Greater than 10x Acceleration of fusion plasma edge simulations using the Parareal algorithm

D. Samaddar
CCFE, Culham Science Centre,UK
Email: Debasmita.Samaddar@ccfe.ac.uk & dsamaddar@alaska.edu

D.P. Coster
Max-Planck-Institut für Plasmaphysik,
Germany

X. Bonnin
CNRS-LIMHP, Université Paris 13,
France

C. Bergmeister
CCFE, Culham Science Centre,
UK

E. Havlíčková
CCFE, Culham Science Centre,
UK

L.A. Berry, W.R. Elwasif and D.B. Batchelor
Oak Ridge National Laboratory
Oak Ridge, USA

Abstract—**Simulations involving the edge of magnetically confined, fusion plasma are an extremely challenging task. The physics in this regime is particularly complex due to the presence of neutrals as well as the wall. These simulations are extremely computationally intensive but are key to rapidly achieving thermonuclear breakeven on ITER-like machines. Space parallelization has hitherto been the most common approach to the best of our knowledge . Parallelizing time adds a new dimension to the parallelization thus significantly increasing computational speedup and resource utilization. This poster describes the application of the parareal algorithm to the edge-plasma code package SOLPS. The algorithm requires a coarse (G) and a fine (F) solver and optimizing G is a challenge. This work explores two approaches for G leading to computational gain ¿10 which is significant for simulations of this kind. It is also shown that an event-based approach to the algorithm greatly enhances performance.**

I. INTRODUCTION

Edge plasma simulations for fusion devices are more complex than the plasma core due to a variety of reasons. The presence of impurities is dominant in this region which reduce plasma heating and may also be responsible for disruptions. These simulations need to include neutral particle transport along with the transport of charged particles. Simulating the plasma edge is extremely important to understand plasma-wall interactions and hence, designing of the wall. However, they often require a huge wallclock time of the order of a few days to a few months.

Temporal parallelization adds a new dimension to the parallelizability thus rapidly enhancing computational gain. The parareal algorithm has been implemented in this work to achieve gains greater that 10 which is significant for such complex systems.

II. SOLPS CODE PACKAGE

The SOLPS (Scrape Off Layer Plasma Simulator) code package [1]consists of the B2.5 code and Eirene. B2.5 is a multifluid transport code solving the Braginskii equations. Eirene uses Monte-Carlo treatment to simulate neutral particle transport. Coupling Eirene with B2.5 allows realistic simulations, but the computational time is significantly increased.

III. PARAREAL ALGORITHM

Since its introduction in 2001 [2] the parareal algorithm has been applied to a variety of problems including turbulence [3]. However, this is the first time that the algorithm is implemented in a transport code simulating the plasma edge. The complex dynamics of the system and strong sensitivity to initial values makes this a unique application. The parareal algorithm is implemented by splitting the entire time series into parts or 'time slices', where each time slice is solved across an individual processor - in parallel. The approach apparently violates causality and is counter intuitive, but that may be achieved using the predictor-corrector technique. The algorithm, described in detail in [2], [3], requires a coarse solver (G) as a predictor and a fine solver (F) as the corrector. For each parareal iteration, k, G and F are performed. G is a serial process yielding a coarse estimate in short wallclock time. F is computationally slow but yields a more accurate solution. F is applied in parallel.

Optimizing the choice for the coarse solver is a challenge. G needs to be significantly faster than F in order to generate a desirable computational gain. On the other hand, if G is very 'coarse', that is, if it's solution is very different from that of F, many parareal iterations will be required to achieve convergence.

IV. RESULTS

Two approaches for the coarse estimate or G have been explored in this poster. For the first case, Eirene was replaced by a neutral fluids model. This certainly compromised accuracy but largely reduced wall clock time. A reduced grid model accompanied by larger time steps (dt) was used as a second approach. In both cases, it was observed that parareal convergence was sensitive to the size of the time slice chosen.

A. *G with neutral fluids model*

Replacing Eirene made the coarse solver faster by about 49 times. The plots of density (nesepm) for fine and coarse computations are shown in figs. 1 and 2 respectively. The coarse estimate is different from the fine run.

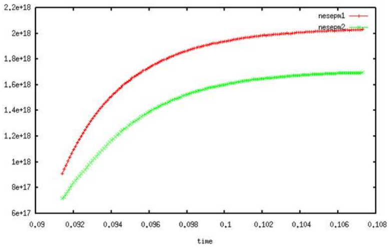

Fig. 1. Density calculated using the SOLPS-Eirene package (fine solver).

The parareal solution evolves with the iterations k as is illustrated in fig. 3. The final solution at $k = 12$ matches the fine solution of fig. 1. A gain of 12.58 was obtained with 240 processors for this case, where each processor solved a timeslice of 10 steps. Parareal convergence was not observed for slices > 20 and < 5.

B. *G with reduced grid model model*

Results obtained from the second approach to the coarse solver are illustrated in fig. 4. Two sets of tokamak plasma were used for this study. The first was MAST plasma where fine model used a grid of 150×36. The second was DIIID plasma with the fine grid being 96×36. Various reduced grid sizes were explored for both cases. For a fine grid of 150×36, sizes such as 150×18, 76×36 and 76×18 were explored as coarse models. For the fine grid of 96×36, reduced models used were 48×36 and 32×36. Reducing the grid size allowed bigger timesteps without violating the CFL condition. It was once again observed that the size of time slice solved per processor affected the iterations required for convergence. With the reduced grid models in G, a computational gain of 32 with 64 processors was achieved for the MAST plasma and a gain of 21.8 was obtained with just 96 processors for the DIIID plasma.

V. EVENT BASED PARAREAL USING IPS

This entire work was performed using the IPS framework [4], [5] written in python which allows event based implementation of the algorithm. It has been shown in earlier work [4] that this technique largely enhances performances. This feature of the IPS framework was once again observed in this work and fig. 5 illustrates this point.

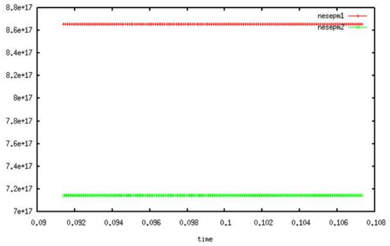

Fig. 2. Density computed where Eirene is replaced by fluid neutral model.

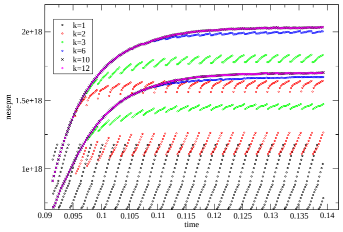

Fig. 3. The plot of density at the separatrix plotted for various parareal iterations, k. At $k = 12$, the solution matches that in fig.1.

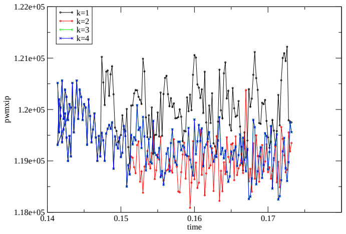

Fig. 4. The parareal algorithm converges in 4 iterations for this case where the coarse model is a reduced grid of 150×18 at $dt_G = 10dt_F$.

ACKNOWLEDGMENT

This work has been funded by the European Fusion Development Agreement (EFDA) under $WP13-SOL-01-01/CCFE/PS$. The authors are also extremely thankful to the consultants at the ITM-Gateway at IPP, Garching, Germany.

REFERENCES

[1] R. Schneider et al, Contrib. Plasma Phys. 46, No. 1-2,3-191 (2006)
[2] J. L. Lions et al,C.R. Acad. Sci. Paris, Serie I, 332 (2001), pp.661668
[3] D. Samaddar et al, J. Comp Phys,18 (229) (2010)
[4] L. A. Berry et. al, Journal of Computational Physics 231(2012) 59455954
[5] W. R. Elwasif et al, 4th IEEE Workshop on Many-Task Computing on Grids and Supercomputers, MTAGS 2011, (2011)

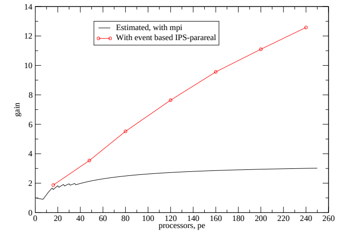

Fig. 5. The computational gain is significantly improved with the event based parareal, applied using the IPS framework as compared to a theoretical estimate of gain using traditional MPI.