GA–A23152

AN ALGORITHM TO PROVIDE REAL TIME NEUTRAL BEAM SUBSTITUTION IN THE DIII–D TOKAMAK

by
J.C. PHILLIPS, K.L. GREENE, A.W. HYATT, B.B. McHARG, Jr.,
and B.G. PENAFLOR

JUNE 1999

DISCLAIMER

GA–A23152

AN ALGORITHM TO PROVIDE REAL TIME NEUTRAL BEAM SUBSTITUTION IN THE DIII–D TOKAMAK

by
J.C. PHILLIPS, K.L. GREENE, A.W. HYATT, B.B. McHARG, Jr., and B.G. PENAFLOR

This is a preprint of a paper to be presented at the 11th IEEE/NPSS Real Time Conference, June 14–18, 1999, Santa Fe, New Mexico and to be published in *Transactions on Nuclear Science*.

Work supported by
the U.S. Department of Energy
under Contract No. DE-AC03-99ER54463

GA PROJECT 30033
JUNE 1999

An Algorithm to Provide Real Time Neutral Beam Substitution in the DIII–D Tokamak*

J.C. Phillips, K.L. Greene, A.W. Hyatt, B.B. McHarg, Jr., and B.G. Penaflor
General Atomics, P.O. Box 85608, San Diego, California 92185-5608

Abstract

A key component of the DIII–D tokamak fusion experiment is a flexible and easy to expand digital control system which actively controls a large number of parameters in real-time. These include plasma shape, position, density, and total stored energy. This system, known as the PCS (plasma control system), also has the ability to directly control auxilliary plasma heating systems, such as the 20 MW of neutral beams routinely used on DIII–D.

This paper describes the implementation of a real-time algorithm allowing substitution of power from one neutral beam for another, given a fault in the originally scheduled beam. Previously, in the event of a fault in one of the neutral beams, the actual power profile for the shot might be deficient, resulting in a less useful or wasted shot. Using this new real-time algorithm, a stand by neutral beam may substitute within milliseconds for one which has faulted. Since single shots can have substantial value, this is an important advance to DIII–D's capabilities and utilization.

Detailed results are presented, along with a description not only of the algorithm but of the simulation setup required to prove the algorithm without the costs normally associated with using physics operations time.

I. INTRODUCTION

The DIII-D tokmak at the National Fusion Facility in San Diego is a pulsed experimental device, producing results in support of fusion as a source of energy. Plasma discharges occur on up to an 8 minute repetition period for the duration of a typical 8 hour operating day. Each discharge produces almost 300 MB of data, a figure which steadily increases. To provide auxiliary heating to the plasma, neutral particle beams (along with various modes of rf power) are employed, for a total neutral beam power of up to 20 MW at pulse lenths of up to 5 s. The fixed costs of operating the facility, the unique nature of the apparatus, as well as the need to take time both for maintenance activities and for upgrades and improvements, mean that the actual operation in support of physics experiments assumes a high value. Steps that can be taken to improve the utilization of the facility, or to yield improved results from the experimental discharges, translate into significant cost savings or an increase in value received.

II. APPLICATION

By their very nature, the neutral beams are subject to high voltage spark-downs. Normally, the ionized particles are accelerated to energies of up to 80,000 V, with gaps as small as 0.25 cm in between them. Anything but pristine vacuum conditions and surfaces can result in a high voltage spark-down, and in such cases the power supplies act to protect both the load and themselves by interrupting the high voltage for 15 ms, followed by an attempt to re-establish the potential difference. Should similar conditions again be detected, the process is repeated, so there may be arbitrary periods of time ranging from 15 ms to as long as the neutral beam pulse was intended (some 4 or 5 s) where the power normally delivered by that ion source is missing (Fig. 1). The time and duration of these events cannot be predicted in advance, so any means to remedy the situation must depend on detection and mitigation in real time.

To mitigate this problem, an algorithm to provide a means for substituting an available beam for a failed beam has been implemented on the plasma control system (PCS) computer. The algorithm responds in real time, and acts to salvage what otherwise might be an underutilized or wasted experimental plasma discharge. Figure 2 shows results using the algorithm, yielding an almost perfect injected power profile to DIII-D.

III. PLATFORM

The hardware platform used for implementation of this algorithm has been well described in the literature [1], and is briefly summarized here. Known as the PCS, it is comprised of eight parallel computers, CSPI i860 based Supercards, with an associated SUN SPARC as a host. Additional hardware allows the digitization of 128 channels of analog information

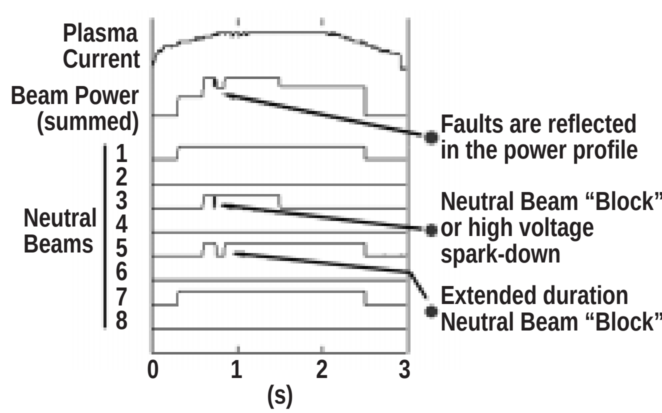

Fig. 1. A typical plasma discharge, with four of eight neutral beams being requested. A brief “block,” or high voltage fault, is evident for Beam 3 number, and a more extensive one for beam 5. These are translated to the summed injected power as a deficit, yielding less than perfect results.

*Work supported by U.S. Department of Energy under Contract No. DE-AC03-99ER54463.

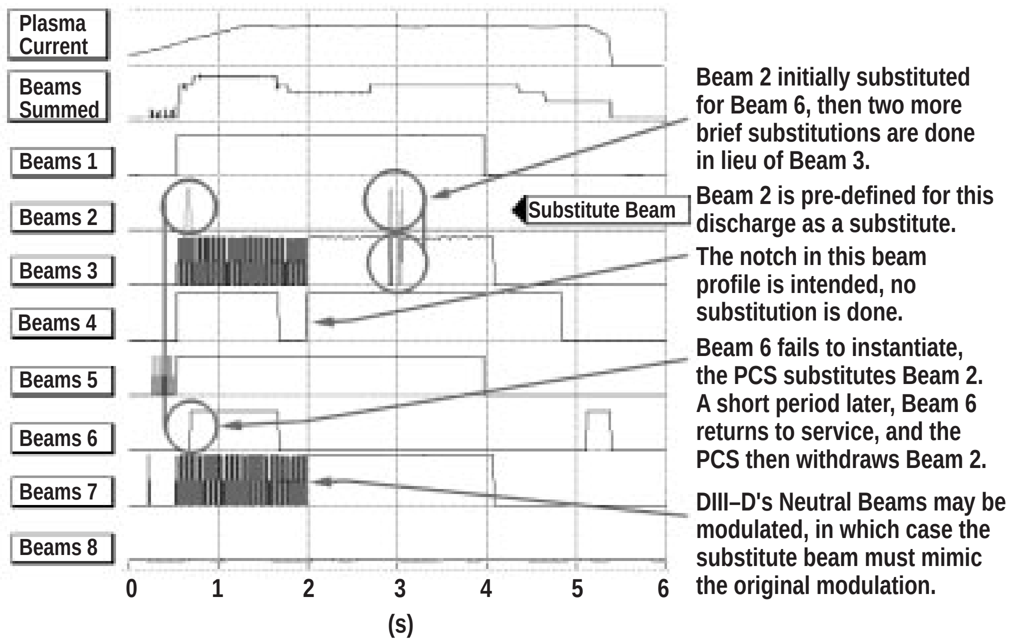

Fig 2. A plasma discharge with neutral beam substitution evident.

at a 100 kHz rate per channel (Fig. 3). These data are assimilated by a FIFO board attached as a daughtercard to each supercard. Data from DIII–D shots are archived and may be retrieved for analyis or display, and in the case of the PCS, may also be used to “replay” historic discharges off-line. This allows use of the hardware as a simulator to verify and develop new PCS algorithms. While these simulations do take longer than real time to complete, they can be an extremely valuable development tool.

IV. ALGORITHM

Algorithms are implemented on the PCS such that, for each time-slice, the input values are examined, applied to functions or control models, then the controlled outputs are set accordingly. The process is repeated for the next time-slice. The cycle time for the PCS during a DIII–D discharge is on the order of 100 µs. The beam substitution algorithm along with others, must come to completion within that timeframe. Application code, generally written in “C,” is compiled and downloaded to the CSPI i860 SuperCards for execution [3]. The algorithm has as inputs an array representing the current state of the Neutral Beams (either on or off) along with a representation of the requested beam profile as entered by the physicists in charge. The physicist, in conjunction with the neutral beam coordinator, typically declares a subset of the neutral beams to be “primary” to the success of the discharge, along with some few beams which are not needed being declared as “available for substitution.” Beams which are to be used as substitutes *must* be controlled by the PCS. Neutral beams declared to be primary are usually *not* controlled by the PCS, which merely monitors their states. For each “primary” beam these declarations, along with the states of various external timing control hardware, form the basis for the rest of the inputs to the algorithm (Table 1).

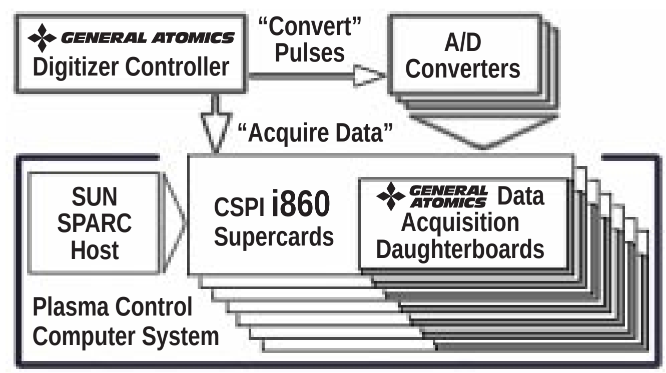

Fig 3. A block diagram of DIII–D’s plasma control system computer.

If events or status needed as input are not represented by the acquired data and associated time-history, they must be allocated static memory and be maintained by the algorithm or function itself. In this class are such things as counters which provide for detecting whether a neutral beam has been “absent” sufficient cycles to be declared “missing,” as well as a similar counter invoked when a neutral beam unexpectedly returns to service in mid-discharge. In addition to handling these inputs, the algorithm produces time ordered data to be archived along with the analog inputs to the PCS from the DIII–D discharge.

The algorithm has structures (priority queues) implemented to make manageable the task of selecting an appropriate beam for substitution. One benefit to this approach is to pre-load the work which must be done. Rather than sequencing through a complicated series of case statements each time a substitution is made, work is instead done when the program is initiated to establish the type and priority of beams available for substitution. To illustrate, if a beam is

Table 1. Simplified Input, Output, and Static data storage for the Neutral Beam Substitution algorithm

Input Data	Output Data	Static Storage
Accelerator Voltage Waveforms, "As Running"	Overall Record of the Beam Requests	Pointers Mapping Virtual to Actual Beams
Requested Beam Power Profiles	Record of Real-Time Substitution	Counter Declaring a Requested Beam as Missing
Hardware Settings for Modulation		Counter Declaring a Requested Beam as Restored

used as a substitute during a portion of the discharge, but then becomes available due to a return to service of the failed beam, this is handled within the queues. In this case, de-queued from the "available substitute" queue, placed in service, and then re-queued as an available substitute. To simplify the daunting housekeeping task of managing which beam is being used in which manner, a map of "real" beams to "virtual" beams is kept. For example, if the primary critical beams are Beams 1, 7, and 8 – with Beams 3 and 4 available as substitutes, the discharge starts with an identical map of real primary beams to virtual primary beams: Beam 1→[Beam 1], Beam 7→[Beam 7], Beam 8→[Beam 8]. In the hypothetical case that Beam 1 beam were to fail in mid discharge, and Beam 3 were substituted by the PCS, the mapping would become: Beam 1→[Beam 3], Beam 7→[Beam 7], Beam 8→[Beam 8]. If Beam 1 then suddenly returns to service, the algorithm's task is to 1) count for a few cycles to verify that the sensed event is not noise 2) to re-map the "usage" of the beams to indicate that the "virtual" Beam 1 and the "real" Beam 1 are once more one and the same, and finally to set the control outputs of the PCS accordingly. To ease the task of checking for the return of a beam previously declared missing, a simple scan of the eight position real-virtual array indicates whether a particular beam is healthy or actually a surrogate, simply by testing the equality of the real and virtual value of that beam. For each time-slice the logic handling "back from the dead" events assumes the short and appealing form: "If the virtual and real beam numbers are not the same (implying substitution has previously occurred) check the status of the real beam for that position, and if healthy, undo the substitution."

V. Simulation and Testing

The DIII–D Tokamak operates in a pulsed mode, approximately 18 weeks per year, the remainder of the time being dedicated to maintenance, calibration, data analysis, and upgrade activities. When operating, the cost of power and ancillary services such as cryogenics, mean that every operating day has high value, and physicists from many institutions vye for limited experimental time. Thus, using machine time for extensive testing is an expensive proposition, and indicates the use of some other means for testing and debug of new codes. The PCS computer system has the ability to run in "simulation" mode. In this mode, data archived from a previous DIII–D pulse are fed to the PCS via a "simulation server" - as if the shot were occuring in the present. Although the PCS runs more slowly in this mode than during a real discharge (a 15 second pulse may take 5 or 6 minutes to simulate) the method does allow for off-line testing. But since the test setup uses the actual PCS processors, this type of simulation can only be done in off weeks, not during operations. To improve the situation, an additional simulator has been developed using one i860 processor subscribing to the discharge simulation data in the same manner. This simulator is useful for cases where the algorithm is not dependent upon the parallel processing capabilities of the full PCS, and has been used for simulating the neutral beam substitution algorithm.

VI. Conclusions and Goals

Neutral Beam substitution has clearly demonstrated its ability to "rescue" an otherwise flawed discharge. That said, the solution outlined here is not a panacea. On DIII–D there are neutral beams which, due to certain diagnostic experiments being dependent on them, are in particular favor for the additional data they produce. For some experimental needs, substituting for Beam 3 for Beam 1 may yield disappointing diagnostic results, even while yielding an almost perfect power profile. This is indeed the reason for returning to primary beams to service during the discharge when possible.

The substitution algorithm was implemented in a stage-wise manner, first with only simple substitution being available ("when a beam fails it's replaced by the first available substitute"). Improvements to the algorithm have included the ability to handle a failed substitute (substitution for substitution) and the ability to substitute with constraints. The most notable being to constrain the algorithm such that only certain classes of beam be used to substitute within the same class - for instance on the basis of orientation with respect to the plasma. As a logical extension to this, substitution may be constrained to substitute only a specific beam. Further refinements to the algorithm's capabilities are to be expected, and given a virtually boundless set of possible outcomes (in terms of small deficits in the neutral beam heating profile for a requested discharge) the algorithm continues to be tested and proved each operating day.

Acknowledgements

The author wishes to acknowledge that the work presented here, relies upon the significant contributions of many others at DIII–D including John Ferron and Mike Walker, whose pioneering work with the PCS has provided the basis for extending its capabilities to algorithms such as this.

REFERENCES

[1] B.G. Penaflor, *et al*., "A Structured Architecture for Advanced Plasma Control Experiments," Proc. of the 19th Symp. on Fusion Technology, Lisbon, Portugal, Vol. 1 (1996) p. 965.

[2] M.L. Walker, *et al.*, "Status of DIII–D Plasma Control," Proc. 16th IEEE/NPSS Symp. on Fusion Eng., Champaign, Illinois, Vol. 2, p. 885 (1996).

[3] J.R. Ferron, *et al.*, "A Flexible Software Architecture for Tokamak Discharge Control Systems," Proc. 16th IEEE/NPSS Symp. on Fusion Eng., Champaign, Illinois, Vol. 2, p. 870 (1996).