Implementation and Characterization of a Hybrid Electromagnetic PIC Code on the HP spp2000

Joseph Wang, Edith Huang, and Paulett Liewer
Jet Propulsion Laboratory

The physics underlying the global solar wind/magnetospheric plasma system is controlled not only by the global dynamics of plasma flow but also by a large variety of small-scale, microscopic processes (e.g. plasma microinstabilities driven by non-thermal ion velocity distributions) which are manifest in collisionless space plasmas. Current models of global space plasma phenomena are mostly based on the equations of isotropic magnetohydrodynamics (MHD). However, such large-scale MHD models have serious limitations for those problems involving characteristic length scales comparable to the ion gyroradius and non-thermal ion velocity distributions and do not include microscopic plasma physics processes. Plasma microinstabilities, which can affect the larger scale couplings, can only be resolved within the framework of plasma kinetic theory. While computer particle simulations provide the best theoretical means to address the microscopic processes in space plasmas, their applications have been mostly limited to small scale simulations by computational power.

Recent advances in parallel computing technology allow for the first time some kinetic plasma simulations be performed to resolve both the microscopic processes and the global dynamics of large-scale space plasma systems for realistic parameters. Research is currently being conducted at JPL to perform one of the first such "mesoscale" simulations of the outer magnetosphere. We have recently ported a sequential two-dimensional electromagnetic hybrid particle-in-cell code to the new HP spp2000 machine at Caltech and the Cray T3D at JPL. In an electromagnetic hybrid PIC code[1], a plasma is modeled by many test particles representing the ions and a massless fluid representing the electrons, and the Maxwell's equations, the electron fluid equations, and the trajectories of individual ion particles are solved self-consistently. The sequential hybrid code has been applied in a wide range of simulation studies in space plasma physics (see [2] and references there in).

The code is implemented using the General Concurrent PIC (GCPIC) algorithm[3]. The algorithm uses spatial decomposition of the physical domain to divide the computation among parallel processors. Each processor is assigned a subdomain and all the particles and grid points in it. To minimize interprocessor communicant ion, each processor also stores "guard cells" to ensure that all part icle-grid interpolations can be performed locally. When a particle moves from one subdomain to another, it must be passed to the appropriate processors, which requires interprocessor communication. Interprocessor communication is also necessary to exchange guard cell information. Interprocessor communication subroutines are based on the MPI library.

Fig.1 shows typical performances of this parallel hybrid PIC code for a relatively homogeneous plasma on the HP spp2000. Fig.1a shows the timing results for scaled problem size runs, where the problem size on each processor is fixed and the total problem size is proportional to the number of processors used. Two problem sizes are used: in S 1, the problem size per PE is 84 x 84 grid cells and about 1.02 million particles (144 particles/cell); in S2, the problem size per PE is 128 x 128 grid cells and about 2.7 million particles (144 particles/cell). When the problem size is scaled to 128 processors, the total problem size is about 0.9 million grids and 130 million particles for S1 and 2.1 million grids and 302 million particles for S2 respectively. (The memory size required to run these two problems is 30 Mbytes/PE for S1 and 70 Mbytes/PE for S2 respectively.) The CPU time per time step is plotted as a function of the grid cell numbers. We find that the parallel hybrid code runs at a high parallel efficiency of ≥90% for both problems. For comparison, we also plot the timing results of running the parallel hybrid code on the Cray T3D (for S1) using up to 128 PE and running the sequential hybrid code on a single CPU of Cray J90. The millions of floating point operations per second per CPU (Mflops/CPU) achieved for these runs are 77 Mflops/CPU for the HP spp2000 (peak speed: 720 Mflops/CPU) and 19.5 Mflops/CPU for the Cray T3D (peak speed: 150 Mflops/CPU) respectively. Using the CPU time divided by the maximum number of particles (or grid cells) used in these runs as a measure of the performance, we find the results displayed in Fig.1a represent a speedup of about 340 on a 128 PE HP spp2000 over a 1 CPU Cray J90. In Fig.1b we show the timing results for running a fixed size problem on the HP spp2000 as a function of the number of processors used. The total problem size in Fig.1b is 256 x 256 grid cells and about 9.44 million particles. Not surprisingly, for fixed size problem runs the parallel efficiency decreases as the number of processors increases due to decreasing workload per processor. In order to achieve a > 90% parallel efficiency for our parallel hybrid code on the HP spp2000, we find that each processor needs to have a minimum workload larger than 64 x 64 grid cells and 0.5 million particles.

Acknowledgement:

We thank our collaborators, S. Peter Gary and Dan Winske of Los Alamos National Lab, for their contributions. This research is supported by the NASA Space Physics SR&T Program. Access to the HP spp2000 at Caltech and the Cray T3D at JPL was provided by funding from NASA Offices of Aeronautics, Mission to Planet Earth, and Space Science.

Reference:

1. D. Winske and N. Omidi, Hybrid codes, methods and applications, in Computer Space Plasma Physics: Simulation Techniques and Software, edited by H. Matsumoto and Y. Omura, Tokyo, 1993.

2. S.P. Gary , J. Wang, D. Winske, and S. A. Fuselier, Proton temperature anisotropy upper bound, J. Geophys. Res., (in press, 1997).

3. P.C. Liewer, and V.K. Decyk, A general concurrent algorithm for plasma particle-in-cell simulation codes, J. Comput. Phys, 85, pp302, 1989.

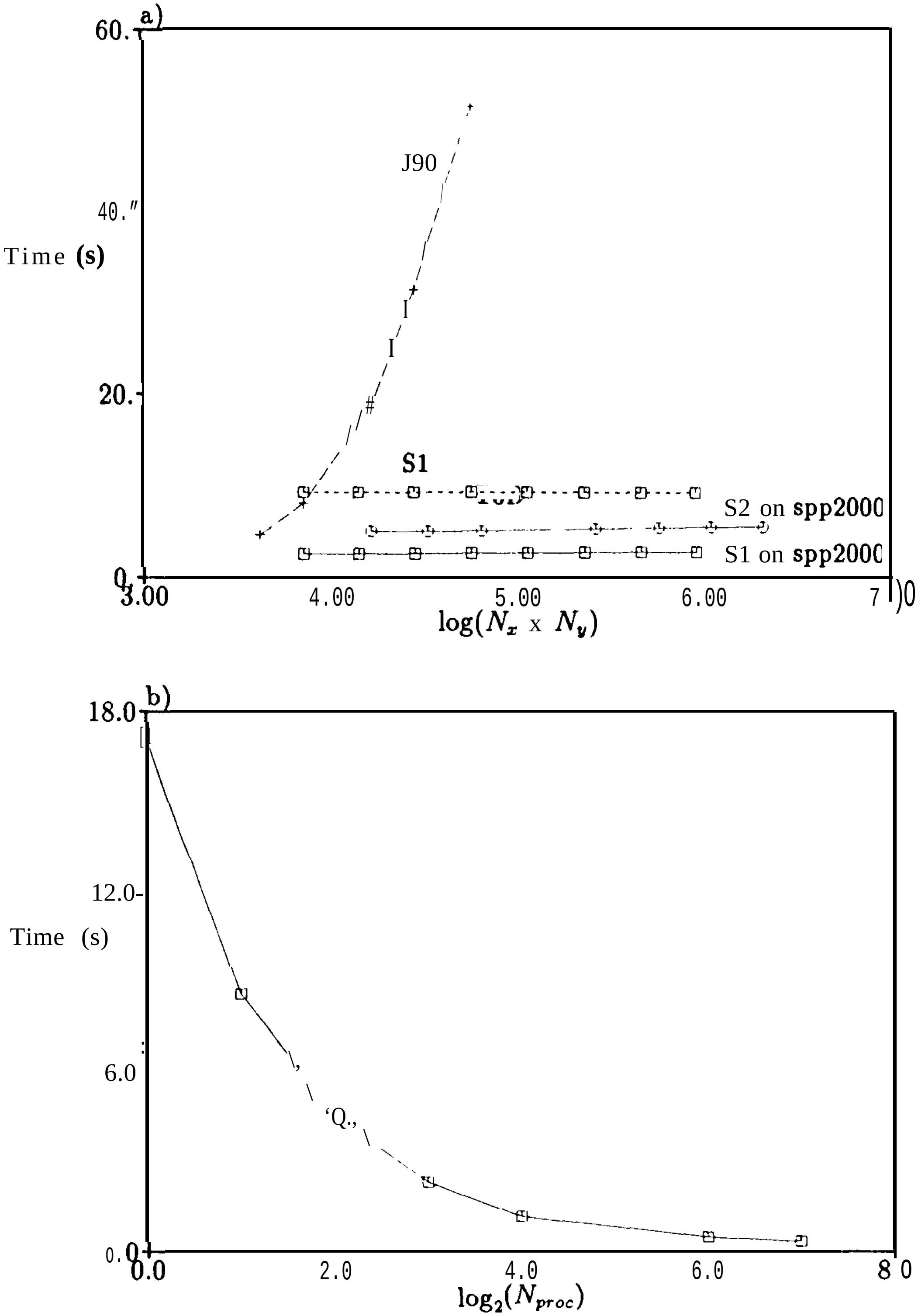

Figure 1: CPUtime perstep ofaparallel hybrid electromagnetic PIC code. a)Time/steps.sa function ofgrid points $(N_x \times N_y)$ for scaled up problem runs on the HP spp2000 (solid lines) and the Cray T3D (dotted line) using 1, 2, 4, 8, 16, 32, 64, and 128 processors. For comparison, the run times for the sequential hybrid code on the Cray J90 (dashed line) is also plotted. b) Time/step as a function of the number of processors used (N_{proc}) for fixed problem size runs on the HP spp2000 using 1, 2, 4, 8, 16, 64, and 128 processors..