

Parallel Molecular Dynamic Code for Large Simulations using Truncated Octahedron Periodics

M.M. Micci, L.N. Long and J.K. Little^a

^aAerospace Department, The Pennsylvania State University,
233 Hammond Bldg, University Park, PA 16802, U.S.A.

The complete evaporation of three-dimensional submicron droplets under both subcritical and supercritical conditions has been modeled using parallel molecular dynamics. In this initial study the two-phase simulations consist entirely of argon atoms. The interatomic forces are based on a Lennard-Jones 12-6 potential, and the resultant atomic displacements are determined using a modified velocity Verlet algorithm. Large system modeling is enabled by parallel coding as well as the use of linked cell lists in combination with Verlet neighbor lists. A particle decomposition technique successfully provides nearly perfect load balancing. A non-cubic periodic boundary, specifically a truncated octahedron, is used to minimize periodicity effects.

1. INTRODUCTION

An analytical means of studying the diffusion process which occurs when a fuel droplet is exposed to a supercritical environment is desired. The combination of high speed collisions and lower droplet surface energy in this environment generates a desirable rapid mixing. Current rocket motors, gas turbines, diesel engines, and many projected advanced combustor designs operate supercritically. While concepts are well established from theory and experiment for subcritical conditions, there are no acceptable theories to fully describe the physics at and above the critical point [1]. Experimental investigations of such a dynamic process in high temperatures and pressures are difficult. Classic analytical techniques, such as continuum-based computational fluid dynamics, require far reaching assumptions and depend on suspect property data. The path chosen to achieve the desired analysis therefore involves molecular dynamics (MD) algorithms.

Molecular dynamics simulations are valuable tools, but they are computationally demanding. Periodic boundaries limit the domain to microscopic repeated systems, but the problem investigated herein involves a dual phase of a droplet and surrounding supercritical environment. Phases near the critical point exhibit long-range structures [2,3]. So the study of supercritical diffusion requires models which are large enough to avoid an overlapping of these structures. A more spherically based periodic boundary, the truncated octahedron, minimizes the required environment size and reduces the periodicity effects on the simulation. This study still requires, however, efficient coding on powerful computer architectures. Specifically, the capabilities of the IBM SP2 parallel computing platform were utilized.

One of the requirements for efficient parallel operations is the even loading of the computations across all the processors. Even with the relatively high densities of supercritical environments, the load imbalance on the various parallel processors due to density variations can become significant. Intermolecular forces are computed based on interactions with neighboring particles, so the computational load is proportional to the square of the density. For a typical case, the density of the droplet is five times greater than the surrounding environment. In response to this significant imbalance a load balanced parallel code was developed.

2. MOLECULAR DYNAMICS

The research supported by the codes presented here involves the molecular dynamic modeling of liquid droplets evaporating in gas environments. The displacement of each atom in a molecular dynamic simulation is simply based on Newtonian particle dynamics. The atomic accelerations are a direct function of inter-atomic forces. These forces are conservative (non-dissipative) and are therefore computed as the negative gradient of the potential energy function between two atoms. The Lennard-Jones potential is used. The atomic displacements are computed using a modified velocity Verlet algorithm. This modification is simply a reordering of the steps to remove the necessity to save acceleration information between time steps. To allow efficient modeling of large droplet simulations, the standard techniques of linked cell lists and Verlet neighbor lists were also implemented. The reader is referred to Allen and Tildesley [4] for detailed information on all of these techniques.

The study of droplet diffusion does not require the evaluation of solid boundaries. In fact, to reduce the effects on the diffusion simulation, a solid wall would require a larger domain (by as much as twenty atomic diameters [5]). Instead, a common boundary condition used in molecular dynamics is the *periodic boundary*. Typically this is a cubic shaped domain. A periodic shape termed the *truncated octahedron*, however, comes much closer than the cube to representing the shape desired for this study. The truncated octahedron is shown in figure 1. The included sphere in a cube accounts for only 52% of the cube's volume. So nearly half of the environment volume is wasted additional simulation. For a truncated octahedron, the included sphere requires 68% of the simulation volume. Perhaps even more important, the truncated octahedron has a much more desirable image ratio. This term refers to the ratio of the largest to the shortest distance between periodic images. The optimum value to reduce the effect of the periodicity on the simulation is one. For a cube this is the ratio of the cross-diagonal to the length of a side, 1.73. The truncated octahedron has an image ratio of just 1.29, by far the best of all the known periodic boundary candidates [6].

This periodic boundary, unfortunately, has an adverse affect on the computational load. The logic to correct the inter-atomic distances is a two step process [7]. The first is identical to the cubic boundary. The additional step involves the correction across the hexagon planes. This would normally be applied during the repetitive force calculations for the entire simulation domain. A much more efficient technique, however, is used. Referring once again to figure 1, the reader should note the shape fits completely within a cubic structure. In fact a complete set of primary atoms and their images perfectly fit within the

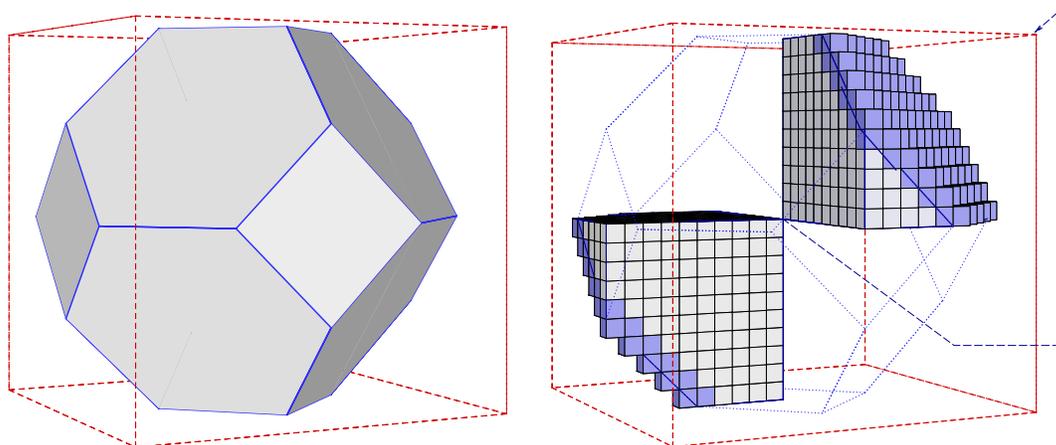


Figure 1. Truncated octahedron computational domain and cross quadrant periodicity.

enclosing cube. The right side of figure 1 highlights this fit by depicting the imaging of the $(-x, -y, -z)$ quadrant into the remaining half of the $(+x, +y, +z)$ quadrant. The small cubes depicted here are linked list cells. The dark, staircase shaped outer elements are cells which contain both primary and image atoms. The image computations are avoided by duplicating the atoms in the boundary cells and appending them to the atomic arrays. Further details of this technique are given by Little [8].

Molecular dynamics simulations involve very large computation loads. Every atom in a liquid simulation interacts with an average of fifty neighboring atoms at each time step. Also, the accurate modeling of the collision dynamics is an essential ingredient required for simulation accuracy. Since these collisions occur over very small time frames, the temporal discretization requires step sizes on the order of only a few femtoseconds (10^{-15} seconds). The large loads, both incremental and overall, would seem to make molecular dynamics a very good candidate for parallel computations. The unstructured nature of MD, however, significantly complicates the achievement of efficient performance. Most notably, the proportionality of the computation load to the square of the density makes load balancing difficult.

In order to load balance the code, the atoms were randomly distributed across the processors. The required neighbor information for each of these atom sets is provided by global arrays containing all of the atomic positions. The generating of these arrays is the only communication required in the technique, and global message passing library calls can perform this using just three lines of code. A thorough review of the technique is provided by Plimpton [9]. He uses the term *atom decomposition* in his description and compares it to two other decomposition strategies: spatial and force decompositions. The interested reader is encouraged to review this reference.

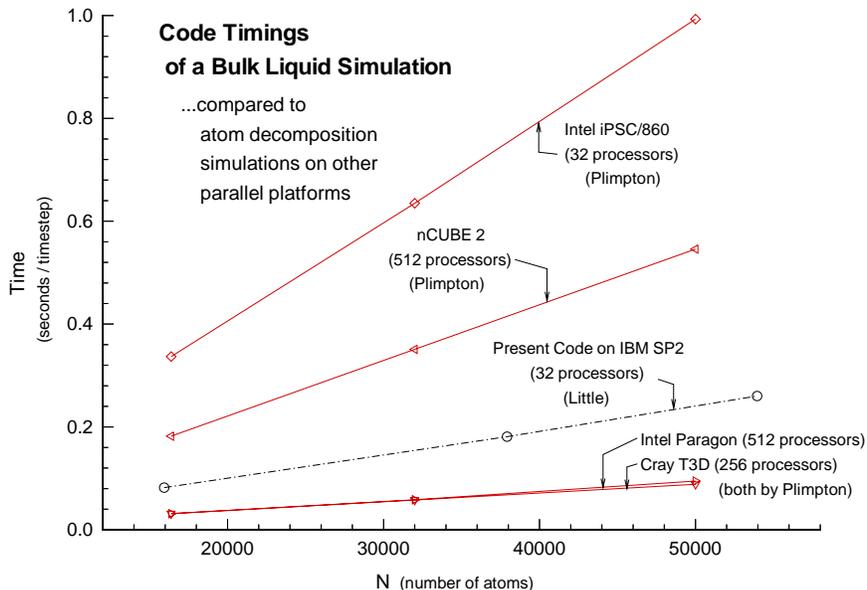


Figure 2. Parallel code performance comparisons. Code performance on the IBM SP2 compared to optimized codes on other parallel architectures.

3. Code Performance

The performance measures begin with a series of comparisons to runs performed by Plimpton [9]. He provides a detailed array of performance measures from a cubic simulation on both serial and parallel platforms. Plimpton's benchmark runs use a bulk liquid structure with a reduced density of $\rho^* = 0.8442$ and a reduced temperature of $T^* = 0.72$. Argon is simulated here with a force cutoff of 2.5σ . A combination linked list and Verlet list with a list radius of 2.8σ is implemented to provide optimal performance on large systems. The time step used is 10 femtoseconds, and the Verlet list is updated at a constant value of 20 steps. An atom decomposition technique is used in the parallel runs, while a vectorized serial algorithm from Grest et al. [10] is used on the single processors of the Cray Y-MP and Cray C90. The parallel runs were performed on the Cray T3D, the nCUBE 2, the Intel iPSC/860, and the Intel Paragon [9].

The performance of the truncated octahedron code on the IBM SP2 was first compared to Plimpton's serial runs on the Cray Y-MP and Cray C90. The parallel code beat the Cray Y-MP performance on just 8 processors, while the stronger Cray C90 performance was surpassed with the 24 processor set. Plimpton states that these performance values of the serial code are the fastest known to date for the molecular dynamics simulations on conventional vector supercomputers. Since the article was updated in June 1994, these comparisons reveal that the SP2 code performs quite well.

The benchmark problem was also run on various parallel architectures. The timings for a particle decomposition code are shown in figure 2. The present code performance on the

32 processor IBM SP2 is also included in the plot. Again, the truncated octahedron code performs fairly well. The linear scaling with problem size evidenced by the benchmark runs is matched by the SP2 case. Also, the simulation rate is consistently double the rate of the 512 processor nCUBE 2 and nearly four times faster than the 32 processor Intel iPSC/860. Only the 512 processor Intel Paragon and the 256 processor Cray T3D show better performance. However, the 512-node Paragon, the 256-node T3D, and the 32-node SP2 have peak speeds of 38 gflops, 38 gflops, and 8 gflops, respectively. So while the Cray and Intel machines are 5 times faster (theoretically) than the IBM, they only run about 2 times faster for this application.

This poor performance is partially due to the poor communications scaling associated with the atom decomposition technique. Plimpton also shows that a force decomposition algorithm run on these large processor sets can reduce these times by another 25% for the Cray T3D and 40% for the Intel Paragon. Also, doubling the processor sets for the benchmark runs reveals virtually no improvements using the particle decomposition technique, but the more scalable force decomposition code shows promising results. A 512 processor Cray T3D and a 1024 processor Intel Paragon yield additional savings of nearly 40% and 25% respectively. These timings provide incentive to expand the droplet diffusion research onto these more massively parallel platforms with a force decomposition code.

As mentioned, all of these benchmark comparisons are for bulk liquid simulations. The diffusion problem consists of distinctly varied density profiles. The droplet requires significantly more computations than the lower density surroundings. The reader is reminded that this load is proportional to the square of the density, so even the relatively compact near-critical environment displays this load imbalance across the geometries. The particle decomposition technique was specifically developed to meet this challenge. Simulation runs revealed that nearly perfect load balancing was achieved. Consistently less than 2% of the simulation times involved processors waiting for other processors to complete their computations. This low imbalance level was seen regardless of the number of processors used. Unfortunately, the anticipated constant communication load was also demonstrated. The code performance improvements were minimal beyond 32 processors as a result. This lack of processor level scalability, however, becomes less significant for more complex molecular dynamic models.

4. Simulation Results

Measuring the diameter of the droplets during the molecular dynamic diffusion simulations was initially performed by defining a surface using a technique by Maruyama [11]. He measures the local atomic density for each atom using the cutoff sphere as the sampling domain. This technique proved useful for the subcritical simulations, but was ineffective for the supercritical tracking. During these cases the droplets quickly lose their spherical geometry due to the surface tension reduction. The measuring of an average radial position is then misleading.

An alternative means which both matched the subcritical radius tracking and provided a logical means of following the supercritical progress was developed to counter the problem. Instead of using a measured surface radius, the volume of the liquid structure is tracked as

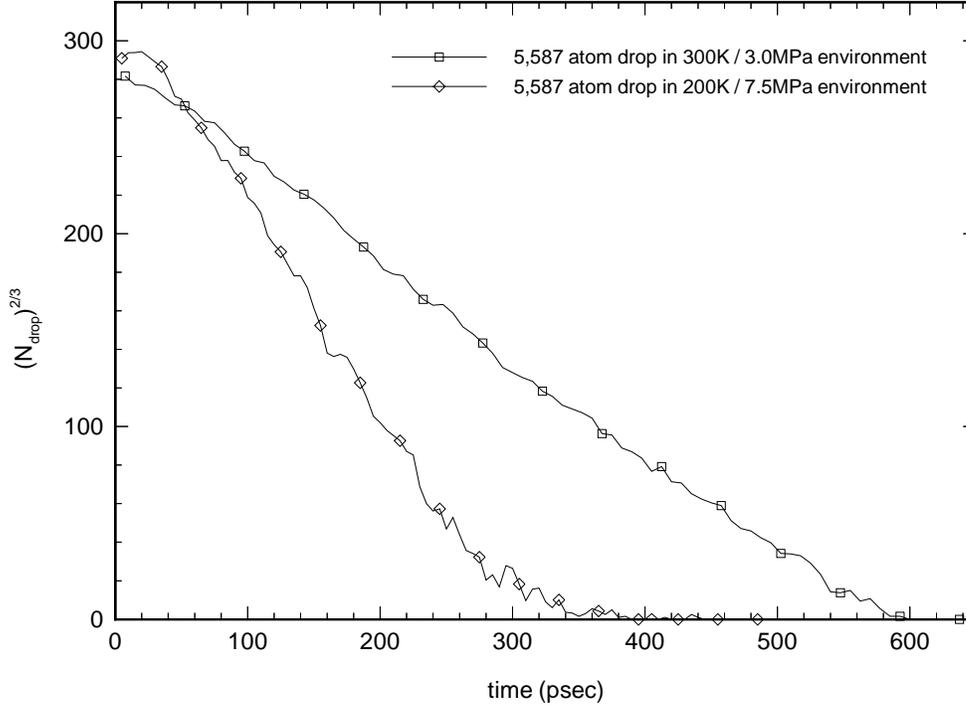


Figure 3. Supercritical and subcritical regressions. Comparison of the $N_{drop}^{2/3}$ regression of the 5,600 atom droplet in the 200 K, 7.5 MPa supercritical environment to the regression in the 300 K, 3.0 MPa subcritical surroundings.

the enumeration of atoms whose local densities exceed a given level. This value, termed N_{drop} , can be regarded as an indicator of droplet volume since the drop is simply the total collection of these particles. Modifying the parameter further to $N_{drop}^{2/3}$ provides a similar indicator of surface area.

The results from two simulation runs are presented using the $N_{drop}^{2/3}$ regression factor in figure 3. The first run models the evaporation of a 5,600 atom drop placed into a subcritical environment at 300 K and 3.0 MPa. The second simulates the same droplet but placed into a supercritical environment of 200 K and 7.5 MPa.¹ Both plots show a linear regression. This is expected for the subcritical simulation (matching the D^2 evaporation law), but the supercritical profile was not anticipated. Also, the evaporation in the lower temperature supercritical environment progressed more rapidly than the high temperature subcritical environment.

Figure 4 is a contour representation of the density, temperature and a qualitative surface tension measure (see Little [8]). This helps explain the more rapid evaporation of the colder supercritical run. The surface tension reaches a steady level in the subcriti-

¹The reader should note that the pressure is the determining factor in the categorization. The critical point for argon is 151 K and 4.9 MPa.

cal simulation and allows the retaining of the spherical integrity of the droplet. In the supercritical run, however, the surface tension quickly dissipates and the droplet surface deforms significantly during the evaporation process. The lower surface energy causes the collisions in this environment to be more effective. This is why supercritical evaporation is so desirable.

5. Conclusions

The code developed for the supercritical evaporation successfully meets the required scope as a first step in the evaluation of supercritical combustion. Significant research continues, however, to continue to meet the challenge. For example, the deformed shape displayed in the supercritical run detailed herein has generated increased interest for further research. Specifically, the effect of larger-scale simulations on this shape is a question addressed further by Little [8]. Also, research is on-going which will adapt the code discussed herein to more complicated molecular structures and dual species evaporation. The measuring of transport coefficients in all of these simulations is also planned and will allow the transference of results to macroscopic continuum based models.

REFERENCES

1. Levelt Sengers, J. M. H. *Critical Behavior of Fluids: Concepts and Applications*. Kluwer Academic Publishers, Netherlands, 1994.
2. Sengers, J. V., and Levelt Sengers, J. M. H. Thermodynamic behavior of fluids near the critical point. *Annual Rev. Phys. Chem.*, **37**, pp. 189–222, 1986.
3. Cummings, P. T. *Molecular Simulation of Near-Critical and Supercritical Fluids*. Kluwer Academic Publishers, Netherlands, 1994.
4. Allen, M. P., and Tildesley, D. J. *Computer Simulation of Liquids*. Oxford University Press, New York, NY, 1984.
5. Haile, J. M. *Molecular Dynamics Simulation: Elementary Methods*. John Wiley and Sons, New York, NY, 1992.
6. Larsen, B. Summary of session I. In *NRCC Proceedings No. 9, Menlo Park, California.*, 1980.
7. Adams, D. J. Periodic, truncated–octahedral boundary conditions. In *NRCC Proceedings No. 9, Menlo Park, California.*, 1980.
8. Little, J. K. *Simulation of Droplet Evaporation in Supercritical Environments using Parallel Molecular Dynamics*. Ph.D. thesis, Pennsylvania State University, 1996.
9. Plimpton, S. Fast parallel algorithms for short–range molecular dynamics. *Journal of Computational Physics*, **117**, pp. 1–19, 1995.
10. Grest, G. S., Dunweg, B., and Kremer, K. Vectorized link cell fortran code for molecular dynamics simulations for a large number of particles. *Computer Physics Communications*, **55**, p. 269, 1989.
11. Maruyama, S. Surface phenomena of molecular clusters by molecular dynamics method. In *The Japan–U.S. Seminar on Molecular and Microscale Transport Phenomena (J-5).*, 1993.

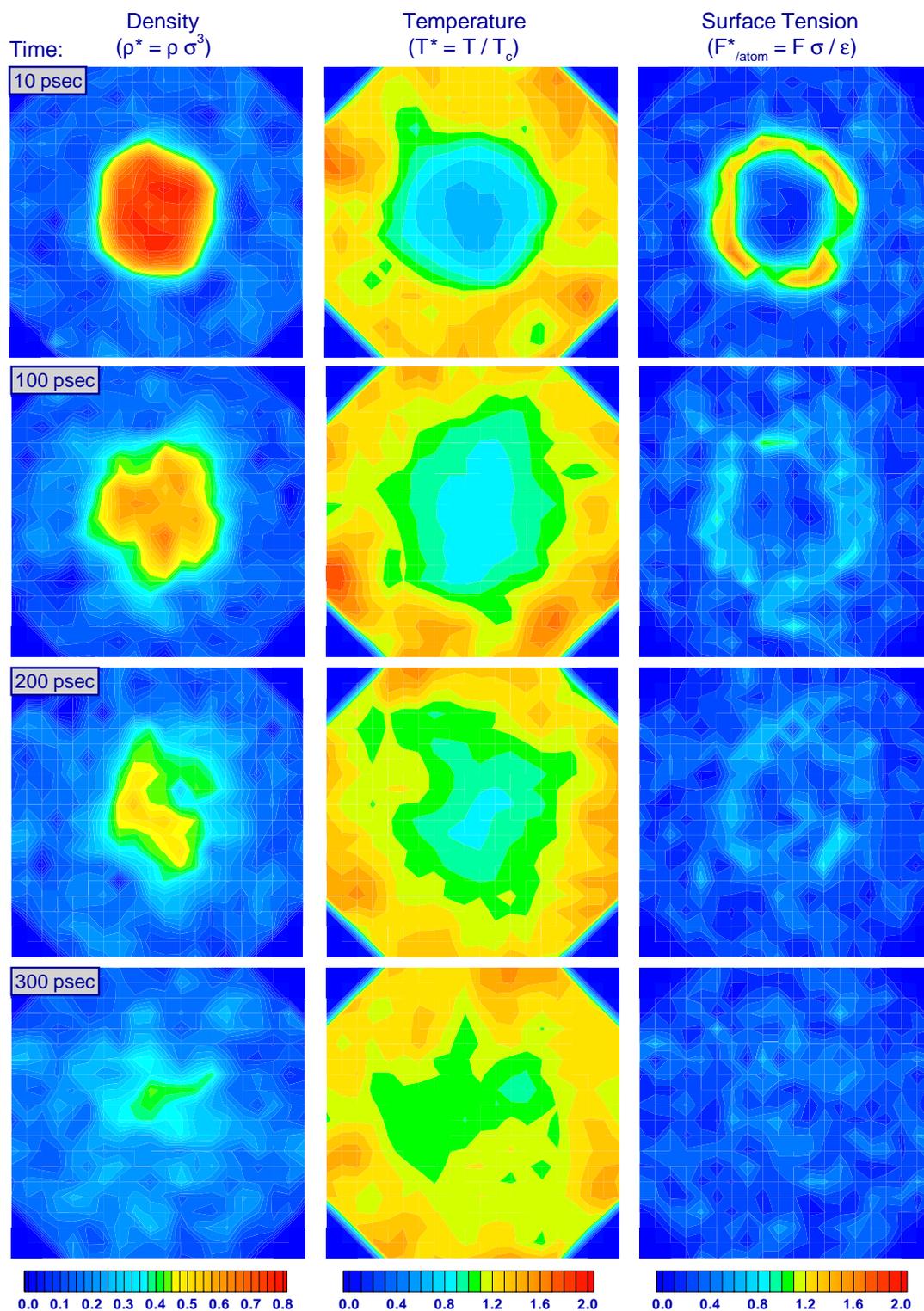


Figure 4. Supercritical contours: Full evaporation. A 5,587 atom drop equilibrated to 100 K and 1200 kg/m³ is placed into a 200 K, 7.5 MPa, and 236 kg/m³ supercritical environment.