



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Petascale Simulation Initiative Summary Final Report

J. May

February 8, 2008

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Auspices Statement

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This work was funded by the Laboratory Directed Research and Development Program at LLNL under project tracking code 04-ERD-102.

Petascale Simulation Initiative
Summary Final Report

John May, February 4, 2008

The Petascale Simulation Initiative LDRD project began in May 2004 and ended in April 2007. The overall goal of the project was to develop a computational model that would help LLNL simulation codes use petaflop-scale computers effectively, with specific emphasis on multiscale simulation.

An early deliverable of the project was a report that surveyed a range of LLNL simulation codes so that the project would better understand their structure and needs. This report was titled “Survey of Selected LLNL Unclassified Parallel Simulation Codes,” UCRL-TR-207192.

Based on the findings of this report, the project proceeded along two complementary paths. One goal was to develop a technique to reduce the amount of redundant calculation in a multiscale simulation by caching certain fine-scale simulation results as they were generated and then interpolating new values, where appropriate, based on cached data. This technique, which we call “Adaptive Sampling,” was successfully applied in a materials modeling simulation, and the results are detailed in a journal paper entitled “Embedded polycrystal plasticity and adaptive sampling,” UCRL-JRNL-226715. In this paper we showed an example multiscale simulation in which we were able to achieve a 40x speedup over the standard technique for doing the same problem and with very little loss of accuracy. A second paper that further describes adaptive sampling was recently accepted as a journal publication: “Adaptive sampling in hierarchical simulation,” UCRL-JRNL-232594.

The second thrust of the project was to develop a parallel programming model that supports dynamic creation and coordination of sequential and parallel subcomputations to form larger “federated” simulations. We call this model “Cooperative Parallelism.” The model is described in “Overview of the Cooperative Parallel programming model,” UCRL-CONF-230029, which was submitted to Supercomputing 2007 (but not accepted for publication). We will be revising the paper and submitting it for publication again. In the mean time the model is under study for use in both AX division and B division, and possibly other programs as well.

Finally, a higher-level view of our work appears in book chapter entitled “Multiscale simulation and petascale computing,” UCRL-BOOK-227348, in *Petascale Computing: Algorithms and Applications*. The description in this chapter sets our work in the larger context of petascale simulation.