A REAL-TIME GRAD-SHAFRANOV PDE SOLVER AND MIMO CONTROLLER USING GRAPHICAL DATAFLOW PROGRAMMING

A.Barp[a], M.Cerna[a], S.Concezzi[a], L.Giannone[b], G.Morrow[a], Q.Ruan[a,], A.Veeramani[a], L.Wenzel[a]

[a]*National Instruments, Austin, TX 78759-3504, Texas, USA*
[b]*Max-Planck-Institute for Plasma Physics, EURATOM-IPP Association, D-85748 Garching, Germany*

Abstract

Plasma control experiments require enormous computational power to solve large problems with critical time constraints. For tokamak control, the non-linear and constrained Grad-Shafranov equation needs to be solved in real-time with a cycle time of less than 1 ms. A new algorithm for the solution of this equation based on discrete sine transforms and a tridiagonal solver rather than the commonly used cyclic reduction algorithm is presented. Input signals from magnetic probes and flux loops are the constraints for the equation that must be continuously solved to calculate the magnetic equilibrium. A number of novel mathematical ideas were introduced and several generally applicable numerical strategies were developed using LabVIEW graphical dataflow programming to meet the critical timing goals. Benchmarks on CPUs are reported. Furthermore, the design of a MIMO controller to demonstrate the possibilities of tokamak position and shape control using graphical dataflow programming is discussed.

Keywords: Plasma Control, Grad-Shafranov Equation, MIMO Controller, Real-time, LabVIEW

1. INTRODUCTION

Nuclear fusion requires the interaction of several control systems. The control parts that deal directly with the plasma are the magnetic control system and the internal control system. The former is responsible for dealing with global plasma properties such as plasma current, position and shape while the latter focuses on disruption mitigation or the stabilization of MHD modes. We only address a specific aspect of shape control in the paper leveraging a highly efficient Grad-Shafranov solver.

The magnetic equilibrium for a tokamak is described by the Grad-Shafranov equation :

$$R\frac{\partial}{\partial R}\left(\frac{1}{R}\frac{\partial \psi}{\partial R}\right)+\frac{\partial^2 \psi}{\partial Z^2} = -\mu_0 R\mathbf{j}(R,Z), \qquad (1)$$

where ψ is the poloidal flux function, $\mathbf{j}$ is the current density, R is the radial component and Z is the axial component (see figure 1). This problem is commonly solved by a cyclic reduction algorithm [1, 2, 3]. A magnetic equilibrium for discharges with plasma current is reconstructed on a 33 x 65 grid using 40 magnetic probes and 18 flux loop difference signals. The right hand side current density term is calculated by a weighted least squares fit to the measurements which yields coefficients for the basis current density profiles [2, 3, 4]. Three basis current density profiles were chosen in the first round of development and found to adequately fit the experimental magnetic probe and flux loop measurements [5]. The currents from the poloidal field coils are also needed to compute the value of ψ on the spatial grid.

Plasma shape control using the isoflux method is typically done with the aid of a collection of PID controllers. The plasma shape in DIII-D was reconstructed using real-time magnetic probe measurements and represented by certain points along the shape. PID controllers maintain well-defined locations (setpoints) for all those points, one PID controller per setpoint [2]. A MIMO (multiple input and output) controller that replaces the independent PID controllers is presented and initial benchmarks are provided. It is worth noting that the the MIMO controller is redesigned in-the-loop without significant performance degradations. This work complements the normalized coprime factorization design MIMO controller in operation on DIII-D [6, 7].

The entire project leveraged LabVIEW's capability to run in a strict real-time setting and its effectiveness as a development tool. LabVIEW's graphical dataflow

Email address: qing.ruan@ni.com (Q.Ruan)

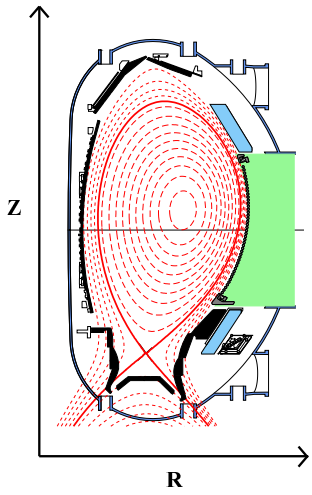

Figure 1: The cross section of the ASDEX Upgrade tokamak showing the flux surfaces of the magnetic equilibrium (red dotted lines) and plasma separatrix (red solid line).

paradigm makes developers more efficient by offering an intuitive diagram-model approach with convenient probe-based debugging and built-in measurement visualization. A major benefit of programming graphically is that it allows engineers and scientists to produce modular real-time algorithms and applications.

2. REAL TIME GRAD-SHAFRANOV SOLVER

A spectral-based method commonly used to solve the Poisson equation in cylindrical coordinates was adapted to solve the Grad-Shafranov equation in an unbounded domain. Our approach is to use multi-channel discrete sine transforms (DST) along the Z-axis and a tridiagonal solver [8, 9] as an alternative to the cyclic reduction algorithm to solve the Grad-Shafranov equation for poloidal flux, ψ. This algorithm has been parallelized and benchmarks as a function of grid size on different CPUs have been reported [10].

2.1. Spectral Method on Bounded Domain

A uniform mesh with constant spacing dR and dZ in the R and Z directions is assumed. The grid points are labeled from 0 to $NZ-1$ and 0 to $NR-1$, where NZ is the number of grid points in the Z direction, and NR is the number of points in the R direction. The five point difference equation with index i in the R direction and index j in the Z direction can be written as :

$$\frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{dR^2} - \frac{1}{R_i}\frac{\psi_{i+1,j} - \psi_{i-1,j}}{2dR} + \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{dZ^2} = -\mu_o R_i \mathbf{j}_{i,j} \quad (2)$$

Introducing the discrete sine transform of ψ and $\mathbf{j}$:

$$\phi_{i,k} = \sum_{j=1}^{NZ-2} \psi_{i,j} sin\left(\frac{\pi jk}{NZ-1}\right) \quad (3)$$

$$J_{i,k} = \sum_{j=1}^{NZ-2} \mathbf{j}_{i,j} sin\left(\frac{\pi jk}{NZ-1}\right) \quad (4)$$

leads to the tridiagonal matrix equations :

$$\beta_i \phi_{i+1,k} - \alpha_k \phi_{i,k} + \gamma_i \phi_{i-1,k} = -\mu_0 R_i dR^2 J_{i,k} \quad (5)$$

where $\alpha_k = 2 + 4S^2 sin^2\left(\frac{\pi k}{2(NZ-1)}\right)$, $\beta_i = 1 - dR/(2R_i)$, $\gamma_i = 1 + dR/(2R_i)$ and $S = dR/dZ$.

The tridiagonal matrix equation is solved with a tridiagonal solver using an LU decomposition algorithm. The LU decomposition generates two bidiagonal matrices subsequently used in the iterative procedure to solve the tridiagonal equations. By using LU decomposition, operations are reduced by a factor of 2 compared to the direct solver algorithm [11].

2.2. Solver on Unbounded Domain

The solver for the Grad-Shafranov equation in an unbounded domain is composed of two fast solver steps [1]. The new algorithm reduces the computing time dramatically by utilizing the above spectral method at each step.

In the first step of the solver, all grid boundaries are set to zero. The right-hand side is set to the current distribution on the flux surfaces, as computed in the previous iteration by a weighted least squares fit to magnetic probe and flux loop measurements. In this step, it is only necessary to compute ψ at points neighboring the grid boundary and a reduced inverse DST can be performed to calculate these values. The columns of ψ inside the boundary edge are :

$$\psi_{i,k} = \frac{2}{NZ-1} \sum_{j=1}^{NZ-2} \phi_{i,j} sin\left(\frac{\pi jk}{NZ-1}\right) \quad (6)$$

where $i = 1$ and $NR-2$, and the rows inside the boundary edge can be calculated in a similar fashion with $k = 1$ and $NZ-2$. All these four edges can be computed using matrix-vector multiplication. This avoids the unnecessary computations performed by a traditional inverse DST operation applied to the entire grid. The gradients in ψ normal to the grid boundary, $(\partial\psi/\partial n)_{boundary}$, are the inputs required for the next solver step. These are the shielding currents that are necessary to force the zero boundary condition of the first solver step. They

are used to calculate the Green's functions for ψ generated by a current hoop of radius a, carrying current I, for each grid point with radial coordinate R, and a vertical distance Z, on the boundary [1, 12, 13] :

$$\psi = \mu_o I \sqrt{(a+R)^2 + Z^2)}((1-k^2/2)K(k^2) - E(k^2)) \quad (7)$$

where $k^2 = 4aR/((a+R)^2 + Z^2))$, $K(k^2)$ is the complete elliptic integral of the first kind and $E(k^2)$ is the complete elliptic integral of the second kind [14]. The actual calculation of the resulting ψ on the boundary is performed as a matrix multiplication with pre-calculated coefficients times the vector of shielding currents.

The second step of the solver is carried out with boundary conditions from the first solver step but without current source terms on the right hand side of the Grad-Shafranov equation. Because only the first and last elements are nonzero, it is possible to use an optimized DST to reduce the computation effort. The faster DST is carried out by the BLAS function *dger* producing :

$$\begin{aligned} D_{ij} &= -\frac{\psi_{i,1} sin\left(\frac{\pi j}{NZ-1}\right) + \psi_{i,NZ-2} sin\left(\frac{\pi j(NZ-2)}{NZ-1}\right)}{dZ^2} \\ &= -\frac{\psi_{i,1} - (-1)^j \psi_{i,NZ-2}}{dZ^2} sin\left(\frac{\pi j}{NZ-1}\right) \end{aligned} \quad (8)$$

The DST of the boundary conditions at the inner and outer radial positions are added to the first and last columns. The tridiagonal solver is applied to this result and is added to the result from the first solver step. The solution of the Grad-Shafranov equation is then calculated by an inverse DST.

Under equivalent boundary conditions, an implementation based on the cyclic reduction algorithm computes all elements on the grid in both solver steps. The Grad-Shafranov solver algorithm described here achieves a significant performance improvement in comparison to cyclic reduction by employing two optimized DST implementations. The first implementation exploits the ability to avoid unnecessary calculations. The second implementation exploits the fact that the right hand side term is zero except at the boundary to greatly reduce the number of operations.

The ψ generated by the external poloidal field coils and passive stabilizing loop on the grid is also realized as a matrix-vector multiplication using factors calculated with Equation 7. The poloidal field coils and passive stabilizing loop are simulated as a finite number of filaments, with each filament carrying an applicable number of turns. Vacuum field shots with current pulses successively in each of the poloidal field coils are carried out to ensure that the best possible estimates of the magnetic probe and flux loop positions and calibration factors of the integrators are used to reconstruct the tokamak magnetic equilibrium with plasma current [5]. A steepest descent algorithm is available to optimize the position and orientation of the probes, the integrator time constants, the poloidal field coil current measurements and the position of the poloidal field coils in order to minimize the difference between measured and calculated probe response to these vacuum field discharges.

A flux matrix compression has been implemented to make the 33x65 values of ψ available to diagnostics on the real-time network that have only a UDP connection. The algorithm constructs 253 coefficients from discrete cosine transforms of the flux matrix.

3. MIMO CONTROLLER

MIMO controllers for tokamaks are discussed in several papers. One approach is to obtain a linear model of the system around the operating point and design a controller based on this operating point and repeating this process for any operating point [15] . In this approach one starts with the linear model as $M\dot{x} + Px = u$, where M, P and u are defined in [15] (equations 2.11 and 2.13). Then the system is cast in the standard state-space model form $\dot{x} = Ax + Bu$ where the state space vector, x , is defined by :

$$x = \left[\; I_s - I_s^o, (z - z^o) I_p^o \, , (R - R^o) I_p^o, I_p - I_p^o \; \right] \quad (9)$$

where R is the radial and z is the axial value of the plasma axis, I_p is the plasma current and I_s are currents in the poloidal field coils and passive stabilising coils. The set point values have the superscript o.

A candidate controller design is based on a Linear Quadratic Regutalor (LQR). To verify the feasibility of redesigning the controller on-the-fly to compensate for changes on the operating point, the following experiment was performed :

1. Choose N_a and N_b sizes ($N_a, N_b \leq 10$ for now).
2. Let M and P be N-by-N matrices. $N = 2N_a + N_b + 1$.
3. Calculate $A = -M^{-1}P$, $B = M^{-1}$.
4. Assume C = Identity (everything is measured). ???????? second state equation not mentioned
5. Compute an LQR with freely available design goals (for example avoiding coil voltage limits or minimising power consumption).
6. Generate K.??????? a LabVIEW output related to setting u / controller gains (reference to CD Toolkit vi used!) ?????

The benchmark results with different sizes are listed in the next section.

4. IMPLEMENTATION AND BENCHMARKS

A Dell T5500 (2xXeon 5677 quad core CPUs) with LabVIEW RT 2011 is used by the magnetics diagnostic on ASDEX Upgrade for the data acquisition of 220 channels sampled at 10 kHz. The cycle time benchmarks for the real time Grad-Shafranov solver using magnetic probes with 58 constraints and 3 basis current functions was 0.49 ms on 4 cores. Extending the solver to include 10 MSE constraints with 6 basis current functions increases the cycle time to 0.80 ms on 4 cores. In the latter benchmark, both solutions are calculated in parallel, as it is a control system requirement that the flux matrix from the equilibrium reconstruction with magnetic probes only is immediately available when MSE measurements are no longer possible.

The achieved cycle time for the Grad-Shafranov solver is therefore satisfactory for the real-time processing requirements of neoclassical tearing mode stabilization experiments where the cycle time of the discharge control system is 1.3 ms [16]. It should be noted that these benchmarks are for a single cycle iteration for the PDE solution. A detailed comparison of real-time magnetic equilibrium reconstruction with well converged solutions from offline calculations show that the small differences found for relatively steady state conditions are not relevant for practical discharge control [2].

The benchmarks for redesigning MIMO controllers described before are displayed in Table 1.

# of points (N_a)	**# of coils** (N_b)	**LQR re-design [ms]**
10	10	4
10	6	3
9	6	2.6
8	6	2.2
7	6	1.8
6	6	1.4

Table 1: Times in msec for redesigning MIMO controllers depending on the number of points that represent the shape and depending on the number of coils (actuators).

It should be noted that all of these benchmark times are in the low millisecond range. This enables a real-time MIMO controller design to adjust to highly nonlinear situations.

5. SUMMARY

A real-time Grad-Shafranov solver on unbounded domain based on a spectral method rather than cyclic reduction has been realized. The resulting tridiagonal equations are solved with a specially developed subroutine based on LU factorization. This tridiagonal solver reduces the number of operations with respect to the iterative direct solver by pre-calculating the reciprocal of the diagonal elements. A reduced inverse DST is required in the first solver step as only the relevant terms for those neighbors of the grid boundary need be calculated. A simplified DST can used for the second solver step where only the first and last elements are non-zero. In this way the full inverse DST of the first solver step is omitted and the DST of the second solver step without current source terms can be calculated with a smaller number of operations. The real-time Grad-Shafranov solver cycle time of 0.49 ms on the delivered Dell T5500 platform satisfies the ASDEX Upgrade real-time processing requirements.

The MIMO controller approach can greatly increase the control power of a closed loop system. Furthermore, the ability to redesign the controller in just a few milliseconds means that it can effectively adapt to highly nonlinear situations.

References

[1] K.Lackner, Comp. Phys. Comm., **12**, 33 (1976)

[2] J.R.Ferron, M.L.Walker, L.L.Lao et. al, Nuc. Fusion **38**, 1055 (1998)

[3] P.J. McCarthy, Physics of Plasmas, **6**, 3554 (1999)

[4] W.Zwingmann et. al, Plasma Phys. Control. Fusion, **43**, 1441 (2001)

[5] L.Giannone, R.Fischer, J.C.Fuchs et. al, http://ocs.ciemat.es/EPS2010PAP/pdf/P4.122.pdf

[6] M.L.Walker, D.A.Humphreys, R.D.Johnson and J.A.Leuer, Fusion Science and Technology **47**, 790, (2005)

[7] M.L. Walker D.A. Humphreys, J.A. Leuer, J.R. Ferronand B.G. Penaflor , Fusion Eng. Design **56-57**, 727, (2001)

[8] R.W.Hockney and J.W.Eastwood, "Computer simulation using particles", p208, Taylor and Francis (1988)

[9] M.H.Hughes, Comp. Phys. Comm., **2**, 157 (1971)

[10] R. Preuss, R. Fischer, M. Rampp, K. Hallatschek, U. von Toussaint, L. Giannone, P. McCarthy, *Parallel equilibrium algorithm for real-time control of tokamak plasmas*, IPP-Report R/47 (2012)

[11] W.H.Press, S.A.Teukolsky, W.T.Vetterling and B.P.Flannery, "Numerical Recipes in C", p50, Cambridge University Press, (1999)

[12] J.D.Jackson, "Classical Electrodynamics", p142, Wiley, (1999)

[13] J.Simpson, J.Lane, C.Immer and R. Youngquist, Simple analytic expressions for the magnetic field of a circular current loop http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20010038492001057024.pdf

[14] M.Abramowitz and I.A.Stegun, "Handbook of Mathematical Functions", p588/591, National Bureau of Standards (1972)

[15] J.B. Lister, A. Sharma, D.J.N. Limebeer, Y. Nakamura, J.P.Wainwright, R. Yoshino et. al, Nuc. Fusion **42**, 708 (2002)

[16] W.Treutterer, L.Giannone, K.Lüddecke et. al, Fusion Eng. Des. (2008) http://dx.doi.org/10.1016/j.fusengdes.2008.12.026