

Parallel Plasma Fluid Turbulence Calculations

J. N. Leboeuf, B. A. Carreras, L. A. Charlton, J. B. Drake,
V. E. Lynch, D. E. Newman, K. L. Sidikman and D. A. Spong
Oak Ridge National Laboratory
Oak Ridge, TN 37831-8071

The study of plasma turbulence and transport is a complex problem of critical importance for fusion-relevant plasmas. To this day, the fluid treatment of plasma dynamics is the best approach to realistic physics at the high resolution required for certain experimentally relevant calculations. Core and edge turbulence in a magnetic fusion device have been modeled using state-of-the-art, nonlinear, three-dimensional, initial-value fluid and gyrofluid codes. Parallel implementation of these models on diverse platforms -- vector parallel (National Energy Research Supercomputer Center's CRAY Y-MP C90), massively parallel (Intel Paragon XP/S 35), and serial parallel (clusters of high performance workstations using PVM) -- offers a variety of paths to both high resolution and significant improvements in real-time efficiency, each with its own advantages. The largest and most efficient calculations have been performed at the 200 Mword memory limit on the C90 in dedicated mode, where an overlap of 12 to 13 out of a maximum of 16 processors has been achieved with a gyrofluid model of core fluctuations. The richness of the physics captured by these calculations is commensurate with the increased resolution and efficiency, and is limited only by the ingenuity brought to the analysis of the massive amounts of data generated.

1. Introduction

Indications are that turbulence is the dominant mechanism of charged particle loss in toroidal fusion devices such as tokamaks. In the fluid approach, the problem of plasma turbulence reduces to solving for the evolution in space and time of a set of moment equations for macroscopic variables (B, V, n, T, \dots). This makes tractable some problems that are at present too complex in kinetic theory¹. The regime of applicability of the fluid approach has recently been extended by inclusion of finite ion gyroradius and of kinetic effects, through closure relations compatible with Landau damping, into gyro-Landau fluid models^{2,3}. High resolution is needed to resolve physically relevant scale lengths in a plasma turbulence problem. This requires the use of large amounts of computer memory. To study the development of steady-state turbulence, the plasma evolution must be followed for many characteristic times. Therefore these calculations also require large amounts of computing time.

Parallel implementation of a fluid turbulence model applicable to the collisional plasma edge, as incorporated in the KITE code⁴, and of a gyro-Landau fluid model more suitable to the hotter plasma core, as incorporated in the DTEM code⁵, is being pursued on the CRAY Y-MP C90, on the Intel family of mainframes, and on clusters of workstations. This is done not only to improve the real time efficiency of these calculations, thereby decreasing the wallclock time for production calculations, but also to have access to more memory. These two elements permit inclusion of more realistic physics by evolving more plasma moments and/or using more complicated evolution equations for these moments. They also allow us to significantly increase problem sizes and resolutions and therefore perform more experimentally relevant calculations. Last, but not least, parallel implementation of our fluid turbulence models means that we will be ready for the next-generation of production supercomputers which are expected to be massively parallel. While different sets of equations are solved as an initial value problem in KITE and DTEM, both codes are spectral in the sense that a finite difference grid is used in the radial direction and a Fourier mode expansion is adopted in the poloidal (short way around the torus) and toroidal (long way around the torus) directions. Other features common to both codes include the implicit treatment of the linear terms, requiring a block-tridiagonal matrix inversion, and the explicit treatment of the nonlinear terms, involving the (analytical) calculation of mode convolutions. The matrix operations and mode convolutions are the most time-consuming parts of the calculations, and much effort has therefore been put into their efficient parallelization. Among these parallel implementations, the greatest physics gains have been realized on the C90 where core turbulence calculations using DTEM have been performed in dedicated multi-cpu mode and at the 200 Mword memory limit as part of the Special Parallel Processing (SPP) allocation program.

2. Dedicated Vector Parallel Calculations on the CRAY Y-MP C90

On the C90, microtasking compiler directives have been incorporated with relative ease in serial versions of the cylindrical geometry KITE and slab geometry DTEM codes, resulting in 98% parallelism. For the matrix solver and the convolutions in particular, the outer loop over the number of modes is now performed in parallel, while the inner loop over the number of radial grid points remains highly vectorized⁶. This parallel algorithm development, along with software and hardware advances at the National Energy Research Supercomputer Center (NERSC), have resulted in an improvement of performance by 2 orders of magnitude over one processor of a CRAY-2 computer. This is illustrated in Fig. 1, where the aggregate speed obtained with a one-equation version of DTEM is plotted as a function of year and mainframe. The highest aggregate speed and overlap have been achieved during short tests in dedicated mode as part of the SPP with a 200 Mword executable, namely 5 Gflops sustained with 14.5 processors accessed simultaneously.

We have reserved our SPP allocation for a specific physics application, namely gyro-Landau fluid calculations of turbulence driven by dissipative trapped electron modes

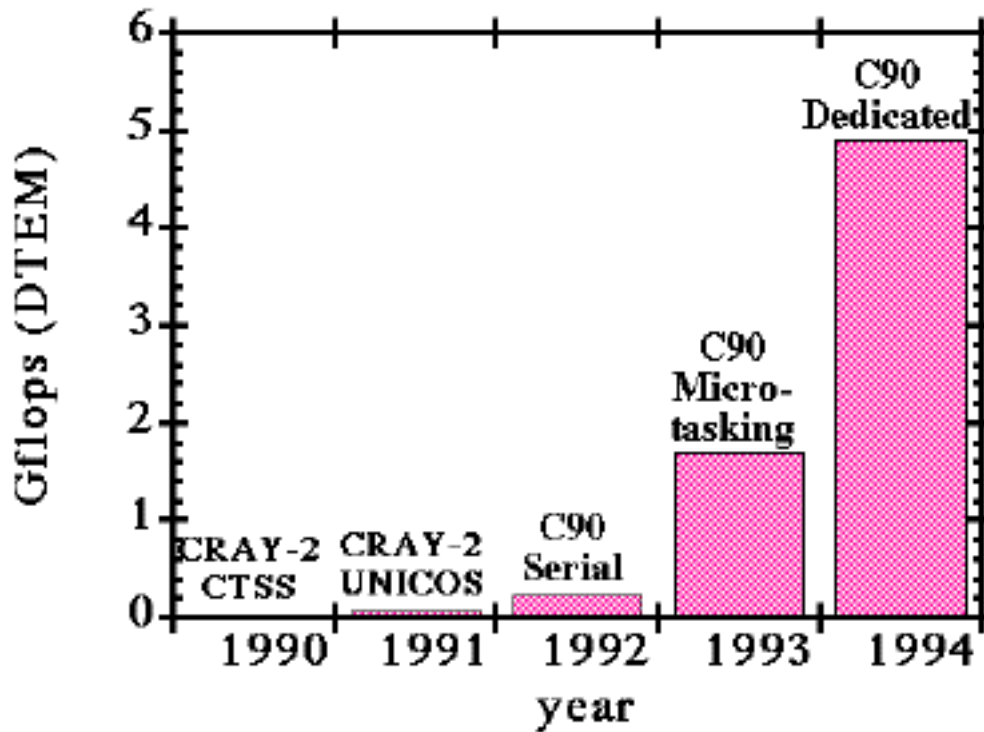


Fig.1 Time history of performance on CRAY computers

at the core of tokamak plasmas. The gyro-Landau fluid model of dissipative trapped electron modes consists of two time evolution equations for the plasma vorticity (Laplacian of the electrostatic potential) and the parallel component of the ion velocity. This two-equation model has been implemented in the DTEM code and includes kinetic phenomena such as finite ion Larmor radius contributions, ion Landau damping and ion magnetic drift effects through closure relations. The dissipative trapped electron instability drive is handled as a simple id phase shift between electron density and electrostatic potential fluctuations. Our SPP allocation has enabled us to perform two nonlinear, multiple helicity, three-dimensional calculations at the allowed 200 Mword limit in dedicated mode with parameters close to those for the core of the TEXT tokamak. These two calculations, herein referred to as Case 1 and Case 2, differ in the distribution of poloidal and toroidal mode pairs and the radial extent they cover, as shown in Fig. 2. Case 1 consists of 2333 modes and 600 radial grid points which

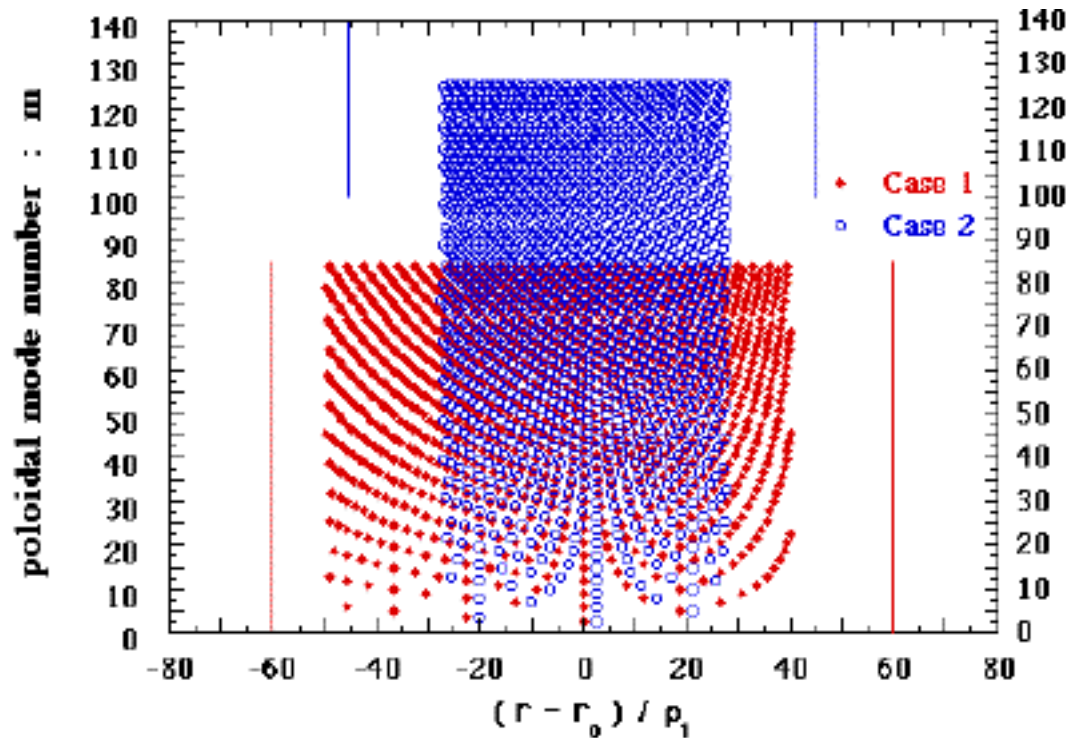


Fig. 2 Mode and grid point arrangement for SPP calculations

extend over 14 cm, or 50% of the TEXT minor radius. Case 2 consists of 3213 modes and 400 grid points covering 40% of the minor radius, with 5 grid points per ion Larmor radius for both. These are the best resolved and most CPU-time-intensive plasma turbulence calculations we have ever attempted. We should also emphasize that the turbulence dynamics we are tackling within the SPP is uncharted territory because of the physically realistic (i.e. small) dissipations used.

Before describing the physics results obtained, two operational issues need to be raised concerning the SPP. Overlaps between 12 and 13 have been routinely achieved for the two calculations performed in dedicated mode. This is somewhat lower than the overlap of 14.5 obtained in dedicated testing mode because production calculations such as the ones carried out here stay in the machine longer and have a greater likelihood of being interfered with by the interactive tasks that are allowed to execute during the dedicated time periods. A more serious deficiency is the inability to load a 200 Mword executable such as ours if interactive and system tasks take up more than the 68 Mword of memory left on the C90.

Nevertheless, the available dedicated SPP periods have enabled us to follow

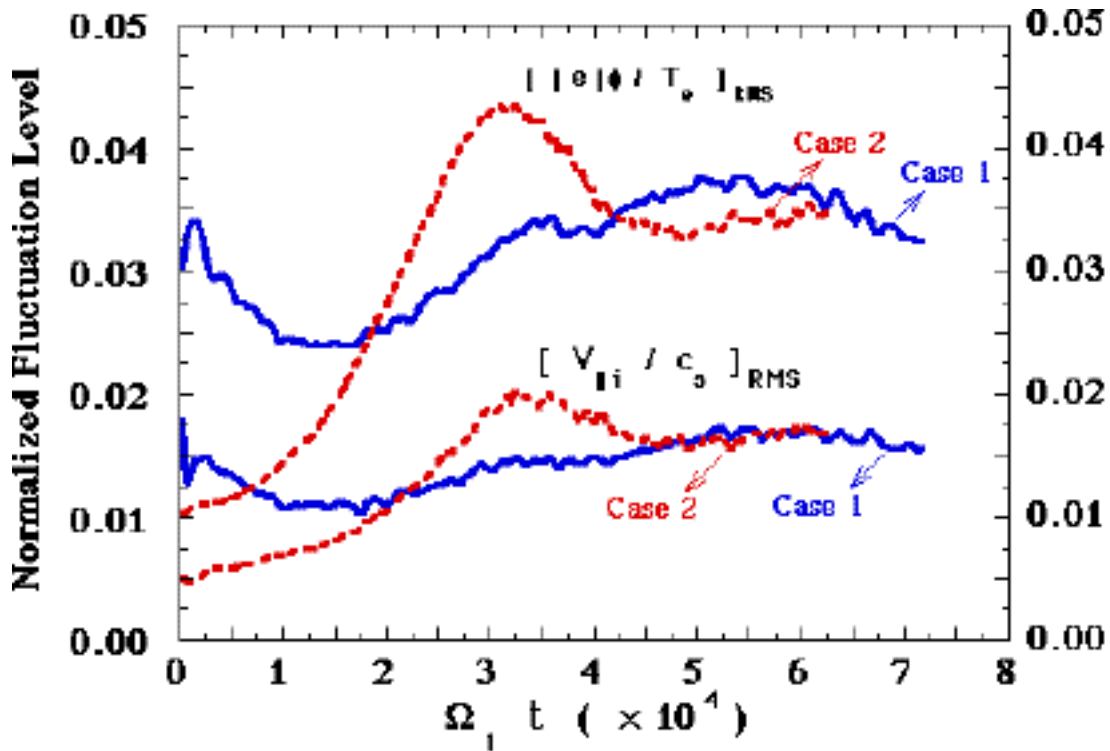


Fig. 3 Fluctuation levels vs time for SPP calculations

the time evolution of dissipative trapped electron mode instabilities to saturation for both Case 1 and Case 2. This is apparent from the time traces of the potential and parallel velocity fluctuation amplitudes displayed in Fig. 3. These time traces cover 15,000 time steps and 10,000 time steps for Case 1 and Case 2, respectively. Even though the calculations were started with different initial amplitudes (higher for Case 1 than for Case 2) and have different resolutions, the saturated fluctuation levels turn out to be comparable in magnitude. Similarly, the poloidal wavenumber spectra, averaged in time over the

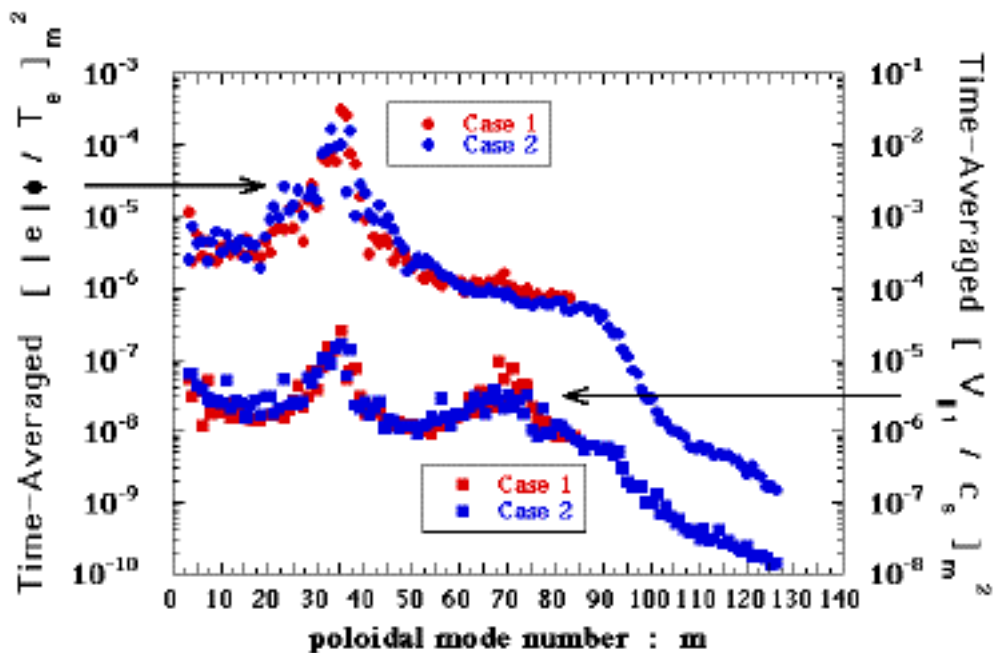


Fig. 4 Poloidal mode number spectra for SPP calculations

saturation phase, displayed in Fig. 4, show a high degree of overlap. The spatial structure of the turbulence is vividly depicted in Fig. 5, where color contours of plasma vorticity

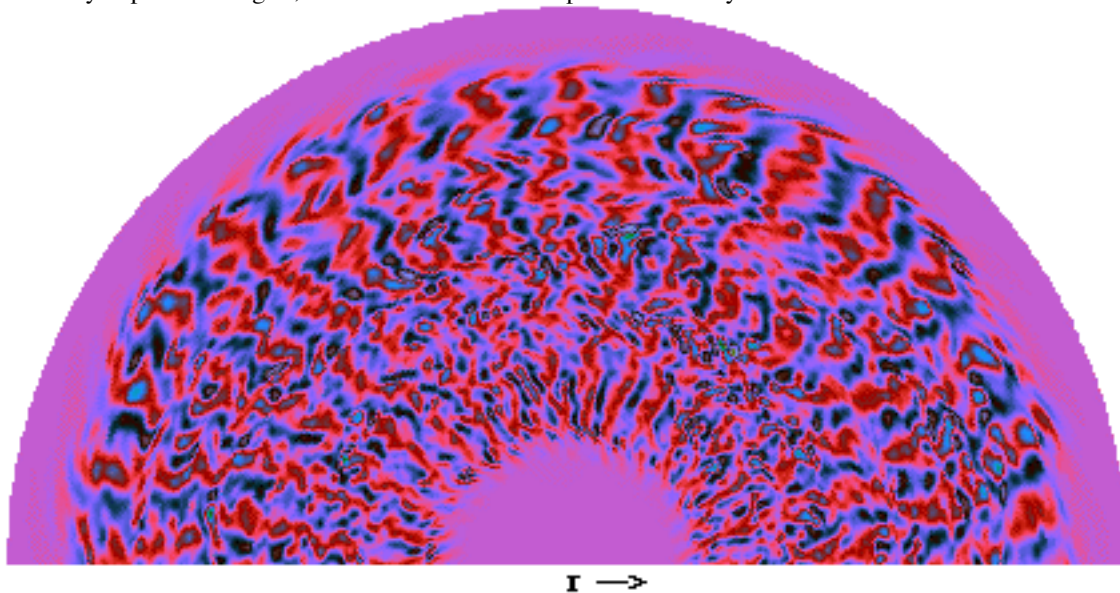


Fig. 5 Contours of vorticity for Case 1 SPP calculation

are displayed as a function of radius and poloidal angle for Case 1. Large and small scale structures are seen to coexist, emphasizing anew the need for highly resolved calculations in both space and time. These core fluctuation studies are far from complete, and additional SPP resources are needed to resolve several outstanding issues. In particular, the calculations will need to be continued further in time to assess whether a turbulent steady state has been achieved. This will also allow us to obtain better converged frequency and wavenumber spectra in steady state which are more directly compared to experimental observations.

3. Massively Parallel Calculations on Intel Mainframes

The C90 is probably the last vector parallel supercomputer to be housed at NERSC. To be ready for the next generation of production supercomputers, we have been porting our workhorse edge turbulence code KITE to a vast array of massively parallel platforms. These include the Intel iPSC/860's at the Oak Ridge National Laboratory (ORNL) and the California Institute of Technology (Cal Tech), the Intel Delta at Cal Tech and Intel Paragons XP/S 5 and XP/S 35 at ORNL, as well as the BBN Butterfly at the Lawrence Livermore National Laboratory. The message passing version of KITE, appropriate for these platforms, did require development of parallel algorithms for the two most CPU-intensive parts of the algorithm, the block-tridiagonal matrix solver and the mode convolutions⁷. For the convolutions, a radial region is allocated in each node. In this way, the convolutions in each radial region are performed in parallel, and communication between processors is limited to the boundary values for each radial region. The memory configuration used in this implementation is a ring. For the block-tridiagonal matrix inversion, a second parallelization in which the poloidal and toroidal mode pairs are distributed in processors is required.

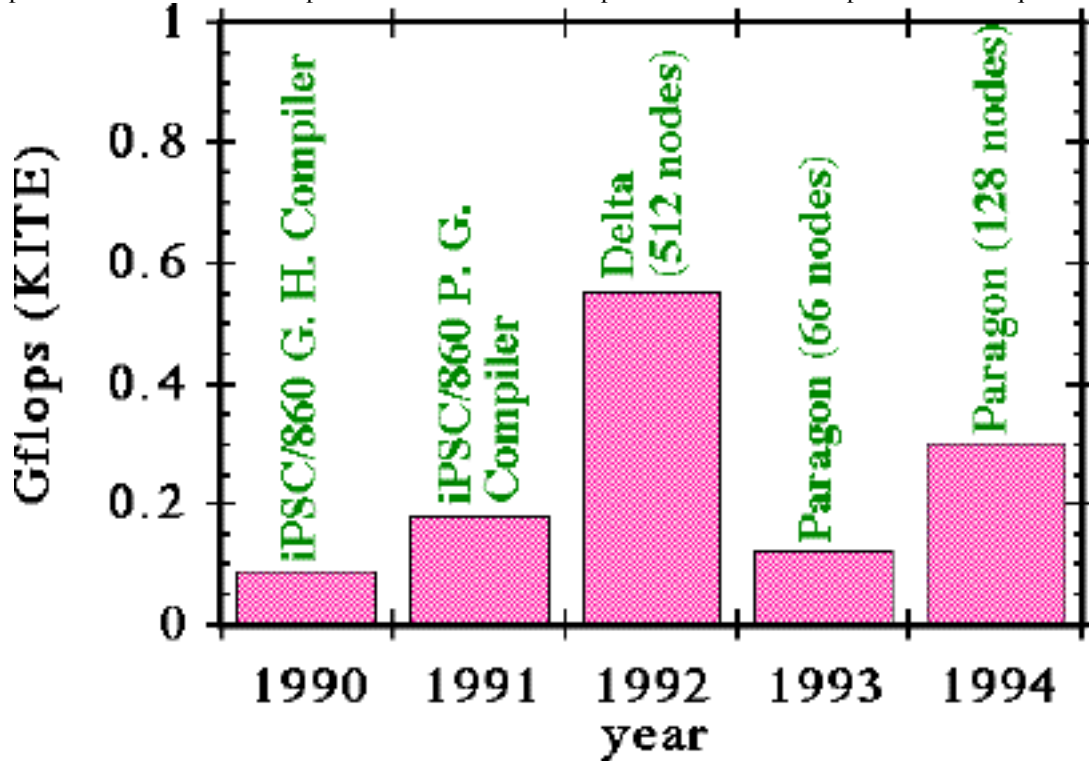


Fig. 6 Time history of performance on the Intels

A compendium of benchmarks with KITE is given in Fig. 6, where the best aggregate speed obtained is displayed as a function of year and mainframe. Significant improvements of performance were realized with the Intel Delta over the iPSC/860. The Intel Delta performance is being approached with 128 processors of the 512 node Intel XP/S 35 and we should be able to exceed it if and when our code can run on more processors without causing machine failures. The many tests performed with KITE on the Intels indicate that the performance of the parallel algorithms clearly improves with the size of the problem in terms of number of grid points and number of modes and that there is an optimal number of processors for each problem size, whereby processors are kept busy computing instead of communicating. The memory per processor must therefore increase to achieve optimal performance as the problem size and number of processors increase⁷. The memory per node on present and future Paragons is set at 32 Mbyte. To make optimal use of the 2048 processors on the soon to be available Paragon XP/S 150, our tests indicate that it is necessary to modify the storage system of the present code.

4. Serial Parallel Calculations on Workstation Clusters

Given the limitations imposed on problem size by the available memory per node on the Intel Paragons, we are investigating the possibility of using workstation clusters as a path to higher resolution. To do so, the message passing version of KITE developed for the Intels has been modified to run under PVM. The modifications consisted mainly in replacing the existing message passing library calls with PVM ones. This

is enabling us to explore the possibility of using heterogeneous clusters of workstations and/or processors of supercomputers for these calculations, while at the same time maximizing the portability of our edge turbulence model. Benchmarks have been carried out using KITE with a resolution of 512 radial grid points, 455 mode pairs and an increasing number of workstations. Using Ethernet connections between IBM RS6000 and DEC Alpha workstations, a factor of 2 speedup can be achieved by using four workstations and

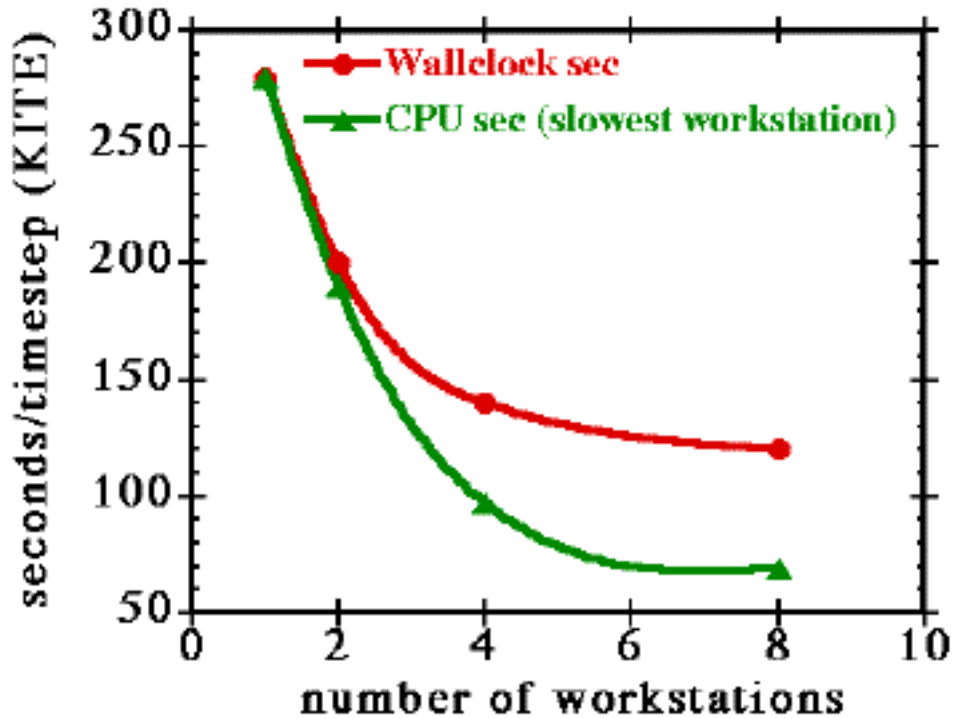


Fig. 7 PVM benchmarks with KITE

more resolution can be used than will fit in the memory of one workstation. With the present workstation connections, eight workstations further increase the memory available, but the calculations take almost as much time as using four, as shown in Fig. 7. It is expected that forthcoming optical connections will alleviate the communications bottleneck and allow efficient use of more workstations under PVM. We are currently exploring the possibility of distributing these calculations to the C90 and CRAY-2 processors at NERSC. Tests indicate that the calculations are much slower on many CRAY computers using PVM than with microtasking on the C90 alone, but we hope to take advantage of the abundant memory available on more than one CRAY.

5. Summary

To date, our best resolved and most efficient plasma fluid turbulence calculations have been performed in dedicated mode on the C90 within NERSC's SPP allocation program. It is hoped that the SPP program will be continued and expanded so that we can capitalize on the tantalizing physics results already obtained. True dedicated mode without interference from other tasks is a must if we are to take advantage of all 16 processors. On the other hand, the full potential of our massively parallel codes has not yet been realized on the existing Intel Paragons. Further algorithm development remains to be carried out to minimize the memory per node used and thus be able to make optimal use of all 2048 processors of the soon to be available Paragon XP/S 150.

Acknowledgments

We would like to thank Bruce Curtis and Roy Troutman from NERSC for their invaluable help in optimizing the vector parallel versions of our plasma fluid turbulence codes and in scientific visualization of the data thus produced. A portion of this work has been carried out as part of the Numerical Tokamak Grand Challenge Project. This research was sponsored by the Office of Fusion Energy and the Office of Scientific Computing, U. S. Department of Energy, under contract DE-AC05-84OR21400 with Martin Marietta Energy Systems, Inc.

References

1. B. A. Carreras, N. Dominguez, J. B. Drake, J. N. Leboeuf, L. A. Charlton, J. A. Holmes, D. K. Lee, V. E. Lynch, and L. Garcia, *Int. J. Supercomput. Appl.* **4**, 97 (1990).
2. G. W. Hammet and F. W. Perkins, *Phys. Rev. Lett.* **64**, 2019 (1990).
3. C. L. Hedrick and J. N. Leboeuf, *Phys. Fluids B* **4**, 3915 (1992).
4. L. Garcia, H. R. Hicks, B. A. Carreras, L. A. Charlton, and J. A. Holmes, *J. Comput. Phys.* **65**, 253(1986).
5. K. L. Sidikman, B. A. Carreras, and L. Garcia, *J. Comput. Phys.* **114**, 100(1994).
6. V. E. Lynch, B. A. Carreras, J. N. Leboeuf, B. C. Curtis and R. L. Troutman, "Multi-CPU plasma fluid turbulence calculations on a CRAY Y-MP C90", *Proceedings of Supercomputing '93*, Portland, Oregon, November 15-19, 1993, p. 308 (Institute of Electrical and Electronics Engineers Computer Society Press, Los Alamitos, Ca., 1993).
7. V. E. Lynch, B. A. Carreras, J. B. Drake, J. N. Leboeuf and P. C. Liewer, "Performance of a plasma fluid code on the Intel parallel computers", *Proceedings of Supercomputing '92*, Minneapolis, Minnesota, November 16-22, 1992, p. 286 (Institute of Electrical and Electronics Engineers Computer Society Press, Los Alamitos, Ca., 1992).