CHAPTER 2

EXTENSION AND VALIDATION OF A HYBRID PARTICLE-FINITE ELEMENT METHOD FOR HYPERVELOCITY IMPACT SIMULATION

Eric P. Fahrenthold[1] **and Ravishankar Shivarama**[2]

Department of Mechanical Engineering, 1 University Station C2200
University of Texas, Austin, TX 78712, USA

[1]Professor, corresponding author, phone: (512) 471-3064, email: epfahren@mail.utexas.edu
[2]Graduate student

Extension and Validation of a Hybrid Particle-Finite Element Method for Hypervelocity Impact Simulation

Eric P. Fahrenthold and Ravishankar Shivarama

Department of Mechanical Engineering, 1 University Station C2200, University of Texas, Austin, TX 78712, USA

Abstract

The hybrid particle-finite element method of Fahrenthold and Horban [1], developed for the simulation of hypervelocity impact problems, has been extended to include new formulations of the particle-element kinematics, additional constitutive models, and an improved numerical implementation. The extended formulation has been validated in three dimensional simulations of published impact experiments. The test cases demonstrate good agreement with experiment, good parallel speedup, and numerical convergence of the simulation results.

Keywords: impact simulation, numerical methods

1. Introduction

In previous work Fahrenthold and Horban [1] developed a hybrid particle-finite element method for hypervelocity impact simulation, and applied that formulation in the analysis of several three dimensional problems. The referenced modeling methodolody employs Lagrangian finite elements to represent material strength effects, namely tension and elastic-plastic shear, and Lagrangian particles to represent inertia, compressed states, and contact-impact effects. Particles and elements are used in tandem throughout the simulation, for all materials, so that no element-to-particle transformations are required. The introduction of both particles and elements is not redundant, since they are employed to account for distinct physical effects. A systematic approach to the formulation of this hybrid numerical scheme is provided by Hamiltonian mechanics. General thermomechanical dynamics are represented,

via the introduction of entropy variables as generalized Hamiltonian coordinates.

Development of the particle-element method just outlined has been motivated by difficulties encountered in the application of pure Eulerian, Lagrangian, or particle based methods to the problem of orbital debris shielding design [2]. The latter application calls for simulation of shock loading and perforation at very high velocities, accurate characterization of strength-dependent structural response, efficient modeling of fragment transport, and general descriptions of contact-impact. Recent research suggests that hybrid [3] or coupled [4] numerical methods are best suited to simulate orbital debris impact on space structures. Numerical methods developed for the latter application may also have application in related problems, such as research on the effects of behind armor debris.

2. Extended Formulation

The hybrid particle-finite element model of Fahrenthold and Horban [1] has been extended to include alternative formulations of the particle-element kinematics, additional constitutive models, and an improved numerical implementation. The extended formulation described in this section has been evaluated for accuracy, numerical convergence, and parallel speedup in a series of three dimensional simulations of published experiments, as described in the sections which follow.

The density interpolation of reference [1] used distinct kernels for reference configuration nearest neighbors and for all other particles. A simplified alternative is the implicit interpolation

$$\rho^{(i)} = {\rho_o}^{(i)} + \frac{1}{8} \sum_{j=1}^{n-1} {\rho_o}^{(j)} \left(\left[\frac{2\alpha h^{(j)}}{r_{ij}} \right]^3 - 1 \right) \Lambda \left(\frac{2\beta h^{(j)}}{r_{ij}} \left[\frac{{\rho_o}^{(i)}}{\rho^{(i)}} \right]^{\frac{1}{3}} - 1 \right) \qquad (1)$$

where $\rho^{(i)}$ is the interpolated density, ${\rho_o}^{(i)}$ is a reference density, $h^{(j)}$ is a particle radius, r_{ij} is a particle center of mass separation distance, the constants α and β are determined by the body centered cubic particle packing scheme, n is the number of particles, Λ denotes the unit step function, and the summation applies for $j \neq i$. Note that the preceding interpolation yields exact results for hydrostatic compression of a particle set arranged in the selected packing scheme, and that in general all neighbor particles interact in the same fashion.

The particle packing scheme used in the present work leads to eight nearest neighbors for body centered particles in the reference configuration. Fahrenthold and Horban [1] used the body centered particle associated with each hexahedral element to define subelement domains. The volumes of these subdomains may be used to determine the interparticle tensile forces associated with material dilatation. This six point integration scheme is computationally expensive, and leads to hexahedral elements which are stiffer than those employed for example in some very successful Lagrangian codes [5]. In the present work the potential energy contribution due to tension is written

$$U^{ten} = \sum_{j=1}^{n_e} \frac{1}{2} (1 - D^{(j)}) \{ V^{(j)} \kappa^{(j)} (J^{(j)} - 1)^2 \Lambda(J^{(j)} - 1) + 2h^{(j)} E^{(j)} \mid \mathbf{c}^{(j)} - \mathbf{c}^{avg(j)} \mid^2 \} \qquad (2)$$

where n_e is the number of elements, $V^{(j)}$ is the element reference volume, $\kappa^{(j)}$ is the element bulk modulus, $E^{(j)}$ is the element Young's modulus, $J^{(j)}$ is the element Jacobian, $D^{(j)}$ is the element normal damage, $\mathbf{c}^{(j)}$ is the position vector of the body centered particle, and $\mathbf{c}^{avg(j)}$ is the average position vector of the particles which define the element nodes. This potential function represents a one point integration of the element, to quantify tension effects, along with a penalty term which positions the body centered particle. Note that interparticle tension is (neglecting surface tension) a strength effect, and is therefore associated with relative motion of a reference configuration neighbor set (in this case the finite elements) and not a time varying neighbor set of the type used to quantify collision forces [6].

Shock physics codes can include options for the treatment of energy conservation errors which may occur during integration or transformation of the system level model. Examples are a default energy discard used in the remap step of CTH [7] and energy errors which may arise from contact-impact calculations in ALE [8] or SPH [9,10] formulations. Although precise energy conservation can be demonstrated in simple benchmark problems, it can be computationally expensive to achieve the same result in the simulation of more complex practical problems. For the hybrid particle-element formulation described here, an optional energy correction term has been added to the particle entropy evolution equations

$$\Delta S^{err} = \epsilon \, \Delta H^{err} \left(\sum_{j=1}^{n} \theta^{(j)} \right)^{-1} \tag{3}$$

where ΔH^{err} is the error in the system Hamiltonian, $\theta^{(j)}$ is a particle temperature, and with $\epsilon = 0.1$ the error correction is introduced over ten time steps. The effect of this term is to satisfy global energy conservation by introducing an internal energy correction for each particle which is proportional to the current particle temperature.

Simulation results obtained using particle based models are in general influenced by the choice of particle packing scheme. The locations and properties of nearest neighbor particles define in general a local stiffness distribution for the medium which is anisotropic, as in the case of pure crystals [11]. In addition the choice of a particle packing scheme determines an effective void ratio for the medium. Hence it is not surprising that particle packing can influence simulation results. The preprocessor used in the present work employs a random number generator to delete a user specified fraction of body centered particles from the original perfect lattice structure. The introduction of these flaws mimics the influence of grain orientations, dislocations, and other defects on the mechanical response of real materials.

The simulations described in references [1] and [3] employed analytic equations of state. Extension of the code described in the latter work has included the introduction of interpolation routines, to accommodate tabulated equations of state in two independent variables. Currently the SESAME tables [12] are used for simulations at very high velocities, and are accessed in their default form, with pressure and internal energy defined on a density and temperature grid. Iteration is therefore used to converge on an internal energy calculated from the Gibbs relation

$$\dot{U}^{int(i)} = \theta^{(i)} \, \dot{S}^{(i)} + \frac{m^{(i)} P^{(i)}}{{\rho^{(i)}}^2} \left(\sum_{j=1}^{n} \frac{\partial \rho^{(i)}}{\partial \mathbf{c}^{(j)}} \, \dot{\mathbf{c}}^{(j)} \right) \tag{4}$$

where $m^{(i)}$ is a particle mass and $P^{(i)}$ is a particle pressure. An initial call to the SESAME routines is used to establish the reference internal energy associated with the simulation initial conditions.

In addition to the preceding work, significant extensions and applications of the hybrid particle-element method discussed here are reported elsewhere. First, Shivarama [13] has extended the basic formulation to include ellipsoidal particles, introducing a nonspherical kernel, rotational motion of the particles, and Euler parameters as state variables. This extension makes possible for example the modeling of thin plate structures at greatly reduced particle counts. It should be noted that previous work on nonspherical particle models for shock physics problems has been very limited [14,15]. Second, application of the method to composite orbital debris shielding problems is reported by Fahrenthold and Park [16], including development and numerical implementation of a rate-dependent material model for Kevlar.

3. Validation Simulations

A series of three dimensional simulations was performed to evaluate the extended formulation just described. The simulations modeled published experiments conducted at impact velocities ranging from one to eleven kilometers per second. Each example problem was modeled at two different mesh densities, to investigate numerical convergence of the simulation results. The selected problems have been previously studied by various investigators, to evaluate other numerical methods and computer codes. Tables 1 and 2 provide details on the problem parameters and material properties for all the validation simulations. Material properties were estimated using data from Steinberg [23]. The first two example problems used a Mie-Gruneisen equation of state, while the third example problem used the SESAME tables.

Table 1. Parameters of the example problems

Parameter	*Sphere*	*Long rod*	*Debris shield*
Projectile material	Aluminum	DU 0.75% Ti	Aluminum
Target/shield/wall material	Aluminum	Steel	Aluminum
First shield thickness (cm)	na	na	0.25
Second shield thickness (cm)	na	na	0.25
Target/wall plate thickness (cm)	0.1143	0.64	0.50
Shield spacing (cm)	na	na	6.0
Projectile velocity (km/sec)	6.56	1.21	11.0
Impact obliquity (deg)	45	73.5	45
Projectile diameter (cm)	0.953	0.767	0.5062
Projectile length-to-diameter ratio	na	10	4.36

na = not applicable

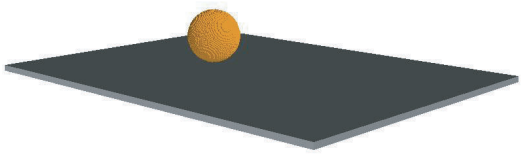

Fig. 1. Element plot of the initial configuration for the sphere impact problem.

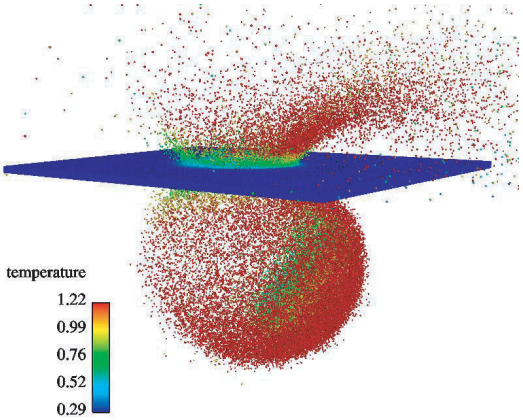

Fig. 2. Particle plot of the simulation results for the sphere impact problem.

Table 2. Material properties used in the example problems

Material property	*Aluminum*	*DU 0.75% Ti*	*Steel*
Shear modulus (Mbar)	0.271	0.74	0.801
Reference density (g/cc)	2.7	18.62	7.842
Initial yield stress (Mbar)	0.0029	0.0095	0.012
Maximum yield stress (Mbar)	0.0058	0.0220	0.025
Strain hardening exponent	0.1	0.095	0.5
Strain hardening modulus	125.0	1000.0	2.0
Melt temperature (degrees Kelvin)	1,220	1,710	2,310
Specific heat (Mbar-cm^3 per g-kilodeg Kelvin)	0.00884	0.00111	0.00448
Spall stress (Mbar)	0.012	0.028	0.032
Plastic failure strain	1.0	1.0	1.0

The first example problem involves the oblique (45 degree) impact of an 0.953 cm diameter aluminum sphere on a thin (0.1143 cm) plate, at 6.56 kilometers per second, and is representative of typical Whipple shield design problems. Figure 1 shows an element plot of the initial configuration of the projectile and target, while Figure 2 shows a particle plot of the simulation results at 6.6 microseconds after impact. The simulation results show good agreement with the experimental radiograph [17]. This problem was run on 16 processors of an IBM Regatta at two different mesh densities, with models composed of 0.67 million and 3.20 million particles. The dimensions of the plate perforation were compared to determine the effect of model resolution on the simulation results. Table 3 shows that an increase in the particle count by nearly a factor of five produced only a small variation in the predicted dimensions of the plate perforation.

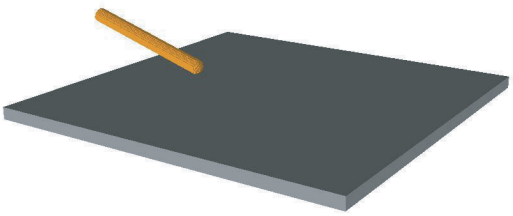

Fig. 3. Element plot of the initial configuration for the long rod impact problem.

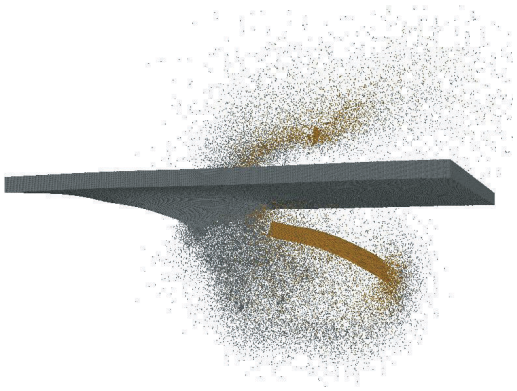

Fig. 4. Particle plot of the simulation results for the long rod impact problem.

Table 3. Simulation results for the sphere impact problem

Problem size (particles)	*Wall clock time (hours)*	*Perforation width (cm)*	*Perforation length (cm)*
0.673 million	4.04	1.90	2.43
3.196 million	39.6	1.81	2.39

The second example problem involves the oblique (73.5 degree) impact of an 0.767 cm diameter uranium alloy rod (L/D = 10) on a steel plate target (thickness 0.64 cm), at a rod velocity of 1.21 kilometers per second (target velocity was 0.217 km/s). This problem is representative of typical armor-antiarmor design applications. Figure 3 shows an element plot of the initial configuration of the projectile and target, while Figure 4 shows a particle plot of the simulation results at 100 microseconds after impact. The simulation results provided in Table 4 show good agreement with the corresponding experimental values [18] of residual rod length (5.55 cm) and residual rod velocity (1.07 km/sec). This problem was run on 16 processors of an IBM Regatta at two different mesh densities, with models composed of 1.56 million and 5.06 million particles. Table 4 shows that a factor of more than three increase in the particle count produced only a small variation in the simulation results.

Table 4. Simulation results for the long rod impact problem

Problem size (particles)	*Wall clock time (hours)*	*Residual length (cm)*	*Residual velocity (km/s)*
1.566 million	13.8	5.74	1.09
5.060 million	74.4	5.78	1.09

The third example problem involves the oblique (45 degree) impact of an aluminum shaped charge projectile (0.5062 cm diameter, 2.2046 cm length) on an aluminum plate target protected by a dual plate aluminum debris shield. The projectile velocity was 11.0 kilometers per second, while the wall plate thickness (0.50 cm), shield thickness (0.25 cm), and total

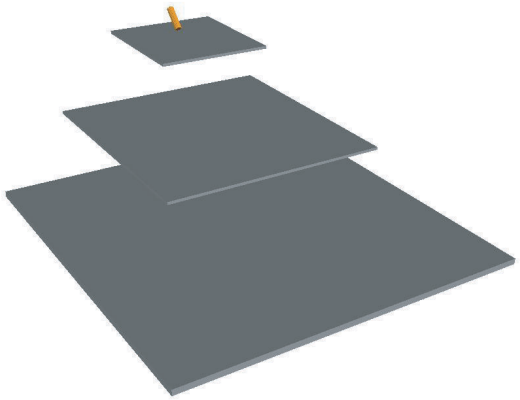

Fig. 5. Element plot of the initial configuration for the dual plate shield problem.

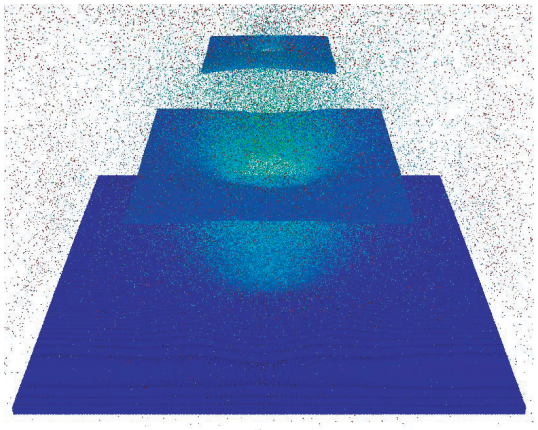

Fig. 6. Particle plot of the simulation results for the dual plate shield problem.

standoff distance (12.0 cm) are representative of orbital debris shielding design applications. This problem has been designated a benchmark for use in the numerical analysis of spacecraft protection systems [19]. Figure 5 shows an element plot of the initial configuration of the system, while Figure 6 shows a particle plot of the simulation results at 150 microseconds after impact. Consistent with the corresponding experiment, the simulation results show bulging but no perforation of the wall plate. This problem was run on SGI Origin systems at two different mesh densities, with models composed of 4.27 million and 12.90 million particles. The smaller model required 56.8 wall clock hours on 256 (400MHz) processors, while the larger model required 332 wall clock hours on an average of 502 (600Mhz) processors. The two simulations indicated very similar wall plate damage. Figures 7 through 9 show details of the shield perforations and wall plate damage, in element plots made at 150 microseconds after impact. This problem illustrates that the hybrid numerical method used here is well suited to represent both the very general contact-impact dynamics illustrated in Figure 6 and the large deformation plasticity illustrated in Figures 7 through 9.

4. Parallel Speedup Tests

The simulations described in the preceding section illustrate that three dimensional hypervelocity impact simulation can be very computer resource intensive. Hence the parallel performance characteristics of the relevant algorithm and numerical implementation is of considerable interest. This section discusses several algorithmic and implementation issues and presents measured parallel speedup data. Parallel implementation [20] of the hybrid method discussed here is based on the OpenMP standard [21], a set of portable compiler directives which may be simply applied to achieve loop level parallelism. Although an OpenMP implementation does not manage distributed memory, the effects of nonuniform memory access on code performance can in some cases be observed and influenced, through changes in the initial data placement scheme.

The hybrid nature of the present numerical formulation requires that both finite element

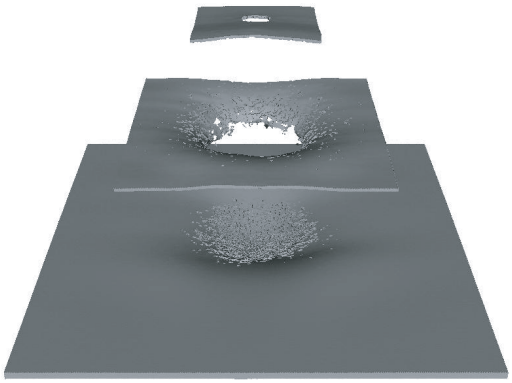

Fig. 7. Element plot of the simulation results for the dual plate shield problem.

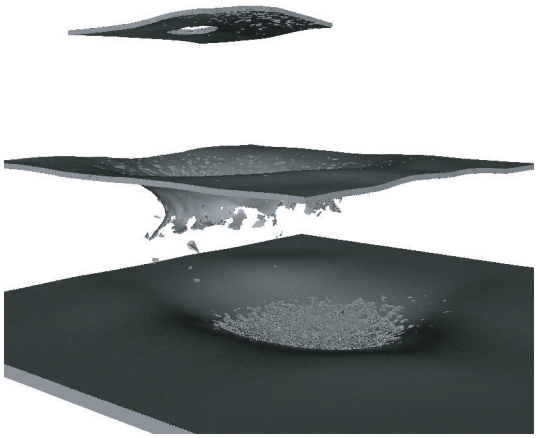

Fig. 8. Oblique view of the simulation results for the dual plate shield problem.

based and particle based computation take place throughout the simulation. The Lagrangian finite element calculations involve an invariant neighbor set for each particle, known at the start of the calculation. This allows for a very efficient parallel implementation, and suggests the use of a material based domain decomposition technique on distributed memory systems. Unfortunately a Lagrangian description of contact-impact and fragmentation effects, applied here and represented using the particle kinematics, involves a second (and time varying) neighbor set for each particle. Calculations involving this neighbor set are relatively inefficient, since a significant number of nominal neighbors identified by a search algorithm will turn out to be outside the effective mechanical or thermal interaction range. This portion of the calculation suggests the use of a geometry based domain decomposition technique, like that used in Eulerian codes, for distributed memory systems. However in this case, unlike Eulerian models, the membership of the contact-impact neighbor set for a given particle is time varying. This can present significant problems with load balancing and communications overhead.

The aforementioned considerations are reflected in the results of performance testing on the code used here, which indicates that: (a) particle based calculations largely determine the required wall clock time, (b) most wall clock time is spent in the routines which calculate the densities and particle interaction forces, via summations over the contact-impact neighbor set, and (c) round-robin (or maximum entropy) initial data placement provides the best overall performance on nonuniform memory access systems.

The authors provided in reference [3] speedup data for test cases run on as many as 128 processors of an SGI Origin, applying a hybrid particle-element modeling methodology to problems as large as 0.5 million particles. As noted in the last section, more recent work has focused on much larger models, hence speedup tests have recently been performed at larger processor counts. These tests employed a 1024 processor SGI Origin and problem sizes as large as fifteen million particles. The test problem used for speedup measurements was the oblique sphere impact problem discussed in the section on validation simulations. Tables 5 and 6 show the results of test cases run for two different model sizes (4.7 and 14.6 million

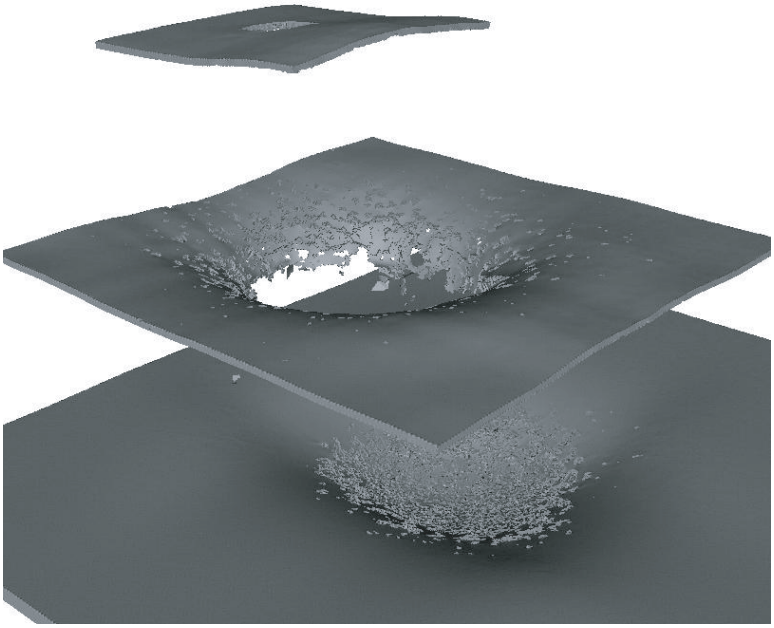

Fig. 9. Second oblique view of the simulation results for the dual plate shield problem.

particles), at various processor counts, ranging from 256 to approximately one thousand. In both cases speedup is measured relative to the wall clock time required for the test problem on 256 processors, and efficiency is defined as the ratio of measured to ideal relative speedup. The wall clock times shown are those required to complete 100 time steps of the simulation.

Table 5 shows good relative efficiency for the smaller test problem, when moving from 256 to 504 cpus, but very poor performance (in fact an increase in wall clock time) when the cpu count is then increased to 1008. Note that in the 1008 processor simulation each cpu is allocated approximately 5,000 particles, a load level apparently too light to allow for efficient parallel performance. Table 6 provides results for the larger test problem, where a minimum load level of nearly 15,000 particles per processor is maintained. In this case parallel performance is significantly improved, with a relative efficiency of almost seventy percent measured as the processor count is increased from 256 to 976. The preceding results indicate that the numerical method and OpenMP implementation tested here can efficiently address rather large scale problems. A dependence of parallel efficiency on processor load, observed here, is not unusual for engineering applications [22].

Table 5. Speedup measurements for a 4.7 million particle test problem

Number of processors	*Particles per processor*	*Wall clock time (hours)*	*Speedup (relative)*	*Efficiency (relative)*
256	18,281	0.7858	1.000	1.000
504	9,286	0.4858	1.618	0.822
1008	4,643	0.5008	1.569	0.400

Table 6. Speedup measurements for a 14.8 million particle test problem

Number of processors	*Particles per processor*	*Wall clock time (hours)*	*Speedup (relative)*	*Efficiency (relative)*
256	57,031	1.8439	1.000	1.000
512	28,516	1.0391	1.775	0.887
640	22,813	0.8803	2.095	0.838
976	14,959	0.7031	2.623	0.688

5. Conclusion

A number of new numerical methods based entirely or in part on particle kinematics are currently under development. They emphasize the fact that some important engineering problems are dominated by noncontinuum physics, such as fragmentation and contact-impact. Such problems may not fit easily into a classical continuum mechanics modeling framework. The present paper has described recent work aimed at extending and validating a hybrid numerical method for hypervelocity impact simulation. The results suggest that the method is accurate, numerically robust, and suitable for large scale parallel computation.

Acknowledgements

This work was supported by the Space Science Branch of NASA Johnson Space Center (NAG9-1244) and the National Science Foundation (CMS99-12475). Computer time support was provided by the Advanced Supercomputing Division of NASA Ames Research Center and the Texas Advanced Computing Center at the University of Texas at Austin.

References

[1] Fahrenthold EP, Horban BA. An improved hybrid particle-element method for hypervelocity impact simulation. *International Journal of Impact Engineering*, 2001; **26**: 169-178.

[2] Fahrenthold EP. Numerical simulation of impact on hypervelocity shielding. Proceedings of the Hypervelocity Shielding Workshop, 1998, Galveston, Texas, pp. 47-50.

[3] Fahrenthold EP, Shivarama R. Orbital debris impact simulation using a parallel hybrid particle-element code. *International Journal of Impact Engineering*, 2001; **26**: 179-188.

[4] Hayhurst CJ, Livingstone IH, Clegg RA, Fairlie GE, Hiermaier SH, Lambert M. Numerical simulation of hypervelocity impacts on aluminum and Nextel/Kevlar Whipple shields. Proceedings of the Hypervelocity Shielding Workshop, 1998, Galveston, Texas, pp. 61-72.

[5] Hallquist JO. Theoretical Manual for DYNA3D, 1983, Lawrence Livermore National Laboratory, Livermore, California.

[6] Fahrenthold EP, Koo JC. Hybrid particle-element bond graphs for impact dynamics simulation. *Journal of Dynamic Systems, Measurement, and Control*, 2000; **122**: 306-313.

[7] McGlaun JM, Thompson SL, Elrick MG. CTH: A three dimensional shock wave physics code. *International Journal of Impact Engineering*, 1990; **10**: 351-360.

[8] Wallin BK, Tong C, Nichols AL, Chow ET. Large multiphysics simulations in ALE3D. Presented at the Tenth SIAM Conference on Parallel Processing for Scientific Computing, Portsmouth, Virginia, March 12-14, 2001.

[9] Birnbaum NK, Francis NJ, Berber BI. Coupled techniques for the simulation of fluid-structure and impact problems. Structures Under Extreme Loading Conditions, ASME PVP-Vol. 361, 1998, San Diego, California, pp. 91-99.

[10] Faraud M, Destefanis R, Palmieri D, Marchetti M. SPH Simulations of debris impacts using two different computer codes. *International Journal of Impact Engineering*, 1999; **23**: 249-260.

[11] VanVlack LH. Elements of Materials Science and Engineering, 1975, Addison-Wesley, Inc., Reading, Massachusetts.

[12] Lyon SP, Johnson JD, editors. SESAME: The Los Alamos National Laboratory Equation of State Database, LA-UR-92-3407, Los Alamos National Laboratory, Los Alamos, New mexico.

[13] Shivarama RA. Hamilton's equations with Euler parameters for hybrid particle-element simulation of hypervelocity impact, PhD dissertation, Department of Mechanical Engineering, University of Texas at Austin, August, 2002.

[14] Shapiro PR, Martel H, Villumsen JV, Owen JM. Adaptive smoothed particle hydrodynamics with application to cosmology: methodology. *The Astrophysical Journal Supplement Series*, 1996; **103**: 269-330.

[15] Owen JM, Villumsen JV, Shapiro PR, Martel H. Adaptive smoothed particle hydrodynamics with application to cosmology: methodology II. *The Astrophysical Journal Supplement Series*, 1998; **116**: 155-209.

[16] Fahrenthold EP, Park Y-K. Simulation of hypervelocity impact on aluminum-Nextel-Kevlar orbital debris shields. *International Journal of Impact Engineering*, accepted for publication.

[17] Piekutowski AJ. Formation and Description of Debris Clouds Produced by Hypervelocity Impact, 1996, NASA Contractor Report 4707.

[18] Hertel ES. A Comparison of the CTH Hydrodynamics Code With Experimental Data, 1992, SAND92-1879, Sandia National Laboratories.

[19] Lambert M. Presentation at the 15th Inter-Agency Debris Committee Conference, European Space Agency (ESTEC), December 9-12, 1997.

[20] Fahrenthold EP. User's Guide for EXOS, 1999, University of Texas, Austin.

[21] Chandra R, Dagum L, Kohr D, Maydan D, McDonald J, Menon R. Parallel Programming in OpenMP, 2001, Academic Press, London.

[22] Gardner DR, Vaughan CT. The Optimization of a Shaped-Charge Design Using Parallel Computers, 1999, SAND99-2953, Sandia National Laboratories.

[23] Steinberg DJ. Equation of State and Strength Properties of Selected Materials, 1996, Lawrence Livermore National Laboratory, UCRL-MA-106439.