F}$ is a fit matrix that maps poloidal field sensors to control coils and is based on the discharge's helicity. This matrix enables the PID algorithm to be coupled with a MIMO system. To implement this in discrete time, equation 7.12 is z-transformed to give the following:

$$\vec{z}^p_k = c_{p1}(\mathbf{F}\vec{y}_k + \mathbf{F}\vec{y}_{k-1}) - c_{p2}\vec{z}^p_{k-1}, \tag{7.13a}$$

$$\vec{z}^i_k = c_{i1}(\vec{z}^p_k + \vec{z}^p_{k-1}) - c_{i2}\vec{z}^i_{k-1}, \tag{7.13b}$$

$$\vec{z}^d_k = c_{d1}(\vec{z}^p_k - \vec{z}^p_{k-1}) + c_{d2}\vec{z}^d_{k-1}, \tag{7.13c}$$

where k denotes the time index. Filter coefficients are defined as,

$$c_{p1} = \frac{\delta t/(2\tau_p)}{\delta t/(2\tau_p)+1}, \qquad c_{p2} = \frac{\delta t/(2\tau_p)-1}{\delta t/(2\tau_p)+1}, \tag{7.14a}$$

$$c_{i1} = \frac{\delta t/(2\tau_i)}{\delta t/(2\tau_i)+1}, \qquad c_{i2} = \frac{\delta t/(2\tau_i)-1}{\delta t/(2\tau_i)+1}, \tag{7.14b}$$

$$c_{d1} = \frac{1}{\delta t/(2\tau_d)+1}, \qquad c_{d2} = \frac{\delta t/(2\tau_d)-1}{\delta t/(2\tau_d)+1}, \tag{7.14c}$$

where δt is the sample time. Finally, the control input is computed as

$$\vec{u}_k = G_p \vec{z}_k^p + G_i \vec{z}_k^i + G_d \vec{z}_k^d. \tag{7.15}$$

Equations 7.13a-7.13c are simple, single pole low-pass filters. Typical values used by the PCS for G_p range from 20-80 V_{com}/V_s, where V_s is the digitizer voltage measurement of the field and V_{com} is the commanded voltage. This method uses only the magnetics signals from the MPID66M (mid-plane poloidal field) array, meaning that $\vec{y}_k$ of equation 7.13a is composed only of B_p probes. For proportional gain only control, G_i and G_d are set to zero. A range of proportional gains was used in simulation ranging from 200 to $700V_{com}/V_s$ and τ_p=0.05ms. Simulations may be done with either current or voltage control VALEN models for the plant, as described in section 3.2, for either PID or LQG control. Design of the LQG observer and controller are described in chapter 6 and are based on an unstable VALEN model that has been reduced in system dimension using model reduction techniques (section 2.2). This controller model is not the same model as the plant, as they have differing values of s and α, but are built from identical equilibria, meaning the 3D structure of the mode is identical. Similarly to the open loop simulation,

the closed loop simulation is run in a for loop as:

$$
\begin{aligned}
&\vec{x}_f = \mathbf{A}'\vec{x}_p + \mathbf{B}'\vec{i}_k + \mathbf{B}''\vec{u} \\
&\vec{y}_k = \vec{y} = \mathbf{M}\vec{x}_f + \vec{\nu} \\
&\hline
&\text{\#PID feedback} \qquad\qquad\qquad\qquad\qquad\qquad \text{\#LQG feedback} \\
&\vec{e}_f = \mathbf{F}\vec{y}, \qquad\qquad\qquad\qquad\qquad\qquad \hat{x}_f = \mathbf{\Phi}\hat{x}_p + \mathbf{K}_\mathrm{o}\vec{y} \\
&\vec{z}^p_f = c_{p1}(\vec{e}_f + \vec{e}_p) - c_{p2}\vec{z}^p_p, \qquad\qquad \vec{u}_k = \vec{u} = -\mathbf{K}_\mathrm{c}\hat{x}_f \\
&\vec{z}^i_f = c_{i1}(\vec{z}^p_f + \vec{z}^p_p) - c_{i2}\vec{z}^i_p, \qquad\qquad \hat{x}_p = \hat{x}_f \\
&\vec{z}^d_f = c_{d1}(\vec{z}^p_f - \vec{z}^p_p) + c_{d2}\vec{z}^d_p \\
&\vec{u}_k = \vec{u} = G_p\vec{z}^p_f + G_i\vec{z}^i_f + G_d\vec{z}^d_f \\
&\vec{e}_p = \vec{e}_f, \;\; \vec{z}^p_p = \vec{z}^p_f, \;\; \vec{z}^i_p = \vec{z}^i_f, \;\; \vec{z}^d_p = \vec{z}^d_f \\
&\hline
&\vec{x}_p = \vec{x}_f,
\end{aligned}
\tag{7.16}
$$

where variable $\vec{u}$ is the voltage feedback command to the C-coil and $\mathbf{F}$ is the fit matrix of equations 7.12 and 7.13a. The vectors $\vec{e}$ and $\vec{z}$ are from equations 7.13a-7.13c. The scalars c are from equations 7.14a-7.14c. $\mathbf{B}''$ is the plant's input matrix for the C-coils. The matrices $\mathbf{K}_\mathrm{c}$ and $\mathbf{K}_\mathrm{o}$ are the controller and observer gains built from the unstable, reduced VALEN controller model described by $\mathbf{A}'_r$, $\mathbf{B}''_r$ and $\mathbf{M}_r$. $\mathbf{\Phi}$ is the closed loop dynamics matrix, $\mathbf{A}'_r - \mathbf{B}''_r\mathbf{K}_\mathrm{c} - \mathbf{K}_\mathrm{o}\mathbf{M}_r$. Variables $\hat{x}_f$ and $\hat{x}_p$ are the future and past observer state estimates and are initialized as zero vectors. The variable $\vec{u}_k$ is a 3 by 15385 array used to store the input at each time step. All other variables are the same as in equation 7.11. For the closed loop cases, the normalized plasma response is also calculated as described in section 7.1 using $\vec{y}_k$ and I-coil currents extracted from the state, $\vec{x}_p$. For the feedback voltages calculated by the feedback algorithm, the commanded voltage and current in each lead

coil can be studied to ensure it is within the capability of the power supply driving the coil. An example of coil feedback, along with current, as extracted from VALEN's state, for LQG voltage control feedback is shown in figure 7.5.

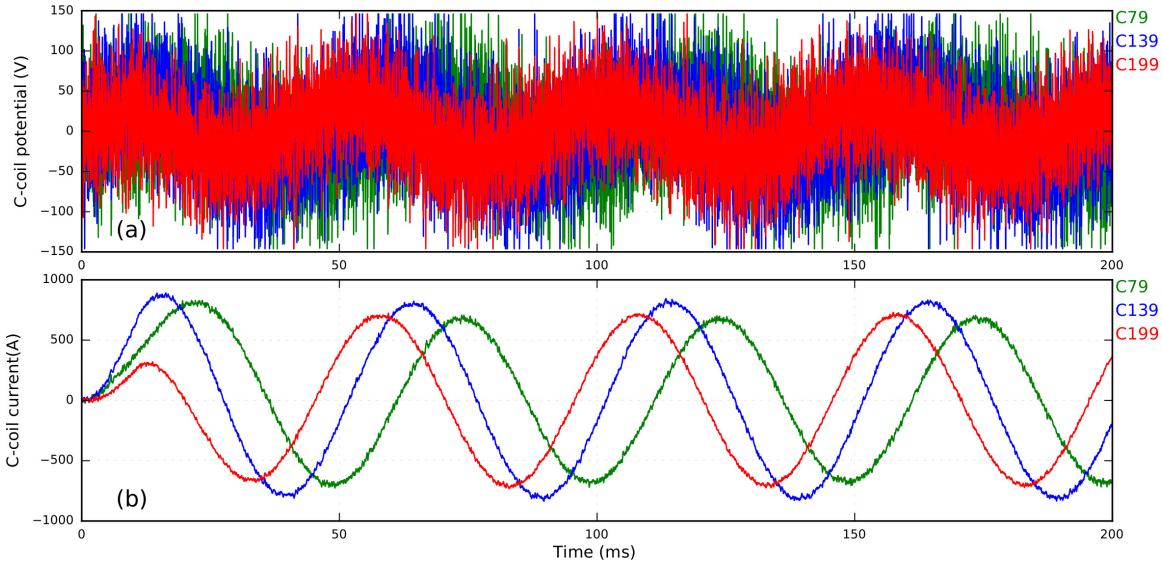

Figure 7.5: Voltage control closed loop feedback coil voltage (commanded) and current (extracted from VALEN state) for 20Hz I-coil perturbation. C79, C139 and C199 are the lead C-coils in each C-coil pair. (a) C-coil voltage command in each lead coil (V). (b) C-coil current in each lead coil (A). The C-coil feedback identifies the perturbation frequency and suppresses the plasma response.

7.3.3 Comparison with Control Development Experiments

Open and closed loop simulations done at 20Hz can be directly compared to experiments done with 20Hz perturbations (discharges 164816-164819, from January 2016). The LQG controller used for simulation was identical in design to one used in discharges 164818 and 164819, described in section 7.1. This controller used VALEN parameters s=5.67e-2 and α=0 with controller parameters n=5, $\rho=1\times10^{-3}$, t_{set}=70ms, $q_c=1\times10^{6}$ and $r_c=1\times10^{-4}$. The closed loop simulation of normalized plasma response well replicates the experimental results and are shown in figure 7.6. VALEN does not simulate

β_N divided by plasma inductance, β_N/li; therefore, the simulation result of normalized plasma response spans the entire horizontal axis.

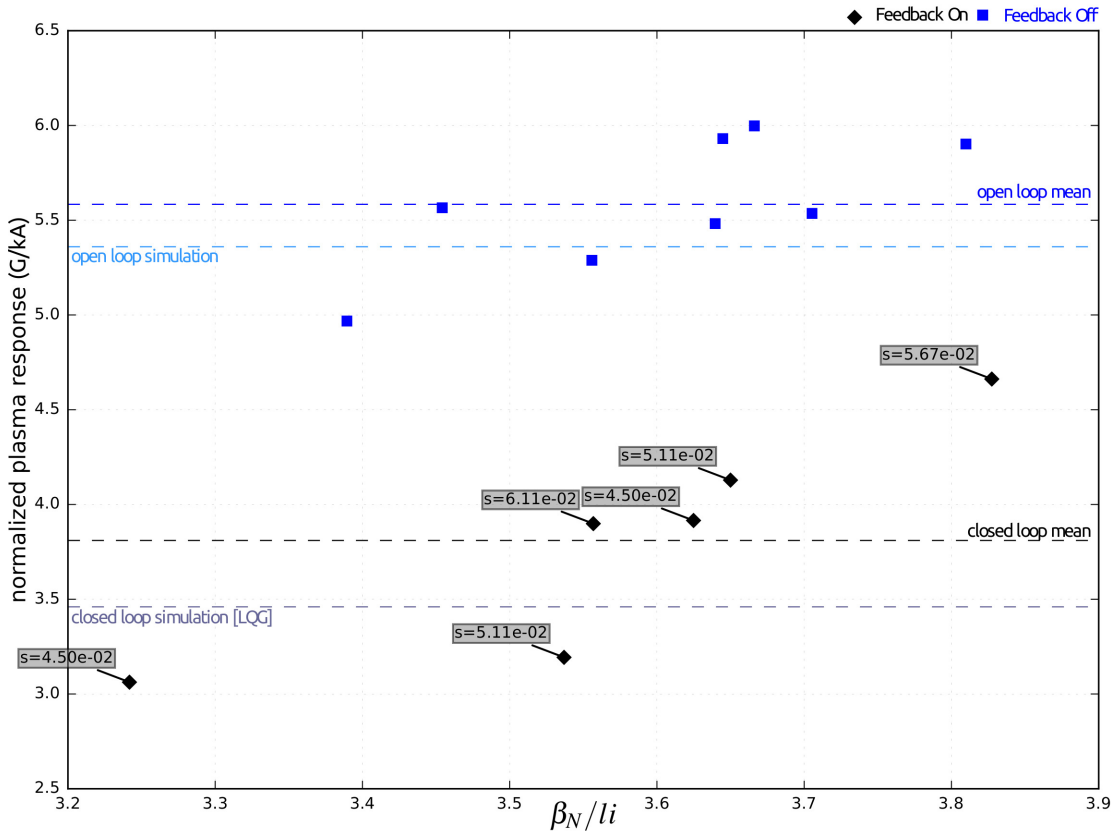

Figure 7.6: Comparison of open and closed loop simulations results with control development experiments (164816-164819) with 20Hz I-coil perturbation. Simulations described in sections 7.3.1 and 7.3.2 well replicate the experimental results of section 7.1. VALEN does not simulate β_N divided by plasma inductance, β_N/li; therefore, the simulation result of normalized plasma response spans the entire horizontal axis.

7.3.4 Frequency Response Simulations

Because of the good correlation between experiment and simulation seen in section 7.3.3 and figure 7.6, this technique was used to scan controller parameters to find a controller that minimized closed loop plasma response while commanding voltages

and currents that are within the capability of coil power supplies. The controller used here and in experiments from February 2017 (section 8.1), was designed in this way, largely by trial and error methods. This controller used VALEN parameters α=0 and s=5.667e-2, with controller parameters, n=18, $\rho=3\times10^{-3}$, t_{set}=70ms, $q_c=1\times10^{6}$ and $r_c=1\times10^{-4}$. The VALEN model used for the controller was also built from the same i2 file as the VALEN model used to simulate the plant. Proportional controllers may be tested in a similar fashion, with proportional gain of 200V_{com}/V_s and τ_p=0.05ms, as shown in figure 7.7. Open and closed loop simulations were run at several perturbation frequencies, and in this way, the performance of each type of controller can be evaluated against the open loop case. Results of these simulations for positive and negative frequencies are shown in figure 7.7. A negative frequency indicates that the helicity of the perturbation is reversed. Simulations done with large proportional gain ($\geq$ 500V_{com}/V_s) reveals a system resonance in the -30 to -40Hz range, as shown by the magenta line in figure 7.7. The cause for this resonance is not completely understood, but may have some connection to the selection of filter parameters in equations 7.14a.

7.3.5 Current Control Simulations

No experiments were done with LQG current control, but simulation results are included to validate the current control modifications that may be done to the VALEN model described in section 3.2.2. For the LQG current controller used in simulation, VALEN parameters were α=0 and s=9.e-2, with controller parameters, n=24, $\rho=1\times10^{-3}$, t_{set}=70ms, $q_c=1\times10^{6}$ and $r_c=4.4\times10^{-7}$. Results of these simulations are shown in figure 7.8. Current control feedback is capable of suppressing plasma response, but is not as effective as voltage control feedback simulations (section 7.3.2). This is likely due to the slow internal

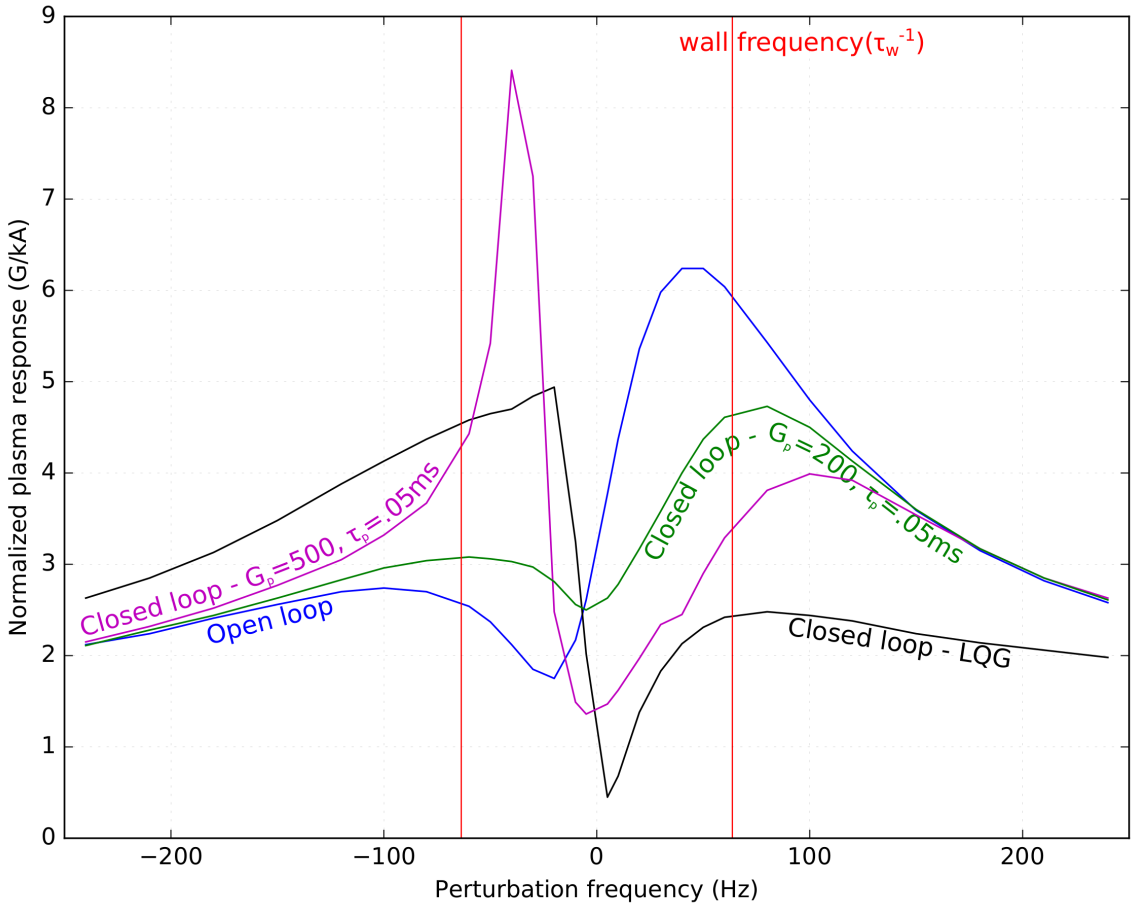

Figure 7.7: Simulation of both open loop and voltage control closed loop feedback using LQG and proportional gain over several perturbation frequencies. The large proportional gain used, $500V_{com}/V_s$, shown in magenta, reveals a resonant frequency between -30 and -40Hz. These simulations reveal that all feedback algorithms are most effective in all cases at lower positive frequencies, where the low-pass filtering effects of the vacuum vessel wall are less severe.

PID control loop inside the current control amplifier (section 5.2.1) which controls current.

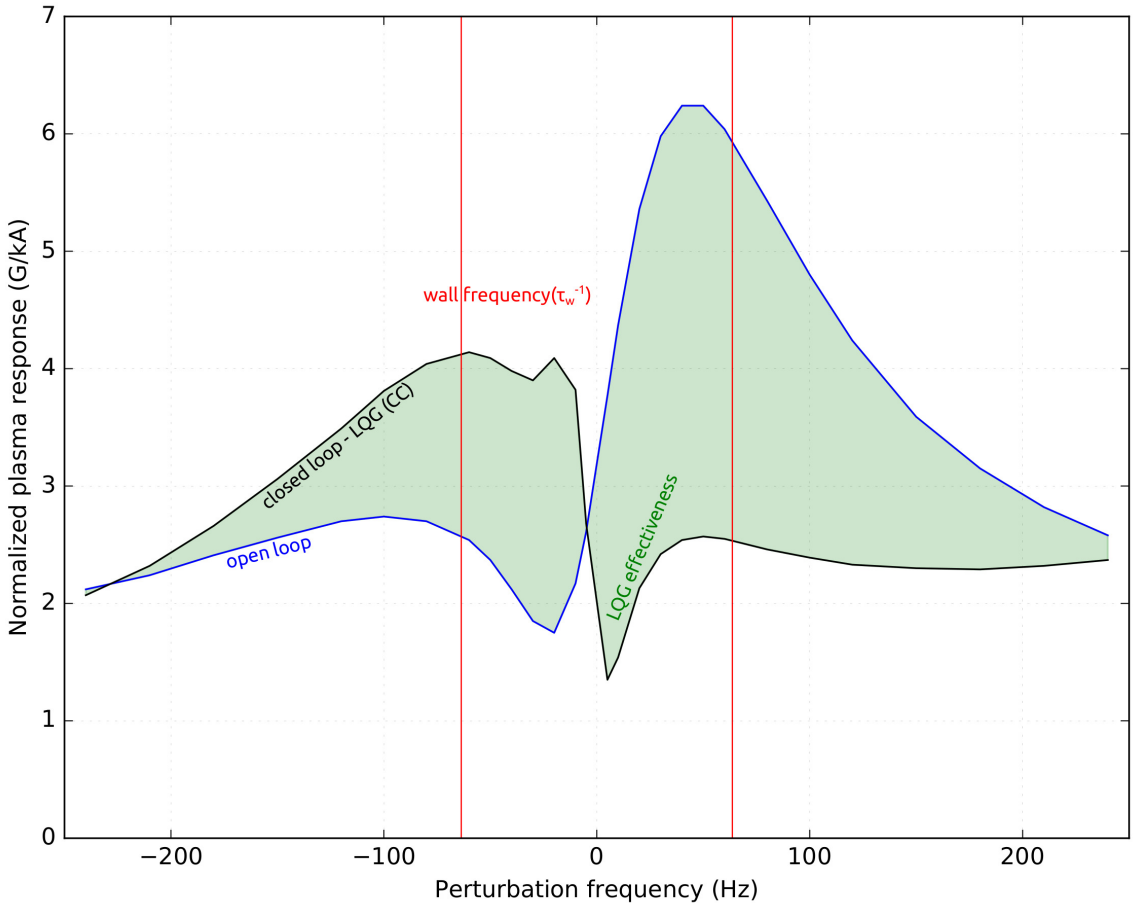

Figure 7.8: Simulation of both open loop and closed loop feedback using LQG current control over several perturbation frequencies. Current control feedback is capable of suppressing plasma response, but is not as effective as voltage control feedback. This is likely due to the slow internal PID control loop inside the current control amplifier (section 5.2.1) which controls current.

This chapter contains material which has been accepted for publication in *Control Engineering Practice* and material which has been submitted for publication in *Nuclear Fusion*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim;

Navratil, Gerald., International Atomic Energy Agency, 2017. The dissertation author was the primary investigator and author of these papers.

Chapter 8

Experimental Results

This chapter summarizes the results of experiments from March 2016 and February 9th and 13th 2017. The February 2017 experiments were dedicated solely to supporting this dissertation. In the March 2017 experiment, feedback using the VALEN-based LQG algorithm was done in "piggy back".

8.1 Frequency Scan Experiments

Using the same experimental technique described in section 7.1, the effect of the frequency of the I-coil perturbations was investigated. The target discharge was also tailored to allow for up to six perturbations during flat-top vice four used in previous experiments. Also, during the interim between experiment days, simulations of this experiment were conducted in VALEN, as described in chapter 7, to design a more optimized controller. This optimized LQG controller was then used to attempt C-coil control of perturbation experiments. For comparison purposes, a proportional controller using DIII-D's plasma control system (PCS) was used for feedback as well. Values for G_p and τ_p were 100V/V and

0.05ms, respectively. A true PID controller designed by Dr. Erik Olofsson and using system identification techniques was also used for comparison. An interesting characteristic of the proportional controller feedback was that the controller often applied significant C-coil current in the -30 to -40Hz range regardless of the I-coil perturbation frequency. Furthermore, more often than not, this application of current at what is suspected to be a resonant frequency or harmonic of the system drove more plasma response than the I-coil perturbation itself. An example of this phenomenon is shown in figure 8.1. The

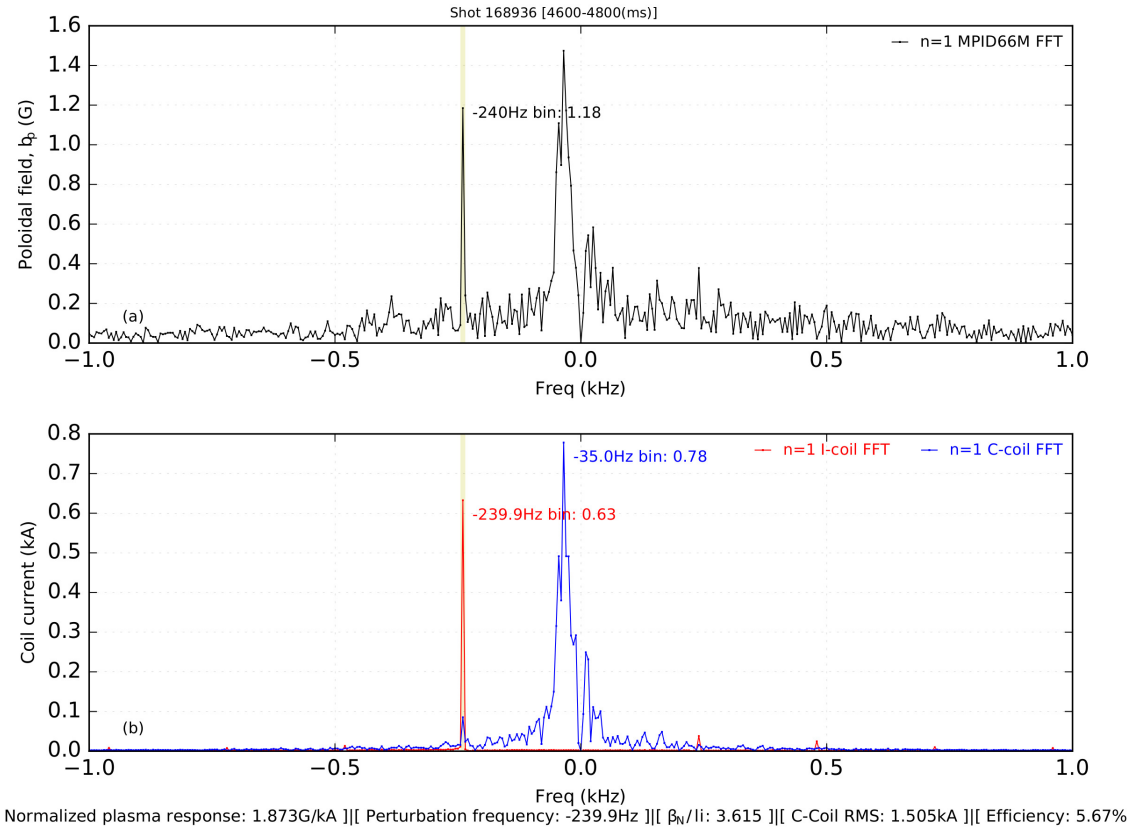

Figure 8.1: (a) Fast Fourier transforms of n=1 plasma response of MPID66M array (B_p poloidal midplane array). (b) n=1 I-coil perturbation and n=1 C-coil feedback for proportional only controller. Proportional controller applies significant current at -35Hz (spike in blue), which is not the I-coil perturbation frequency of -240Hz (spike in red). As a result, the C-coils drive more plasma response at -35Hz than the I-coils at -240Hz. The proportional controller drove more plasma response at the resonant frequency than at the perturbation frequency in over half cases.

bottom plot of figure 8.1 shows the I-coil perturbation current at -240Hz (red trace) which is also reflected by the spike at -240Hz in the top plot showing the $n = 1$ poloidal field (B_p) response. Figure 8.1 also shows the C-coil feedback current (blue trace) is greater at -35Hz than at -240Hz, which causes a spike in plasma response at -35Hz as shown in the top plot. The spike in plasma response at -35Hz is greater than at -240Hz. This resonance was also predicted by VALEN simulations with high proportional gain (G_P=500V_{com}/V_s), as shown in figure 7.7. A more conservative calculation of normalized plasma response may also be used in which the measured plasma response at the frequency of maximum plasma response is used, even if that differs from the measurement at the perturbation frequency (as used in section 7.1). This method is used in the discussion that follows and is denoted as *max plasma response* in the figures.

The normalized plasma response versus perturbation frequency is shown in figure 8.2 for open loop (no feedback) and closed loop (with feedback) discharges. The points highlighted above the entire data set have been filtered for $3.7 \leq \beta_N/li < 3.8$. The filtering on time-averaged β_N/li is done in order to isolate and compare similar discharges, as normalized plasma response is proportional to β_N/li. The vertical red lines in figure 8.2, and labeled *wall frequency*, are the angular frequency inverse wall times, $\pm 63.7\text{Hz} = 1/(\tau_w 2\pi)$, where $\tau_w = 2.5$ms. At frequencies above this value, the flux from the C-coils is effectively screened out because the vacuum vessel wall acts as a low-pass filter and thus feedback at higher frequencies is not expected to have as much of an impact on suppressing plasma response as it does at lower frequencies. LQG feedback was most successful at suppressing plasma response at frequencies below the wall frequency as can be seen by the trend of black diamonds in figure 8.2. Open loop and LQG closed loop simulation results, using VALEN parameters fit from experiments described in section

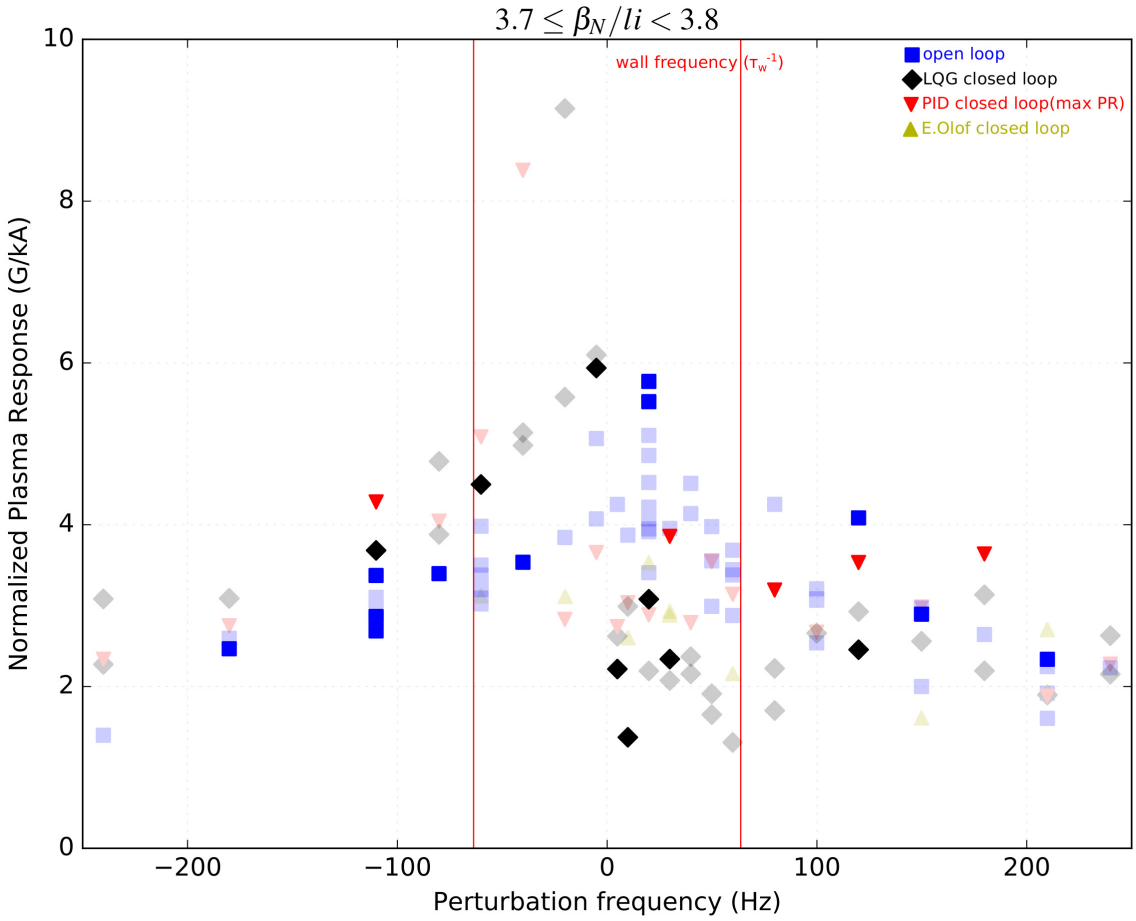

Figure 8.2: Normalized plasma response (G/kA) versus I-coil perturbation frequency (Hz) for open loop (blue), LQG closed loop (black) and proportional closed loop (red) for the time averaged $3.7 \leq \beta_N/li < 3.8$. The filtering on time-averaged β_N/li is done in order to isolate and compare similar discharges, as normalized plasma response is proportional to β_N/li. LQG feedback was most successful at suppressing plasma response at frequencies below the wall frequency as can be seen by the trend of black diamonds.

7.1, of normalized plasma response versus perturbation frequency is shown in figure 8.3, along with open loop (no feedback) and closed loop (with feedback) for discharges with time-averaged $\beta_N/li \geq 3.7$. The decent agreement between simulation and experiment

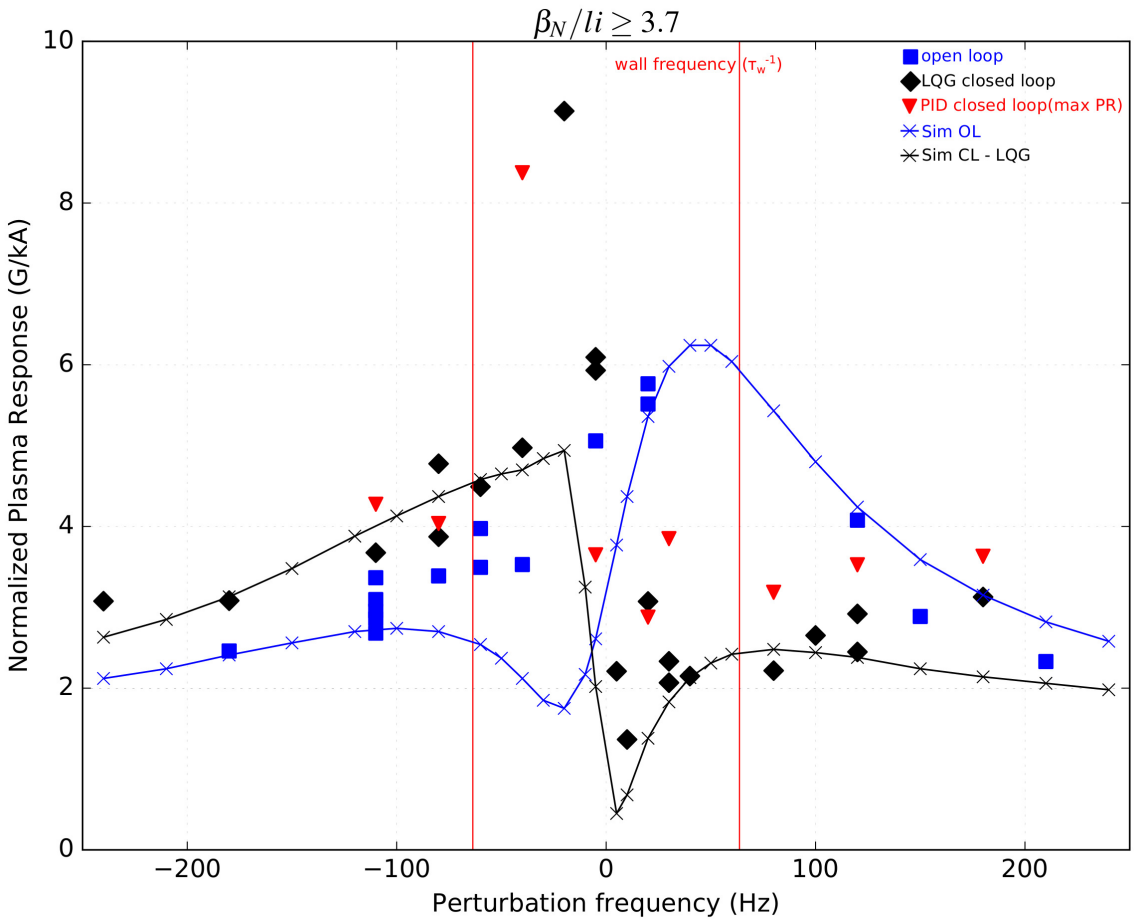

Figure 8.3: Normalized plasma response (G/kA) versus I-coil perturbation frequency (Hz) for open loop (blue), LQG closed loop (black) and proportional closed loop (red) for time averaged $\beta_N/li \geq 3.7$. VALEN simulation results for closed loop (black line) and LQG closed loop (blue line) are also shown. Simulation results well approximate the experimental results for discharges with $\beta_N/li \geq 3.7$.

for higher values of β_N/li shown in figure 8.3 shows the simulation techniques of chapter 7 were successful at designing an LQG controller that worked well with minimal testing in actual experiments. Figure 8.4 shows C-coil RMS current versus I-coil perturbation

frequency for both LQG and proportional controllers. The filtering effects of the wall were modeled in VALEN; the results show that C-coil RMS current using the LQG controller based on VALEN (black diamonds) uses considerably less current at higher frequencies than the proportional controller (red triangles). Moreover, the LQG controller applies more current at frequencies below the wall frequency because that frequency region is where feedback will be most effective. Together figures 8.2 and 8.4 show that the proportional controller (red triangles) applied far more current at higher frequencies without any observable improvement in reduction of normalized plasma response compared to the LQG controller (black diamonds).

Another performance metric for feedback is a parameter called "efficiency", defined as $\mathrm{eff} = \frac{\sqrt{2}}{2} \times \frac{I_*}{I_{\mathrm{RMS}}} \times 100\%$, where I_* is the current applied at the perturbation frequency (as shown in the FFT plot), and I_{RMS} is the RMS current of the n=1 fit of the C-coil current during the perturbation period. The $\frac{\sqrt{2}}{2}$ is a normalizing scale factor to ensure that a purely sinusoidal wave would not have an efficiency of greater than 100%. In qualitative terms, the efficiency demonstrates how well the controller identifies the perturbation in comparison to other noise and disturbances that may be occurring at frequencies other than the perturbation frequency. C-coil efficiency versus perturbation frequency for closed loop discharges with $\beta_N/li > 3.0$ is shown in figure 8.5. This plot shows the variation of the proportional controller's (red triangles) frequency dependence on efficiency: that is, as frequency increases, a large fraction of the proportional controller's power is not being used to suppress the plasma response but rather being wasted to drive plasma response at the system's resonant frequency. In contrast, the efficiency of the LQG controller (black diamonds) is high across a wide range of perturbation frequencies.

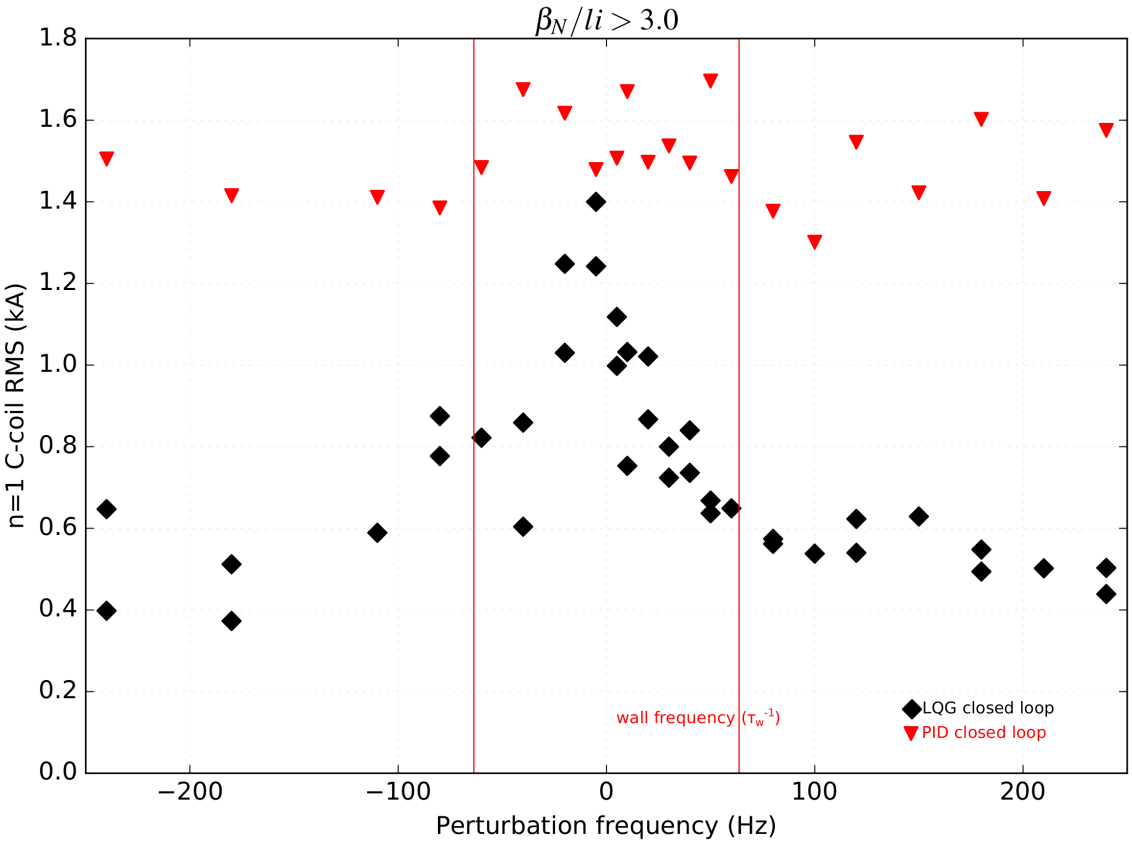

Figure 8.4: C-coil RMS current (kA) versus I-coil perturbation frequency (Hz) for LQG closed loop (black) and proportional closed loop (red) for time averaged $\beta_N/li > 3.0$. The filtering effects of the wall were modeled in VALEN; the results show that C-coil RMS current using the LQG controller based on VALEN uses considerably less current at higher frequencies than the proportional controller.

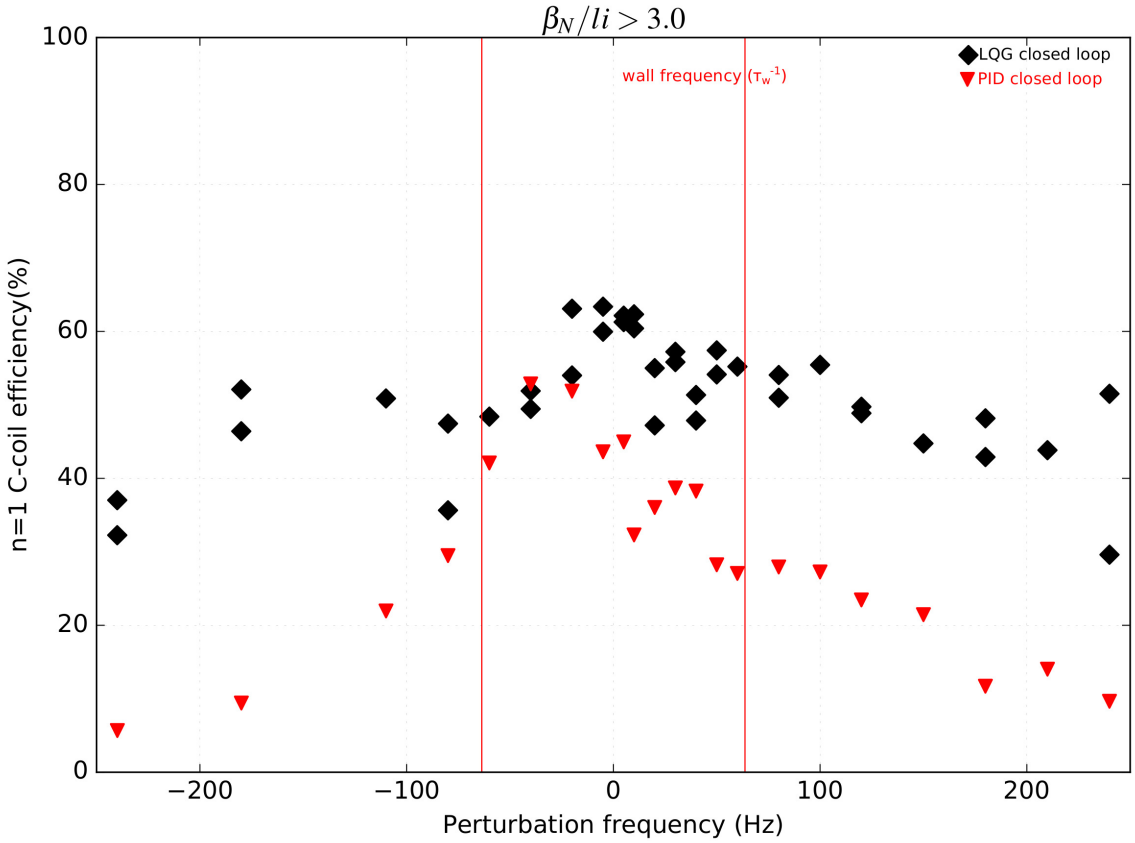

Figure 8.5: C-coil efficiency (%) versus I-coil perturbation frequency (Hz) for LQG closed loop (black) and proportional closed loop (red) for time-averaged $\beta_N/li > 3.0$. At higher frequencies a large fraction of the proportional controller's power is not being used to suppress the plasma response but rather being wasted to drive plasma response at the system's resonant frequency. In contrast, the efficiency of the LQG controller is high across a wide range of perturbation frequencies.

8.2 Perturbations at a Single Frequency

Figure 8.6 shows the normalized plasma response versus β_N/li for all open and closed loop perturbations done at 20Hz from 2016 and 2017 experiments. A clear trend is visible in which normalized plasma response rises with β_N/li, but the relation is not necessarily linear. An improvement between the 2016 controllers (mark 1) and the 2017 controller (mark 2) can be also be seen in figure 8.6. A more optimized controller was created by trial and error by changing the penatly matrices of the linear quadratic regulator (equation 6.1) using the techniques of section 6.4. Each newly designed controller was tested in closed loop simulation as described in section 7.3.2 until a controller was found that was most effective at suppressing the normalized plasma response with achievable control inputs in each control coil. Dozens of simulations were done in this manner until the mark 2 controller was arrived upon.

8.3 Attempts at High Performance Plasmas

8.3.1 Feedback on High Performance Plasma

The LQG controller developed in section 7.1 was briefly tested during high performance plasma experiments on DIII-D in March 2016. These experiments were conducted in β_N regimes in which RWMs are likely to occur. Figure 8.7 shows the results of two similar discharges with the major exception that feedback is on or off. Discharge 165885, in which there was no C-coil feedback, terminates early due to an RWM around 2100ms, as shown by the large excursion observed in poloidal magnetic field activity and highlighted in yellow of subplot (a). This large mode is responsible for the collapse in β_N seen in subplot

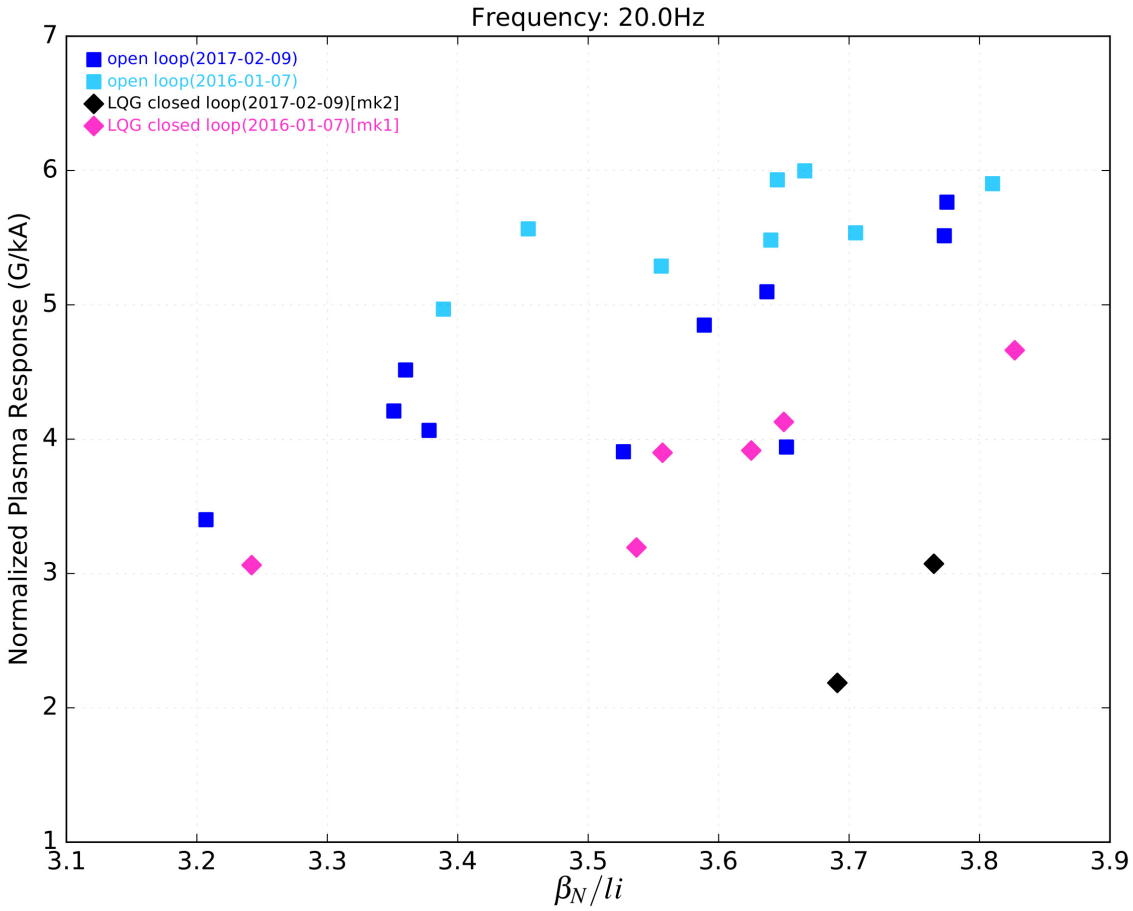

Figure 8.6: Comparison of β_N/li versus normalized plasma response (G/kA) at 20Hz perturbation frequency for all open and closed loop experiments (2016 and 2017). The cyan squares and magenta diamonds depict the 2016 experiment described in section 7.1, while the blue squares and black diamonds are data from the 2017 experiment described in section 8.1.

(b) of figure 8.7. Discharge 165898 (the trace in blue), on the other hand used the VALEN-based LQG algorithm for C-coil feedback, identical to one used during the perturbation experiments, and was thus able to achieve the desired β_N until the end of the discharge. Subplot (b) in figure 8.7 shows traces of β_N for feedback-off (black trace) and feedback-on (blue trace) discharges. The time period highlighted in red in (b) is expanded in subplot (a) of figure 8.7 and displays the $n = 1$ fit of the midplane poloidal field (B_p) sensors. The feedback-off trace (black trace) shows significant $n = 1$ MHD activity around 2080ms and is an RWM; the early stages of the RWM are highlighted in yellow. Subplot (c) expands the highlighted yellow region in subplot (a); the linear fit of the log of the feedback-off magnetics signal, .88ms, is consistent with the expected growth time of the RWMa and is shown in sublplot (d). Subplot (e) shows the $n = 1$ fit of the C-coil feedback signal used for the feedback-on case, in which an RWM did not occur.

8.3.2 Dedicated High Performance Plasma Discharges

Several discharges were devoted to testing the LQG and proportional controllers on high β_N plasmas, a parameter that should be maximized for Advanced Tokamak operation. For these discharges, the β_N target for the neutral beam injection system is ramped from 3.1 to 5.1, from 3 to 5 seconds, or from 3 to 4 seconds. Graphical summaries for the maximum β_N reached for LQG and proportional control are shown in figures 8.8 and 8.9, respectively. While both controllers nominally reach the same maximum β_N at approximately the same time, the proportional controller uses 77% more current, as shown in figure 8.9 (note the different scales for C-coil current in the two figures), much of which is at the resonant frequency -31.7Hz (see figure 8.10). This significant amount of field generated by the C-coils drives a strong plasma response at that frequency, as

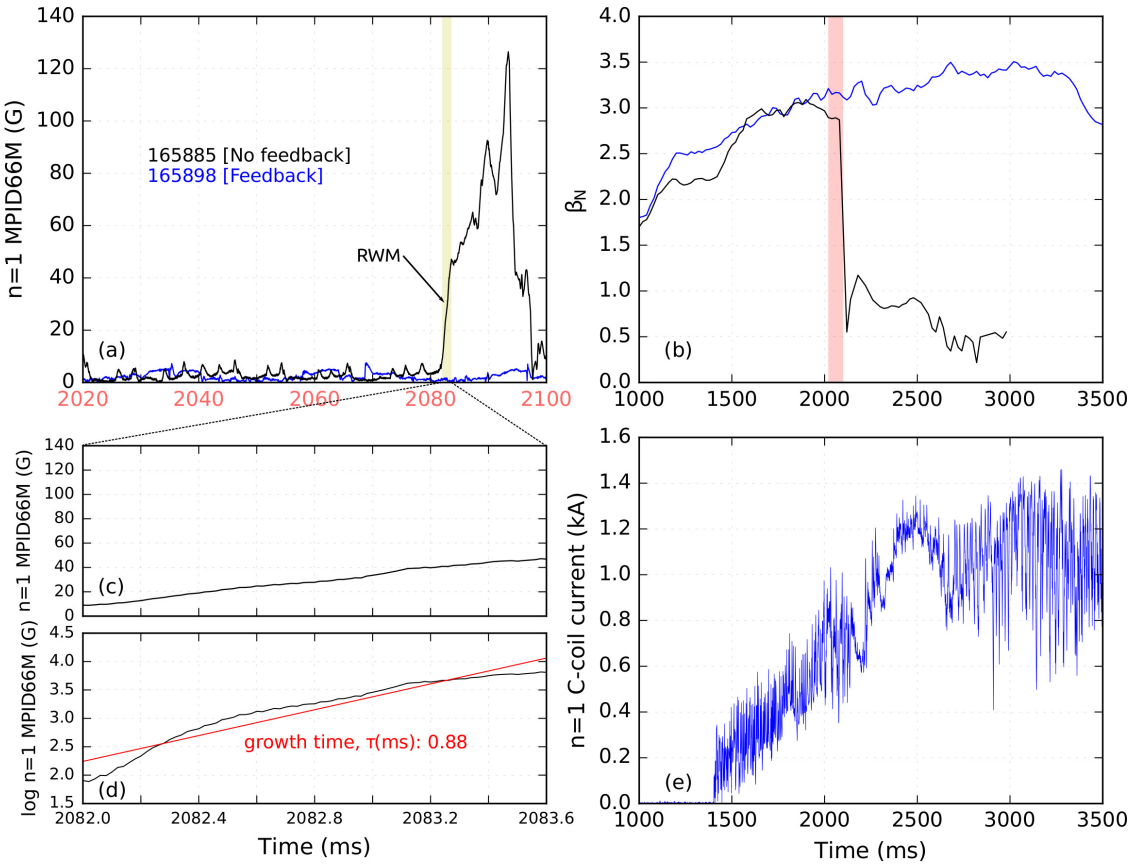

Figure 8.7: (a)(c)(d) the n=1 fit of midplane poloidal field, B_p, magnetics (MPID66M), (b) normalized plasma beta, β_N, and (e) n=1 C-coil current are shown for high performance plasma discharges 165885 (black) and 165898 (blue). 165885 terminates early due to an RWM (region highlighted in red) and suffers a collapse in β_N. 165898 is able to achieve the desired β_N by C-coil feedback. (a) depicts the magnetics during the time highlighted in red in (b), during the RWM. (c) and (d) highlight the magnetics during the 1.6ms period highlighted in yellow in (a).

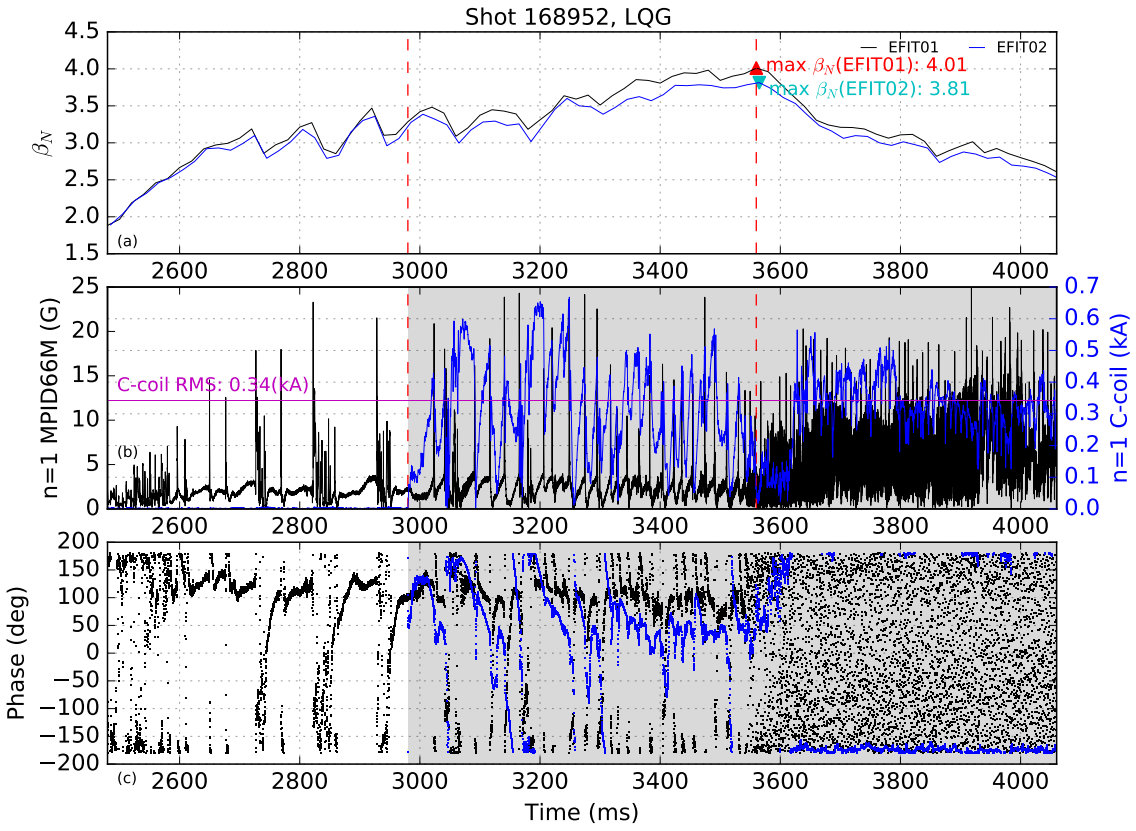

Figure 8.8: (a) β_N as calculated by the magnetic equilibrium reconstruction codes EFIT01 and EFIT02; (b) n=1 poloidal field (black)/C-coil current (blue); (c) n=1 poloidal field (black)/C-coil (blue) phase versus time for 168952 using the mark 2 LQG controller. They highlighted grey region indicates when feedback is on. The β_N target for the neutral beam injection system is ramped from 3.1 to 5.1, from 3 to 5 seconds, or from 3 to 4 seconds.

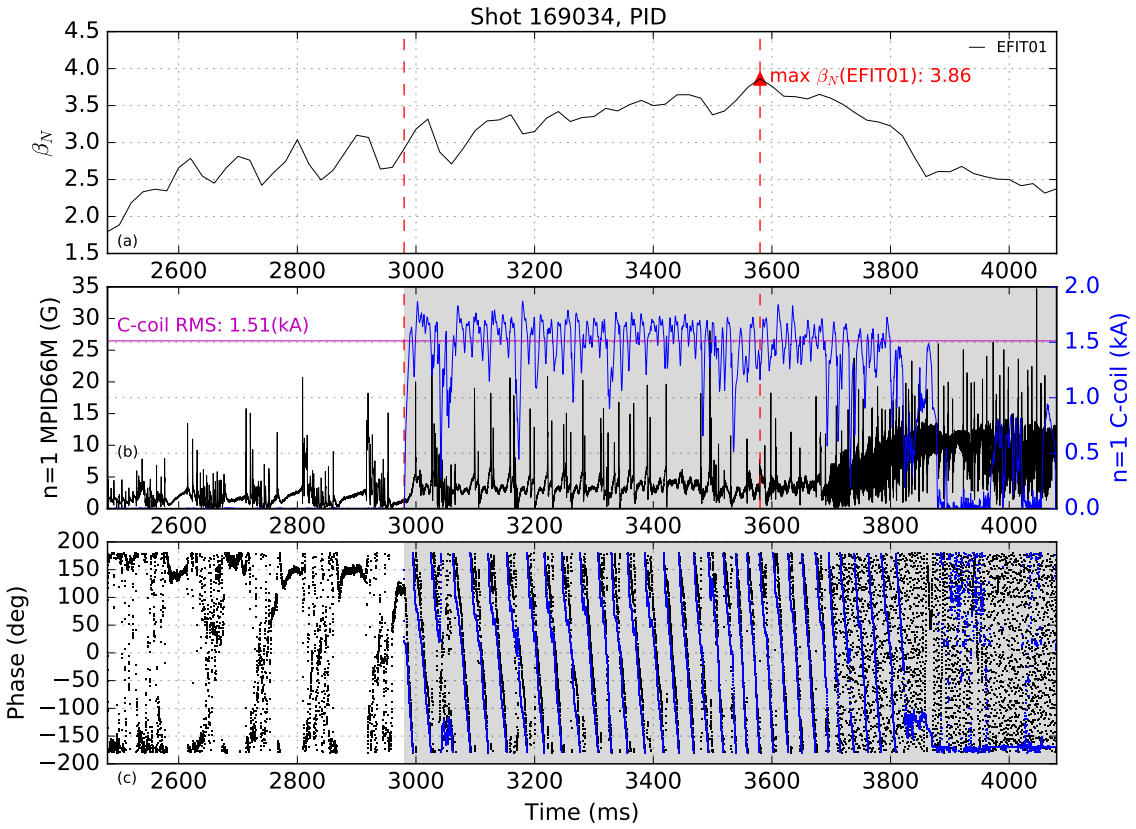

Figure 8.9: (a) β_N as calculated by the magnetic equilibrium reconstruction code EFIT01; (b) n=1 poloidal field (black)/C-coil current (blue); (c) n=1 poloidal field (black)/C-coil (blue) phase versus time for 169034 using the proportional controller. They highlighted grey region indicates when feedback is on. The β_N target for the neutral beam injection system is ramped from 3.1 to 5.1, from 3 to 5 seconds, or from 3 to 4 seconds. The phase plot shows plasma response being driven at the resonant frequency -31.7Hz by the C-coils.

shown in the phase plot of figure 8.9. Most of these discharges were ultimately hindered

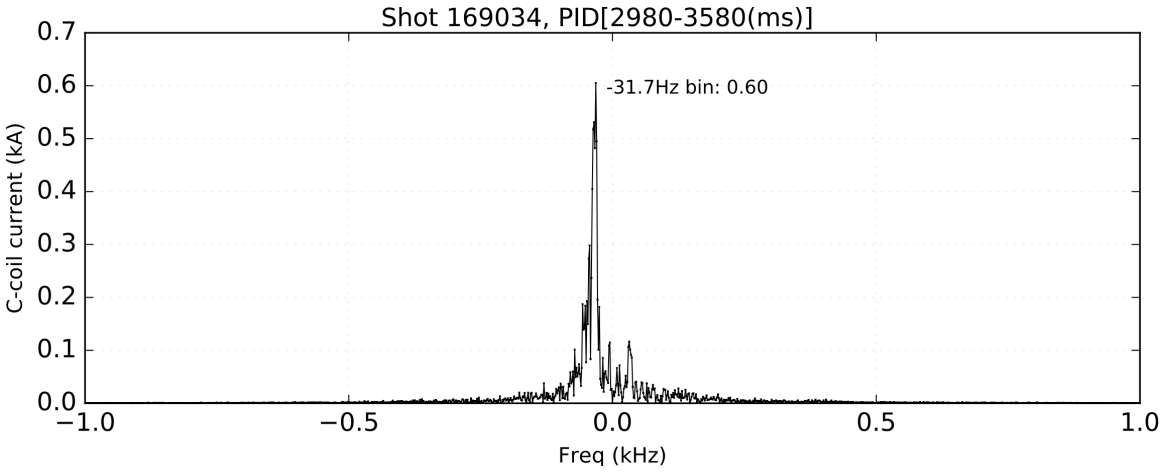

Figure 8.10: FFT of the n=1 C-coil current for 169034 starting from 2980ms to maximum β_N using proportional control. A large amount of current, .6kA, is being wastefully applied at the resonant frequency of -31.7Hz.

from reaching higher β_N due to the onset of tearing modes, which are seen in figures 8.8 and 8.9 and which are not affected by feedback control, shortly after the β_N peak. Tearing modes are indicated by the high frequency (> 5kHz) activity in the n=1 fit of poloidal field magnetics. Two independently generated kinetic EFITs[32] were made for discharge 168952 at time 3565, the peak β_N reached, and were analyzed using the stability code CalTrans (Corsica)[61] in order to determine a refined β_N from a kinetically constrained equilibrium as well as identify the no-wall and ideal-wall β_N limits. The calculated results for β_N were 3.57 and 3.67, the no-wall β_N limits were 3.52 and 3.67 and the ideal-wall β_N limits were 4.36 and 4.7.

8.3.3 Comparison of Steady-State Advanced Tokamak Discharges

The discharge 133103 has been identified as a DIII-D discharge exhibiting desirable properties for a steady-state Advanced Tokamak type discharge due to its noninductive current fraction near unity, high bootstrap current fraction and β_N>3.5 for over 1

second[62]. Discharge 133103 was used as the target discharge (with modifications, most notably the β_N target for Neutral Beam Injection (a plasma heating source) for all of the experiments described in sections 7.1 and 8.1. Discharge 133103 used the PCS' PID algorithm with the I-coils and SPA power supplies for RWM feedback. A similar high β_N discharge was attempted in shot 168953, similar to 168952 using the LQG controller on the C-coils. This controller was able to achieve a similar β_N for a longer period than shot 168952 due to less detrimental tearing activity. Discharge 168953 was able to nearly replicate the β_N trace of 133103 as shown in figure 8.11, demonstrating the effectiveness of the LQG controller using a more reactor relevant control coil set.

This chapter contains material which has been accepted for publication in *Control Engineering Practice* and material which has been submitted for publication in *Nuclear Fusion*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Atomic Energy Agency, 2017. The dissertation author was the primary investigator and author of these papers.

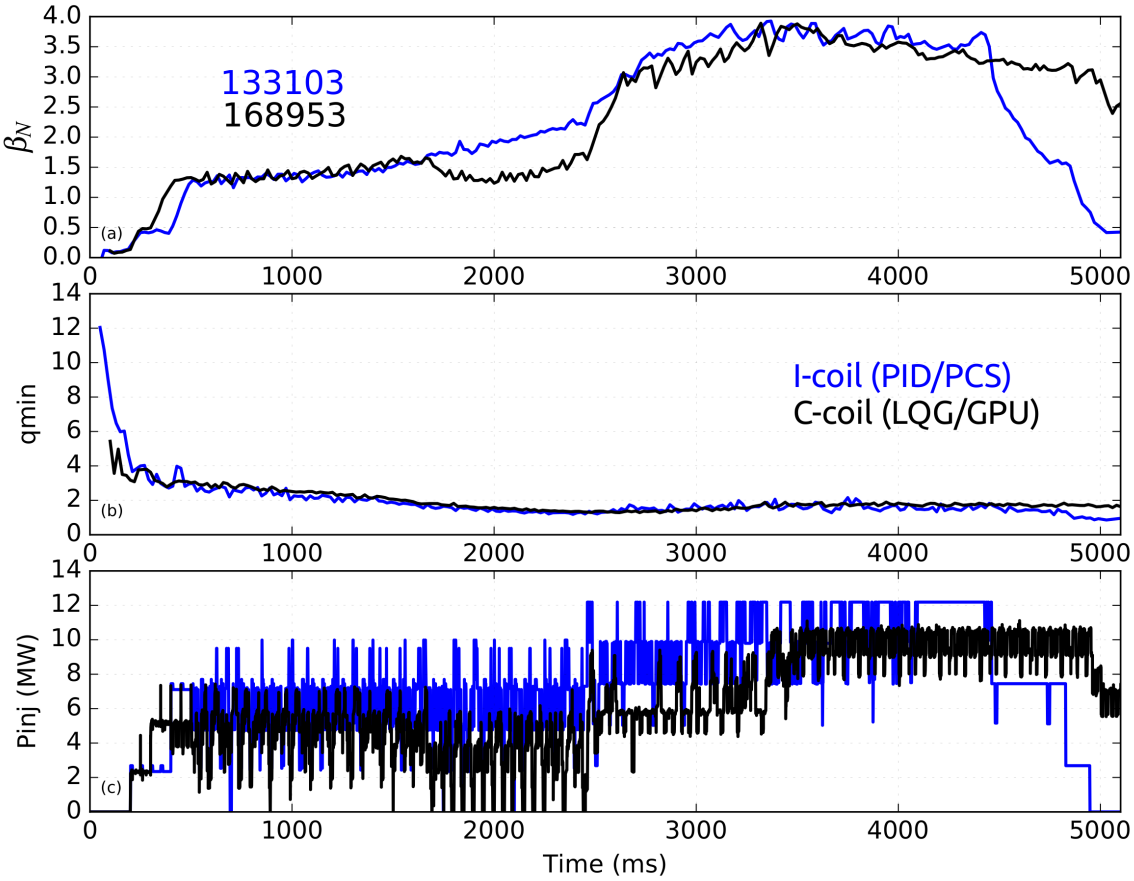

Figure 8.11: (a) β_N(EFIT01), (b) q_{min}(EFIT01) and (c) injected beam power for discharges 133130 (blue) and 168953 (black). 133103 is a DIII-D discharge with β_N at values for what is expected in a steady-state Advanced Tokamak. Discharge 168953 was able to effectively reproduce this discharge using external vice internal control coils.

Chapter 9

Conclusions and Future Work

Experimental results and simulations presented in this dissertation indicate that $\mathcal{H}_2$ optimal control techniques, also known as Linear Quadratic Gaussian (LQG) control, used with the VALEN RWM model are effective at mitigating resistive wall modes. These techniques were proven to be effective when used with control coils external to the tokamak's vacuum vessel wall that is relevant for next-step nuclear-grade fusion experiments. Perturbation experiments using DIII-D's I-coils (in-vessel coils) were used to validate controller designs and simulations of these experiments using VALEN were used to design controllers with optimized performance characteristics. These controllers were then used to demonstrate RWM control using external C-coils for feedback control.

9.1 Conclusions

An LQG controller based on the VALEN model has been developed for RWM feedback on DIII-D using its ex-vessel control coils (C-coil). This control algorithm has been implemented on a low-latency GPU-based control system using 24 magnetics sensors,

measuring poloidal and radial field at various toroidal and poloidal locations surrounding the plasma volume. The algorithm was first tested in experiments in which DIII-D's external coils provide feedback on plasma perturbations driven by the tokamak's internal control coils (I-coil). In this experiment, the I-coils were used to drive an n=1 plasma response on a stable plasma that mimics the 3D structure of an RWM. Parameters for the VALEN model were fit to perturbation experiments carried out at a single frequency, and using these empirically derived parameters, these experiments were simulated using VALEN. By simulation, feedback algorithms could be tested and optimized and led to a an optimized controller used in later experiments.

The controller was then tested during discharges that achieve high plasma pressure and conditions conducive to MHD modes like the RWM. Often these discharges failed to complete without any form of RWM feedback, but were able to attain higher β_N with either I-coil feedback using proportional only feedback gain driving the SPAs, or C-coil feedback using the LQG algorithm driving the AAs. Steady state, high β_N discharges that had only previously used I-coil feedback were very nearly reproduced using VALEN-based LQG feedback and the C-coils; an approach that may be extendable to next-step nuclear-grade tokamak devices.

9.2 Future Work

LQG control assumes the spectrum of noise in the system is white (colored for sampled data systems) and Gaussian for measurement noise and process noise. This is not very realistic. The Kalman filter described in this work was optimized very little during the course of experiments and simulation. Measurements in tokamaks are indeed corrupted by some amount of measurement noise (thermal noise) but often there are other MHD

events occurring that are not the RWM. One such example are edge localized modes (ELMs) that appear as period bursts in magnetics signals during high confinement or H-mode discharges. One method to deal with these would be to incorporate a disturbance model during observer design. This ,however, may require a linear model for ELMs, which may be impractical. Another optimal control technique that is worth investigating is $\mathcal{H}_\infty$ optimal control[42]. In LQG or $\mathcal{H}_2$ optimal control, the control law and observer attempt to minimize the system's $\mathcal{H}_2$ transfer function norm. In $\mathcal{H}_\infty$ control, it is the $\mathcal{H}_\infty$- norm that is minimized. This technique is often referred to as a *worst-case design* and is intended to improve system robustness. Other filtering techniques are worth investigating such as particle filters[42], which are non-linear filters and very computationally intensive. Particle filters rely on sequential Monte Carlo methods based on "particle" representations of probability densities, they are probability-based estimators and are well equipped to handle non-Gaussian noise. These techniques may offer improvements in removing system disturbances such as ELMs from any system state estimate and may help to further improve feedback.

The LQG feedback kernel described in chapter 6 and appendix B uses only one of the 13 available GPU Streaming Multiprocessors (SMs). Sequences of operations running on the GPU may be further parallelized into streams, which may run concurrently on the GPU[55]. A stream may run on one or more SMs, depending on the number of threads required for the stream. To fully utilize the computational resources of the GPU for real time computations, additional tasks may be run as concurrent streams. One such algorithm is Error Field Correction (EFC)[44], which seeks to cancel out magnetic fields generated by construction imperfections in the tokamak. The EFC algorithm produces essentially a DC field and is computed from toroidal field and plasma current measurements. An EFC

kernel has been run in parallel with the LQG feedback kernel, with the vector sum of the outputs of both algorithms being sent to power supplies. This summation is accomplished by a third kernel that waits for the completion of the other two algorithms to complete by waiting a prescribed number of clock cycles and then reading from their output buffers and adding the commands together. This is shown graphically in figure 9.1. The EFC algorithm computes currents on each lead coil and is used with current control power supplies. Because this field is essentially DC, a voltage control algorithm could easily be derived using Ohm's law, but remains to actually be done. Addition of the EFC algorithm brings the total number of concurrent streams running, and number of SMs used, up to three. One excellent use of the unused SMs and computing capability on the GPU would be for MHD equilibrium reconstruction. This is currently accomplished at DIII-D using the EFIT code[32], which iteratively solves the Grad-Shafranov equation. Parallelization of the EFIT code for use on the GPU, named P-EFIT, has already been demonstrated at the EAST tokamak[63]. Real time equilibrium reconstruction is essential for discharge shape control, also known as the "isoflux" technique[64]. This could be easily handled by an additional stream as shown in figure 9.1. This is the future of GPU powered control.

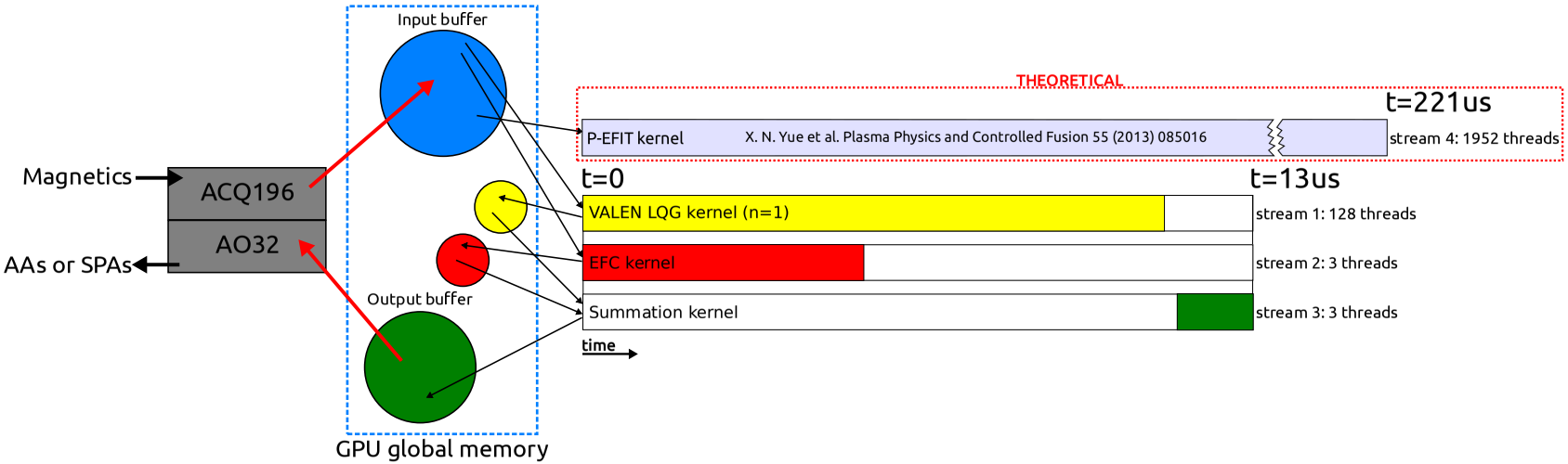

Figure 9.1: A simple graphical representation of the GPU streams that currently run during feedback. The P-EFIT kernel could theoretically run in parallel on other available GPU Streaming Multiprocessors (SMs).

Appendix A

Mode Fitting

This section describes how the amplitude and phase of MHD modes, with toroidal mode number n, are extracted from an array of sensor measurements that are spaced toroidally at different poloidal locations around a tokamak[65]. To solve for the sine and cosine components for a given sensor, the following over determined set of equations is used:

$$\begin{aligned} S*\sin(n*\phi_1^{sensor})+C*\cos(n*\phi_1^{sensor}) &= \Phi_1^{sensor} \\ S*\sin(n*\phi_2^{sensor})+C*\cos(n*\phi_2^{sensor}) &= \Phi_2^{sensor} \\ &\vdots \\ S*\sin(n*\phi_N^{sensor})+C*\cos(n*\phi_N^{sensor}) &= \Phi_N^{sensor}, \end{aligned} \tag{A.1}$$

where ϕ is the toroidal angle of the sensor, and N is the number of sensors in the array. Equation A.1 may be more compactly written as a matrix vector equation:

$$\mathbf{A}_{N\times 2}\left[\begin{smallmatrix} S \\ C \end{smallmatrix}\right]_{2\times 1} = \vec{\Phi}^{sensor}_{N\times 1}. \tag{A.2}$$

The matrix $\mathbf{A}$ in equation A.2 is often called a *fitting* matrix. To solve equation A.2 for S and C, the pseudo-inverse of matrix $\mathbf{A}$ is used, which will give a least squares solution:

$$\begin{bmatrix} S \\ C \end{bmatrix}_{2\times 1} = [\mathbf{A}^{+}]_{2\times N} \vec{\Phi}^{sensor}_{N\times 1}. \tag{A.3}$$

The amplitude and phase of the mode can be determined from the sine (S) and cosine (C) components of the mode as,

$$\text{amplitude} = \sqrt{C^2 + S^2} \tag{A.4a}$$

$$\text{phase} = \arctan(C/S). \tag{A.4b}$$

Typically, $n = 1$ is the mode of interest, as it normally contains most of the mode's energy. Equation A.3 can be used for fitting the current in control coils as well, where ϕ would represent the location of the center of the coil and the vector of sensor fluxes, $\vec{\Phi}^{sensor}$, would be replaced by a vector of currents from each control coil, $\vec{I}^{coil}$. Equation A.3 may also be used for a time series of data, which is to say that the matrix $\mathbf{\Phi}^{sensor}_{N\times M}$ represents the M measurements in time from N sensors, and sine and cosine components become $1 \times M$ arrays. To remove any DC component from a signal or arrray, the mean of the array is subtracted from each element, $X_0 = X - \bar{X}$. For Fast Fourier Transforms (FFT), the $n = 1$ component may be transformed by creating a complex number composed of the sine and cosine components, $z = C_0 + iS_0$, and then taking the FFT of z.

For fitting complex phasors of magnetics or currents, as described in section 7.2.1, the fitting matrix must become complex as well. This new complex fitting matrix will fit the sine and cosine components into one complex number, meaning the matrix of equation A.3 will now have dimension $1 \times N$. The new fitting matrix is constructed by

taking each column of sine and cosine components and making a single complex column entry with the form: $.5 * (C + iS)$. The factor of $.5$ accounts for the signal contribution at negative frequencies that arises when taking the Fourier transform of a real signal.

Appendix B

Control Algorithm Implementation

This chapter describes the operation of the valiant LQG control algorithm based on the DIII-D VALEN RWM model. Also described here are the scripts and functions used on Tesibius to implement real-time RWM control.

B.1 Building VALEN models

B.1.1 makeVALEN.py: A Python Script to Make VALEN Models

`makeVALEN.py` is located on the General Atomics Venus computing cluster in `/u/clementm/valiant/VALEN` and has option parsing capability; information on these options may be accessed by entering:

```
# makeVALEN.py -h .
```

Important parameters that are passed to `makeVALEN.py` are VALEN parameters s and α, the i1 and i2 files which specify the conducting finite elements as well as the magnetic field normal component to the unperturbed plasma surface and the sample rate to which the continuous time model will be discretized to. This python script makes calls to the

three VALEN executables, `CUPR13`, `CUDR13NN` and `CUREVF02NN`. These executables ultimately generate the $\mathbf{L}$, $\mathbf{R}$ and $\mathbf{M}$ matrices described in chapter 3. From here the script puts VALEN into state-space form, converts the flux measurement to field measurements by computing the sensor differencing and using the sensor area and number of coil turns. This continuous time state-space model is saved as a python numpy .zip file. From here another script, `reduceVALEN.py`, does both the discretization based on the sample rate specified earlier and the balanced realization and reduction using the SLICOT routine `AB09BD`[52]. The ultimate product of this script is a numpy .zip file containing the discrete full and reduced models, the sample rate and a description of the model capturing its important parameters.

B.1.2 The VALEN Python Class

The python class, `VALENcls`, defined in `/u/clementm/valiant/VALEN/VALENclass.py` takes the numpy .zip file generated by `makeVALEN.py` and creates a python object whose attributes are the important state-space VALEN matrices, sample rate and description of the VALEN model. This class is used in the scripts subsequently described in this chapter when they are passed a VALEN model stored in a numpy .zip file. For example, the VALEN .zip file when loaded in to the VALEN class would have stored in its `.desc` attribute: "VALEN built from i1 file: i1ga.HANSONwcc.2015.#1, i2 file: i2v.o133103_3330.008. s parameter: 5.667e-02. alpha parameter: 0.000e+00. Discretized version of ssVALEN_201701261547.npz. Balanced and truncated using SLICOT AB09BD."

B.2 Operation

This section describes python scripts and cuda functions that are used for real-time feedback control on Tesibius (section 5.3). On Tesibius, control functionality is provided in two main directories, `/pcshome/clementm/valiant/` and `/opt/control/`. `/pcshome/clementm/valiant/` contains code relevant to real-time feedback while `/opt/control/` contains code needed to initialize and control D-TACQ hardware. To setup digitizers the script `init_rtm-t.sh` is sourced as the root user in order to get the RTM-T ready for control and sets the environment variable `RTM_T_READY` to `YES`. For example, to set the RTM-T to receive a hard trigger from an external clock (i.e. the DIII-D timing system), enter:

```
# .  init_rtm-t.sh lemo .
```

A more inclusive script for getting Tesibius ready for operation is the script `rockNroll.sh`. This script terminates any active threads, resets the GPU, initializes RTM-Ts for a hard trigger and boosts GPU clocks to their maximum frequency (758MHz),

```
# .  rockNroll.sh .
```

B.2.1 Host Script: valiant.py

The python script `valiant.py` will execute RWM feedback for DIII-D discharges requiring it. Three parameters are passed to the control script from the command line: shot number, feedback commence time (in ms) and feedback termination time (ms), if feedback is desired for one sustained period. For perturbation experiments as described in chapter 8, where six periods of feedback are required, 13 parameters are passed to the control script: shot number, period 1 feedback commence time (ms), period 1 feedback

termination time (ms), period 2 feedback commence time, etc. Based on the number of inputs it receives, `valiant.py` is able to determine if one period or six periods of feedback are required. If the script is unable to determine how many periods are desired based on the arguments it receives, the user is prompted for the shot number and feedback times. As an expample, feedback for shot 900000 from 2 seconds to 5.5 seconds would be commanded like so:

```
# valiant.py 900000 2000 5500 .
```

`valiant.py` also has option parsing capability; information on these options may be accessed by entering:

```
# valiant.py -h .
```

If `valiant.py` is not called as the root user, `valiant.py` will inform the user and exit the python environment. `valiant.py` establishes the low latency control definitions required for the ACQ196 digitizer and AO32 output modules, including the sample rate to be used, and arms them for operation. The reduced VALEN model, pre-comupted controller gains and vacuum field compensation matrices are imported and GPU memory allocation is handled via the `PyCUDA` software package[66]. `valiant.py` requires the imported VALEN discretized model to match the sample rate. Based on the sample rate specified in the script, `valiant.py` calculates three chronometric milestones, the TLATCH when feedback will commence, or digitzer timestamp, the TLATCH when feedback will terminate and 20ms prior to when feedback will commence. No DIII-D shot is expected to last longer than 17 seconds, beginning from trigger 6A which occurs at -10 seconds and 7 seconds of discharge. Based on this and the sample rate, `valiant.py` calculates how many samples it should receive from the digitizer after being triggered. From the size of the reduced VALEN model, `valiant.py` calculates the size of the thread block it will

launch based on a predefined parameter called `WORK_SIZE`. Once all initializations are completed by `valiant.py`, it launches the valiant control kernel, defined in `valiant.cu`, on the GPU and turns over all control to it. The next section describes the GPU kernel that is responsible for control. While the kernel waits for trigger 6A, `valiant.py` continuously probes the PCS state from the lockserver. If a shot is not imminent or gets aborted prematurely, the kernel is terminated. `valiant.py` then copies GPU buffers that contain the lead coil voltages calculated during the discharge, archives them to MDSPlus and archives information about the VALEN model used as well as the parameters used to design the controller gain. `valiant.py` then frees GPU memory and performs other cleanup operations on the host side.

B.2.2 The valiant() Control Kernel

In practice, `valiant.cu` is compiled to a .cubin shared object file to be used by `valiant.py`. `valiant.py` launches on the GPU the real-time control kernel `valiant()` or `hexa_valiant()`. Important arguments to `valiant()/hexa_valiant()` are pointers to GPU global memory `hb` and `ao_hb`. These locations are the memory buffers to which the ACQ196 sends it data and the AO32 reads output commands, respectively. Other important matrices passed to `valiant()/hexa_valiant` are: 1) the closed loop dynamics matrix Φ, 2) the observer gain $\mathbf{K}_o$, 3) the controller gain $\mathbf{K}_c$ and 4) the vacuum field compensation matrix $\mathbf{C}_{comp}$. First, each thread must identify itself and some important parameters to determine its location in a matrix. `valiant.py` launches a 1-dimensional block of 128 threads. This means each thread will have a `threadIdx.x` from 0 to 127. From this thread ID, each thread then computes which block column it is in, `blockCol = threadIdx.x / 32`. For a reduced VALEN model's system matrix, there are four block columns in this

matrix and `blockCol` may have values from 0 to 3. Next, each thread determines its row in the matrix, `row = threadIdx.x % 32`, where `%` is the modulo operator. Next, shared memory locations are allocated for all matrices and variables to be used for the algorithm. The time invariant matrices passed to `valiant()/hexa_valiant()` are then copied into shared memory. Shared memory is used because it is much faster than using global memory and because `valiant()/hexa_valiant()` can be handled by a single Streaming Multiprocessor (SM), meaning all threads have access to the same shared memory[55]. At this point, the kernel enters its 13μs control loop. Every time new data is streamed from the ACQ196, it is time stamped with a TLATCH, starting from 0. Each thread probes the memory location in the input buffer for a new TLATCH. Once a new TLATCH has been pushed onto the input buffer, a thread then goes to work on data in the input buffer. The global memory location `hb` is allocated as a region containing unsigned 32 bit integers, because TLATCH is an unsigned 32 bit integer. The actual 96 channels of data are actually digitized as signed 16 bit integers. Bearing this in mind, the first 12 threads of block column 0 read the first 12 memory locations from `hb`, then performs a bit shift of 16 bits where necessary, to get the 24 magnetics signals into a vector in shared memory. In a similar fashion, threads 13 to 16 of block column 0 read the next four memory locations to get the 8 digitized coil currents needed for vacuum field compensation. Next, each thread must determine its next action based on a comparison of the current TLATCH to the predetermined time milestones. If the current TLATCH is less than 20ms prior to the feedback commence time, no computation is necessary. If the current TLATCH is greater than 20ms prior to the feedback commence time but less than the feedback commence time, each signal is summed in a buffer. This is done for signal baselining and at the end of the 20ms period, the summed signals are then divided by the number of samples in

the 20ms baseline period to give the average value of the signal. If the current TLATCH is greater than the feedback commence time and less than the feedback terminate time, each thread enters the feedback section of the control loop.

To handle the matrix-vector multiplication required by LQG control, a general description of how `valiant()` does the general computation $\vec{y} = \mathbf{A}\vec{x}$ follows[67]. With each thread knowing where it resides in the matrix $\mathbf{A}$ with regard to row and block column and where it resides in vector $\vec{x}$ with regard to vector row (vecRow = blockCol), each thread then multiplies corresponding matrix-vector locations by traversing `WORK_SIZE(8)` columns along its row in its respective block column of $\mathbf{A}$ and then traversing `WORK_SIZE(8)` rows down the vector $\vec{x}$. This operation then forms a submatrix that is 32 rows by 4 columns `(32 / WORK_SIZE)`. Threads in the first block column then go onto sum the rows of this submatrix to give the final result $\vec{y}$. A graphical description of this operation is shown in figure B.1.

Using the vector of coil currents that was created early in the control cycle, vacuum field is calculated,

$$\vec{y}_{\mathrm{vac}} = \mathbf{C}_{\mathrm{comp}}\vec{y}_{\mathrm{coil}}; \tag{B.1}$$

next, a new state estimate is computed using the old state estimate and the new measurements

$$\hat{x}_{\mathrm{k+1}} = \mathbf{\Phi}\hat{x}_{\mathrm{k}} + \mathbf{K}_{\mathrm{o}}(\vec{y}_{\mathrm{k}} - \vec{y}_{\mathrm{baseline}} - \vec{y}_{\mathrm{vac}}). \tag{B.2}$$

The state estimate is used for the controller,

$$\vec{u}_{\mathrm{k}} = -\mathbf{K}_{\mathrm{c}}\hat{x}_{\mathrm{k+1}}, \tag{B.3}$$

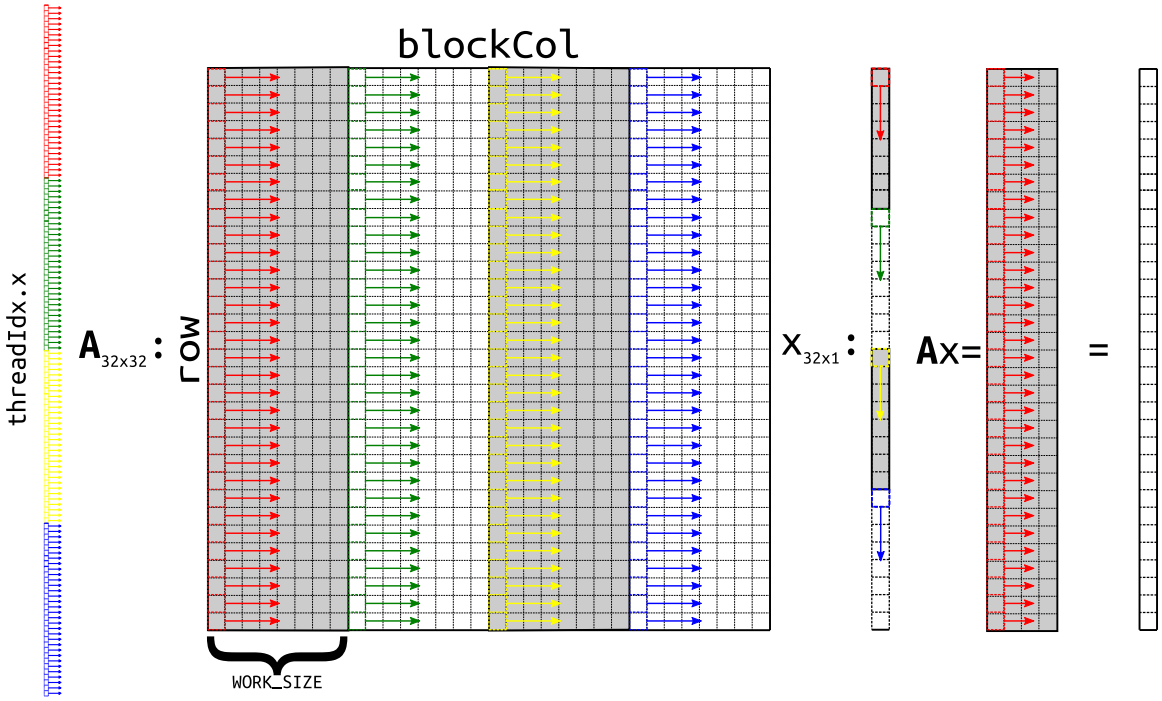

Figure B.1: A graphical description of how valiant() does the general matrix-vector product $\vec{y} = \mathbf{A}\vec{x}$.

and finally the state is advanced in time,

$$\hat{x}_k = \hat{x}_{k+1}. \tag{B.4}$$

Before the commands in $\vec{u}$ are written to the output buffer, a check of plasma current is done, to ensure that a disruption has not occurred and that a shot is still in progress. If the plasma current check is passed, the voltage is converted to digitizer counts and scaled to accommodate the Audio Amplifier gain and copied to `ao_hb`, the output buffer. If the current TLATCH is greater than the feedback termination time, output voltages are commanded to 0. Finally at the end of a control cycle, the output voltage and TLATCH are archived to buffers in GPU global memory for archiving by `valiant.py` once control is relinquished by the GPU. After completion of all control cycles, coil and magnetics

baselines are copied to global memory for archiving by `valiant.py`. At this point the kernel terminates and control is handed back over to `valiant.py` on the host.

B.2.3 Kernel Determinism

Reliably ensuring the completion of the feedback computation performed by `valiant()` within a control cycle is imperative for real-time control on the GPU. NVIDIA makes no guarantees about the deterministic nature of computation of their GPUs. This means that they make no guarantee about how fast a kernel may run from one time to the next. For this reason, many engineers are reluctant to use a GPU for real-time control. For this reason, a statistical study of the time it takes the `valiant()` kernel to run, once new data has been pushed by the digitizer into global memory, was performed. This was also used to determine the sample rate of the system. Timing of the kernel was performed by using the CUDA function `clock64()` [55] which allows processor clock cycles to be counted. By counting the number of cycles required to complete a cycle, from immediately after the kernel has determined new data is in the buffer to the archiving of TLATCH, and knowing the clock speed (758MHz), the wall-time required for computation may be determined. This measurement was carried out for over 1 million runs of the kernel and is shown in figure B.2. From this the jitter of the kernel may be determined. The mean time for computation is around 11.4μs with a standard deviation of less than .2μs. This is very much within an acceptable range for real-time control application.

B.2.4 Host Script: makeGains.py

The script `makeGains.py` is used to make LQG matrices for real-time feedback, as described in chapter 6, and has option parsing capability; information on these options

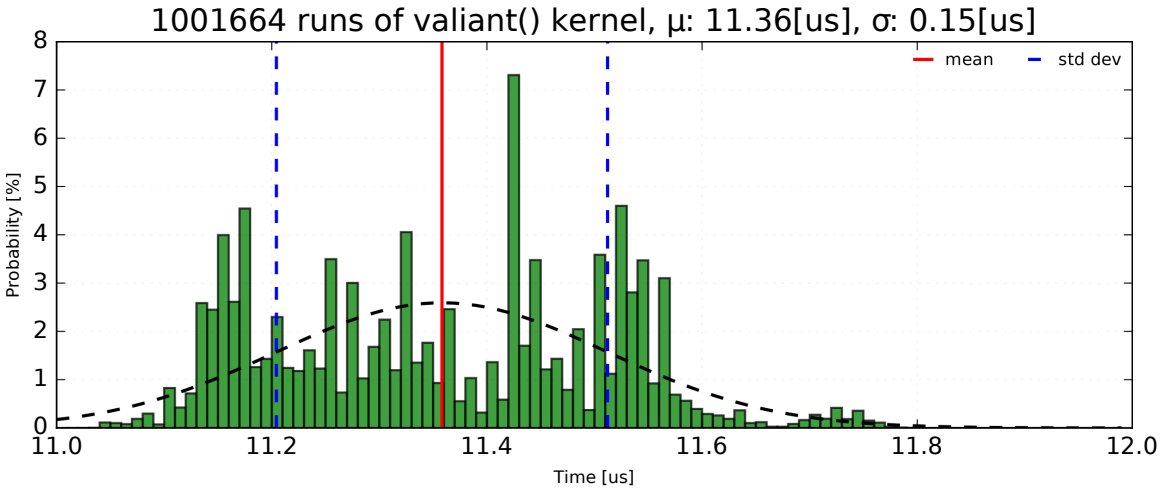

Figure B.2: Measurements of the wall-time for 1001664 runs of the `valiant()` control kernel. The measurement is made by counting the number of clock cycles required for the kernel to complete once it receives new data from the digitizer in GPU global memory. The mean and standard deviations are shown as the red and blue dashed lines respectively, which gives an idea of the jitter for the kernel. The black line is a normal distribution with the computed mean and standard deviation.

may be accessed by entering:

```
# makeGains.py -h .
```

Important options which may be specified from the command line are `qc`, `rho`, `n` and `tset`. These parameters only affect the design of the controller gain matrix $\mathbf{K}_c$. The observer gain is designed the same way for every VALEN model (see section 6.3). The scalar option `rho` will affect the value of the diagonal of the penalty matrix $\mathbf{Q}_c$ by setting the diagonal value of $\bar{\mathbf{Q}}_c$ defined in equation 6.3. The scalar option `n` defines the number of Hankel singular values to keep: that is to say the number of diagonal entries of $\mathbf{H}$, defined in 6.4, which are set to unity. The scalar option `rho` sets the parameter ρ defined in equation 6.6. The option `tset` defines the settling time.

B.2.5 Host Script: pyPCS.py, GUI to Control Scripts

The scripts `valiant.py`, `magsCal.py` and `pwrSupTest.py` may also be controlled by a graphical user interface (GUI), which is launched via the script `pyPCS.py`. The GUI allows the user to enter times, shot number, select options, specify a VALEN model and launch the script using check boxes, text entry boxes and buttons. The GUI also enables control of error field correction (EFC) from Tesibius, initialization of the RTM-Ts and resetting the GPU. A screenshot of the GUI is shown in figure B.3.

This chapter contains material which has been accepted for publication in *Control Engineering Practice*. Clement, Mitchell; Hanson, Jeremey; Bialek, Jim; Navratil, Gerald., International Federation of Automatic Control, 2017. The dissertation author was the primary investigator and author of this paper.

Figure B.3: A screenshot of the graphical user interface (GUI) launched by the host script pyPCS.py.

Bibliography

[1] F.F. Chen. *Introduction to Plasma Physics and Controlled Fusion*. Springer International Publishing, 2015.

[2] P.M. Bellan. *Fundamentals of Plasma Physics*. Cambridge University Press, 2008.

[3] J.P. Freidberg. *Ideal MHD*. Cambridge University Press, 2014.

[4] D. J. Griffiths. *Introduction to Electrodynamics*. Prentice Hall, 3rd edition, 1999.

[5] F Troyon, R Gruber, H Saurenmann, S Semenzato, and S Succi. Mhd-limits to plasma confinement. *Plasma Physics and Controlled Fusion*, 26(1A):209, 1984.

[6] J. Wesson and D.J. Campbell. *Tokamaks*. International Series of Monographs on Physics. OUP Oxford, 2011.

[7] I. B. Bernstein, E. A. Frieman, M. D. Kruskal, and R. M. Kulsrud. An energy principle for hydromagnetic stability problems. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 244, pages 17–40. The Royal Society, 1958.

[8] M. D. Kruskal, J. L. Johnson, M. B. Gottlieb, and L. M. Goldman. Hydromagnetic instability in a stellarator. *The Physics of Fluids*, 1(5):421–429, 1958.

[9] V.D. Shafranov. On Magnetohydrodynamical Equilibrium Configurations. *Soviet Journal of Experimental and Theoretical Physics*, 6:545, 1958.

[10] D. Pfirsch and H. Tasso. A theorem on MHD-instability of plasmas with resistive walls. *Nuclear Fusion*, 11(3):259, 1971.

[11] M S Chu and M Okabayashi. Stabilization of the external kink and the resistive wall mode. *Plasma Physics and Controlled Fusion*, 52(12):123001, 2010.

[12] Yueqiang Liu, M.S. Chu, I.T. Chapman, and T.C. Hender. Modelling resistive wall modes in ITER with self-consistent inclusion of drift kinetic resonances. *Nuclear Fusion*, 49(3):035004, 2009.

[13] Yueqiang Liu, M. S. Chu, A. M. Garofalo, R. J. La Haye, Y. Gribov, M. Gryaznevich, T. C. Hender, D. F. Howell, P. de Vries, M. Okabayashi, S. D. Pinches, H. Reimerdes, and EFDA-JET contributors. Modeling of resistive wall mode and its control in experiments and ITER. *Physics of Plasmas*, 13(5):056120, 2006.

[14] I. T. Chapman, Y. Q. Liu, O. Asunta, J. P. Graves, T. Johnson, and M. Jucker. Kinetic damping of resistive wall modes in ITER. *Physics of Plasmas*, 19(5):052502, 2012.

[15] J. W. Berkery, S. A. Sabbagh, H. Reimerdes, R. Betti, B. Hu, R. E. Bell, S. P. Gerhardt, J. Manickam, and M. Podestà. The role of kinetic effects, including plasma rotation and energetic particles, in resistive wall mode stability. *Physics of Plasmas*, 17(8):082504, 2010.

[16] O. Katsuro-Hopkins, J. Bialek, D.A. Maurer, and G.A. Navratil. Enhanced ITER resistive wall mode feedback performance using optimal control techniques. *Nuclear Fusion*, 47(9):1157, 2007.

[17] E. J. Strait. Magnetic control of magnetohydrodynamic instabilities in tokamaks. *Physics of Plasmas*, 22(2):021803, 2015.

[18] M. Ariola, G. De Tommasi, A. Pironti, and F. Villone. Control of resistive wall modes in tokamak plasmas. *Control Engineering Practice*, 24:15 – 24, 2014.

[19] P. Martin, L. Apolloni, M.E. Puiatti, J. Adamek, M. Agostini, A. Alfier, S.V. Annibaldi, V. Antoni, F. Auriemma, O. Barana, M. Baruzzo, P. Bettini, T. Bolzonella, D. Bonfiglio, F. Bonomo, M. Brombin, J. Brotankova, A. Buffa, P. Buratti, A. Canton, S. Cappello, L. Carraro, R. Cavazzana, M. Cavinato, B.E. Chapman, G. Chitarin, S. Dal Bello, A. De Lorenzi, G. De Masi, D.F. Escande, A. Fassina, A. Ferro, P. Franz, E. Gaio, E. Gazza, L. Giudicotti, F. Gnesotto, M. Gobbin, L. Grando, L. Guazzotto, S.C. Guo, V. Igochine, P. Innocente, Y.Q. Liu, R. Lorenzini, A. Luchetta, G. Manduchi, G. Marchiori, D. Marcuzzi, L. Marrelli, S. Martini, E. Martines, K. McCollam, S. Menmuir, F. Milani, M. Moresco, L. Novello, S. Ortolani, R. Paccagnella, R. Pasqualotto, S. Peruzzo, R. Piovan, P. Piovesan, L. Piron, A. Pizzimenti, N. Pomaro, I. Predebon, J.A. Reusch, G. Rostagni, G. Rubinacci, J.S. Sarff, F. Sattin, P. Scarin, G. Serianni, P. Sonato, E. Spada, A. Soppelsa, S. Spagnolo, M. Spolaore, G. Spizzo, C. Taliercio, D. Terranova, V. Toigo, M. Valisa, N. Vianello, F. Villone, R.B. White, D. Yadikin, P. Zaccaria, A. Zamengo, P. Zanca, B. Zaniol, L. Zanotto, E. Zilli, H. Zohm, and M. Zuin. Overview of RFX-mod results. *Nuclear Fusion*, 49(10):104019, 2009.

[20] J.R. Drake, P.R. Brunsell, D. Yadikin, M. Cecconello, J.A. Malmberg, D. Gregoratto, R. Paccagnella, T. Bolzonella, G. Manduchi, L. Marrelli, S. Ortolani, G. Spizzo, P. Zanca, A. Bondeson, and Y.Q. Liu. Experimental and theoretical studies of active control of resistive wall mode growth in the EXTRAP T2R reversed-field pinch. *Nuclear Fusion*, 45(7):557, 2005.

[21] K. Erik J. Olofsson, Anton Soppelsa, Tommaso Bolzonella, and Giuseppe Marchiori. Subspace identification analysis of RFX and T2R reversed-field pinches. *Control Engineering Practice*, 21(7):917 – 929, 2013.

[22] S. A. Sabbagh, R. E. Bell, J. E. Menard, D. A. Gates, A. C. Sontag, J. M. Bialek, B. P. LeBlanc, F. M. Levinton, K. Tritz, and H. Yuh. Active stabilization of the resistive-wall mode in high-beta, low-rotation plasmas. *Phys. Rev. Lett.*, 97:045004, Jul 2006.

[23] J. Dalessio, E. Schuster, D.A. Humphreys, M.L. Walker, Y. In, and J.S. Kim. Robust control of resistive wall modes in tokamak plasmas using μ-synthesis. *IFAC Proceedings Volumes*, 41(2):6402 – 6409, 2008.

[24] Y. In, J. S. Kim, D. H. Edgell, E. J. Strait, D. A. Humphreys, M. L. Walker, G. L. Jackson, M. S. Chu, R. Johnson, R. J. La Haye, M. Okabayashi, A. M. Garofalo, and H. Reimerdes. Model-based dynamic resistive wall mode identification and feedback control in the diii-d tokamak. *Physics of Plasmas*, 13(6):062512, 2006.

[25] R. E. Kalman. Contributions to the theory of optimal control. *Bol. Soc. Mat. Mexicana*, 5(2):102–119, 1960.

[26] R. E. Kalman. On the general theory of control systems. *Proceedings of the 1st IFAC Congress*, 1:481–492, 1960.

[27] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17–32, Feb 1981.

[28] Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms. *IEEE Transactions on Automatic Control*, 32(2):115–122, 1987.

[29] Yi Liu and Brian DO Anderson. Singular perturbation approximation of balanced systems. *International Journal of Control*, 50(4):1379–1405, 1989.

[30] Alan J Laub. *Matrix analysis for scientists and engineers*. Siam, 2005.

[31] C. S. Hsu and D. Hou. Reducing unstable linear control systems via real schur transformation. *Electronics Letters*, 27(11):984–986, May 1991.

[32] L.L. Lao, H. St. John, R.D. Stambaugh, A.G. Kellman, and W. Pfeiffer. Reconstruction of current profile parameters and plasma shapes in tokamaks. *Nuclear Fusion*, 25(11):1611, 1985.

[33] James Bialek, Allen H. Boozer, M. E. Mauel, and G. A. Navratil. Modeling of active control of external magnetohydrodynamic instabilities. *Physics of Plasmas*, 8(5):2170–2180, 2001.

[34] A. H. Boozer. Feedback equations for the wall modes of a rotating plasma. *Physics of Plasmas*, 6(8):3180–3187, 1999.

[35] A. H Boozer. Equations for studies of feedback stabilization. *Physics of Plasmas*, 5(9):3350–3357, 1998.

[36] A. H. Glasser. The direct criterion of Newcomb for the ideal MHD stability of an axisymmetric toroidal plasma. *Physics of Plasmas*, 23(7):072505, 2016.

[37] W.A. Newcomb. Hydromagnetic stability of a diffuse linear pinch. *Annals of Physics*, 10(2):232 – 267, 1960.

[38] A. M. Garofalo, A. D. Turnbull, M. E. Austin, J. Bialek, M. S. Chu, K. J. Comer, E. D. Fredrickson, R. J. Groebner, R. J. La Haye, L. L. Lao, E. A. Lazarus, G. A. Navratil, T. H. Osborne, B. W. Rice, S. A. Sabbagh, J. T. Scoville, E. J. Strait, and T. S. Taylor. Direct observation of the resistive wall mode in a tokamak and its interaction with plasma rotation. *Phys. Rev. Lett.*, 82:3811–3814, May 1999.

[39] M. J. Lanctot, H. Reimerdes, A. M. Garofalo, M. S. Chu, Y. Q. Liu, E. J. Strait, G. L. Jackson, R. J. La Haye, M. Okabayashi, T. H. Osborne, and M. J. Schaffer. Validation of the linear ideal magnetohydrodynamic model of three-dimensional tokamak equilibria. *Physics of Plasmas*, 17(3):030701, 2010.

[40] Michael Athans and Peter L Falb. *Optimal control: an introduction to the theory and its applications*. Courier Corporation, 2013.

[41] G. F. Franklin, J. D. Powell, and M. L. Workman. *Digital Control of Dynamic Systems*. Addison-Wesley, 2nd edition, 1990.

[42] Dan Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[43] J. D. King, E. J. Strait, R. L. Boivin, D. Taussig, M. G. Watkins, J. M. Hanson, N. C. Logan, C. Paz-Soldan, D. C. Pace, D. Shiraki, M. J. Lanctot, R. J. La Haye, L. L. Lao, D. J. Battaglia, A. C. Sontag, S. R. Haskey, and J. G. Bak. An upgrade of the magnetic diagnostic system of the DIII-D tokamak for non-axisymmetric measurements. *Rev. Sci. Instrum.*, 85(8):083503, 2014.

[44] J.L. Luxon, M.J. Schaffer, G.L. Jackson, J.A. Leuer, A. Nagy, J.T. Scoville, and E.J. Strait. Anomalies in the applied magnetic fields in DIII-D and their implications for the understanding of stability experiments. *Nuclear Fusion*, 43(12):1813, 2003.

[45] E. J. Strait, J. M. Bialek, I. N. Bogatu, M. S. Chance, M. S. Chu, D. H. Edgell, A. M. Garofalo, G. L. Jackson, R. J. Jayakumar, T. H. Jensen, O. Katsuro-Hopkins, J. S. Kim, R. J. La Haye, L. L. Lao, M. A. Makowski, G. A. Navratil, M. Okabayashi, H. Reimerdes, J. T. Scoville, A. D. Turnbull, and DIII-D Team. Resistive wall mode stabilization with internal feedback coils in DIII-D. *Physics of Plasmas*, 11(5):2505–2513, 2004.

[46] Robicon. *Instruction and Maintenance Manual For 5 KA Pulse Power Supply*.

[47] Techron. *7700 Series Power Supply Amplifiers Technical Manual*.

[48] M. K. Vijaya Sankar, E. Eisner, A. Garofalo, D. Gates, T. H. Ivers, R. Kombargi, M. E. Mauel, D. Maurer, D. Nadle, G. A. Navratil, and Q. Xiao. Initial high beta operation of the HBT-EP tokamak. *Journal of Fusion Energy*, 12(3):303–310, 1993.

[49] N. Rath, J. Bialek, P.J. Byrne, B. DeBono, J.P. Levesque, B. Li, M.E. Mauel, D.A. Maurer, G.A. Navratil, and D. Shiraki. High-speed, multi-input, multi-output control using GPU processing in the HBT-EP tokamak. *Fusion Engineering and Design*, 87(12):1895–1899, 2012.

[50] N. Rath. *GPU-based, Microsecond Latency, Hecto-Channel MIMO Feedback Control of Magnetically Confined Plasmas*. PhD thesis, Columbia University, 2013.

[51] NVIDIA Corporation. Developing a linux kernel module using gpudirect rdma, 2017.

[52] Peter Benner, Volker Mehrmann, Vasile Sima, Sabine Van Huffel, and Andras Varga. *SLICOT—A Subroutine Library in Systems and Control Theory*, pages 499–539. Birkhäuser Boston, Boston, MA, 1999.

[53] Control Systems Library for Python. `http://github.com/python-control/python-control`, 2017.

[54] O. Katsuro-Hopkins. *Optimized Feedback Control System Modeling of Resistive Wall Modes for Burning Plasmas Experiments*. PhD thesis, Columbia University, 2007.

[55] NVIDIA Corporation. *CUDA C Programming Guide*. NVIDIA Corporation, 2015.

[56] Jong-Kyu Park, Allen H. Boozer, Jonathan E. Menard, Stefan P. Gerhardt, and Steve A. Sabbagh. Shielding of external magnetic perturbations by torque in rotating tokamak plasmas. *Physics of Plasmas*, 16(8), 2009.

[57] M. Newville and M. Rivers. Python Libraries from GSECARS, 2002.

[58] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[59] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Q. Appl. Math.*, 2:164–168, 1944.

[60] M. Okabayashi, J. Bialek, M. S. Chance, M. S. Chu, E. D. Fredrickson, A. M. Garofalo, M. Gryaznevich, R. E. Hatcher, T. H. Jensen, L. C. Johnson, R. J. La Haye, E. A. Lazarus, M. A. Makowski, J. Manickam, G. A. Navratil, J. T. Scoville, E. J. Strait, A. D. Turnbull, M. L. Walker, and DIII-D Team. Active feedback stabilization of the resistive wall mode on the DIII-D device. *Physics of Plasmas*, 8(5):2071–2082, 2001.

[61] J.A. Crotinger, S. W. Hanley, and T. B. Kaiser. CalTrans, 2000.

[62] C. T. Holcomb, J. R. Ferron, T. C. Luce, T. W. Petrie, P. A. Politzer, C. Challis, J. C. DeBoo, E. J. Doyle, C. M. Greenfield, R. J. Groebner, M. Groth, A. W. Hyatt, G. L. Jackson, C. Kessel, R. J. La Haye, M. A. Makowski, G. R. McKee, M. Murakami, T. H. Osborne, J.-M. Park, R. Prater, G. D. Porter, H. Reimerdes, T. L. Rhodes, M. W. Shafer, P. B. Snyder, A. D. Turnbull, and W. P. West. Optimizing stability, transport, and divertor operation through plasma shaping for steady-state scenario development in diii-d. *Physics of Plasmas*, 16(5):056116, 2009.

[63] X N Yue, B J Xiao, Z P Luo, and Y Guo. Fast equilibrium reconstruction for tokamak discharge control based on gpu. *Plasma Physics and Controlled Fusion*, 55(8):085016, 2013.

[64] J.R. Ferron, M.L. Walker, L.L. Lao, H.E. St. John, D.A. Humphreys, and J.A. Leuer. Real time equilibrium reconstruction for tokamak discharge control. *Nuclear Fusion*, 38(7):1055, 1998.

[65] E. J. Strait, J. D. King, J. M. Hanson, and N. C. Logan. Spatial and temporal analysis of DIII-D 3D magnetic diagnostic data. *Review of Scientific Instruments*, 87(11):11D423, 2016.

[66] Andreas Klöckner, Nicolas Pinto, Yunsup Lee, Bryan Catanzaro, Paul Ivanov, and Ahmed Fasih. PyCUDA and PyOpenCL: A Scripting-Based Approach to GPU Run-Time Code Generation. *Parallel Computing*, 38(3):157–174, 2012.

[67] H. H. B. Sørensen. *High-Performance Matrix-Vector Multiplication on the GPU*, pages 377–386. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.