Advanced Computational Multi-Body Dynamics for Next Generation Simulation-Based Engineering

Dan Negrut, Hammad Mazhar

1. Abstract

Motivated by the microprocessor industry transition from single to many-core chip designs, this project investigates a framework that leverages heterogeneous computing to tackle many-body dynamics problems. The heterogeneous computing infrastructure relies on a combination of Central Processing Units (CPUs) and Graphics Processing Units (GPUs), the latter regarded as co-processors or accelerators, capable of delivering substantial efficiency gains in simulating many-body dynamics problems. Examples of such problems include granular material dynamics, simulation of fluid-solid interaction, and molecular dynamics analysis.

3. A Heterogeneous CPU/GPU HPC Environment for Computational Dynamics Applications

Computational dynamics aims at determining how a system of mutually interacting elements changes in time. Heterogeneous computing, that is, parallel computing drawing on both CPU and GPU hardware, becomes relevant when the number of interacting elements is very large. High-performance heterogeneous computational dynamics requires:

(a) the ability to partition the problem according to a one-to-one mapping (i.e., spatial subdivision: one subdomain to one computing unit).

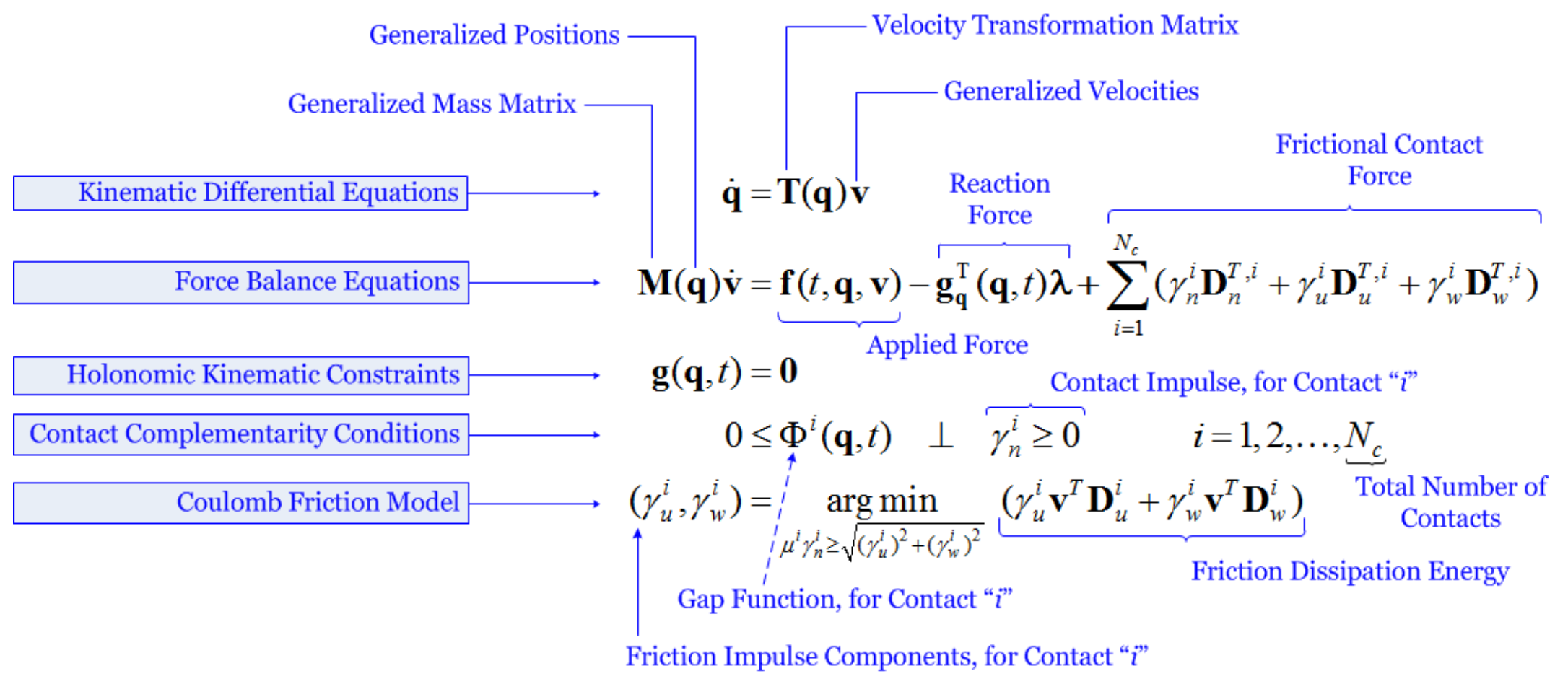
(b) a protocol for passing data between any two computing unites.

(c) algorithms for element proximity computation.

(d) the ability to carry out post-processing for data analysis and visualization in a distributed fashion.

4. Computational Dynamics: Many-Body Dynamics Analysis

When handling rigid and flexible bodies interacting through contact and friction, the physics-based system simulation requires the solution of a combined set of differential-algebraic equations and inequality constraints:



$$\dot{\mathbf{q}} = \mathbf{T}(\mathbf{q})\mathbf{v}$$

$$\mathbf{M}(\mathbf{q})\dot{\mathbf{v}} = \underbrace{\mathbf{f}(t,\mathbf{q},\mathbf{v})}_{\text{Applied Force}} - \overbrace{\mathbf{g}_{\mathbf{q}}^{T}(\mathbf{q},t)\boldsymbol{\lambda}}^{\text{Reaction Force}} + \overbrace{\sum_{i=1}^{N_c}(\gamma_n^i \mathbf{D}_n^{T,i} + \gamma_u^i \mathbf{D}_u^{T,i} + \gamma_w^i \mathbf{D}_w^{T,i})}^{\text{Frictional Contact Force}}$$

$$\mathbf{g}(\mathbf{q},t) = \mathbf{0}$$

$$0 \le \Phi^i(\mathbf{q},t) \perp \gamma_n^i \ge 0 \qquad i = 1,2,\ldots,N_c$$

$$(\gamma_u^i, \gamma_w^i) = \underset{\mu^i\gamma_n^i \ge \sqrt{(\gamma_u^i)^2+(\gamma_w^i)^2}}{\arg\min} \underbrace{(\gamma_u^i \mathbf{v}^T \mathbf{D}_u^i + \gamma_w^i \mathbf{v}^T \mathbf{D}_w^i)}_{\text{Friction Dissipation Energy}}$$

The goal is to devise numerical solution procedures for the above problem that map well onto heterogeneous hardware; i.e., that leverage both CPU and GPU computing.

2. Example of Heterogeneous Supercomputer:

Leveraging financial support from TARDEC, NVIDIA® and Microsoft®, a 5,760 GPU Scalar Processor Cluster with 48 CPU Cores running Windows HPC2008 was assembled at the Modeling, Simulation and Visualization Center at the University of Wisconsin-Madison (see Fig. 1).

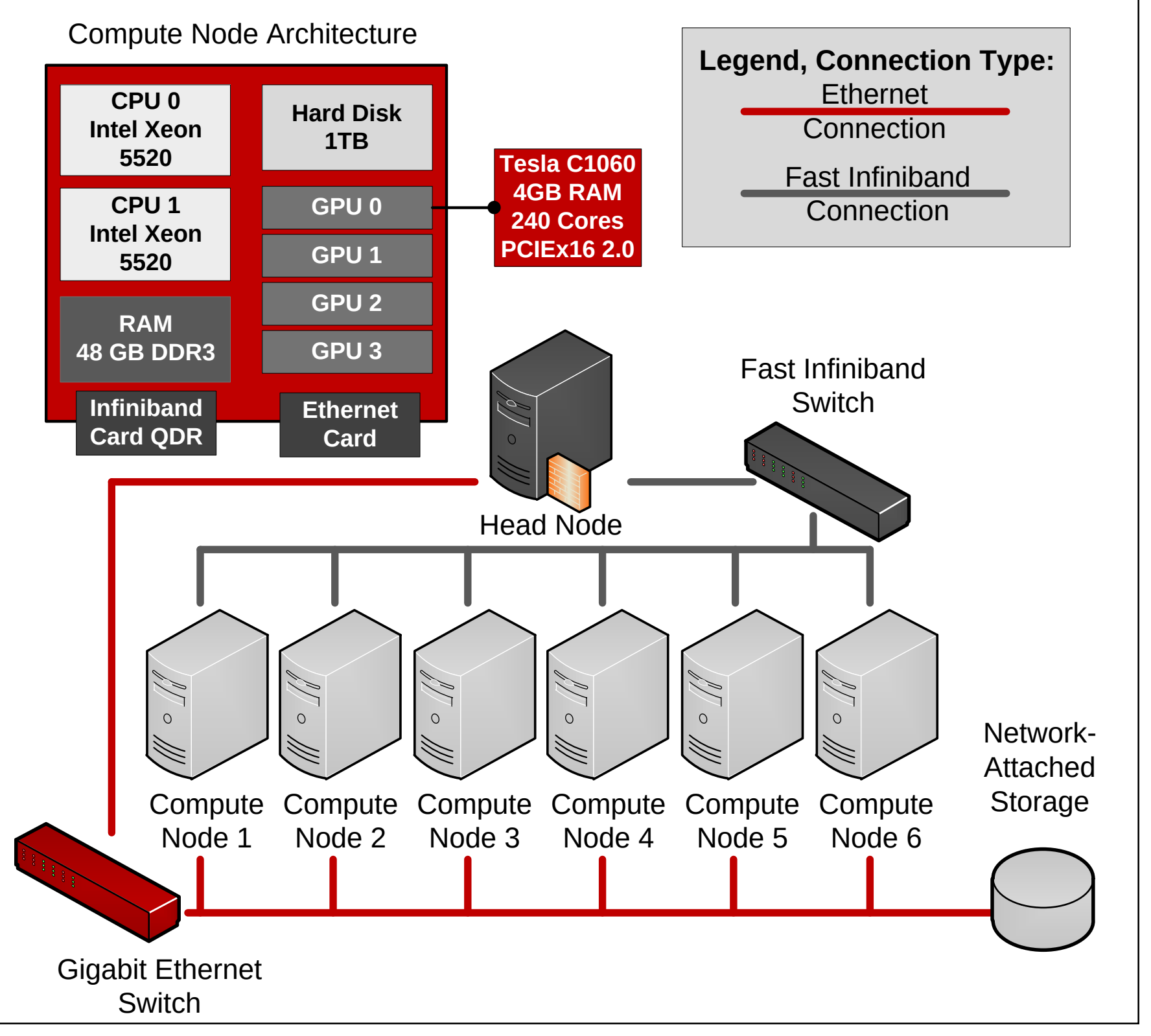
This 24-GPUs heterogeneous cluster:

a) Serves as the vehicle used to experiment with the heterogeneous computing paradigm

b) Is used to perform compute-intense simulations that require parallel processing

c) Is used in the course "High Performance Computing for Engineering Applications" offered in Spring 2011 at UW-Madison. Enrolment: 45 students from 14 departments

Fig.1 CPU-GPU Cluster



5. Collision Detection for Many-Body Problems

Leveraging the concept of heterogeneous computing a set of four GPUs was managed by a four core Intel Xeon CPU to simultaneously solve a collision detection problem with 5.6 billion contacts in approximately 3 minutes. This represents the approximate number of collisions in one cubic meter of sand. The software and hardware stack are shown in Fig.4. The solution implemented uses Open Multi-Processing (OpenMP) and NVIDIA's Compute Unified Device Architecture (CUDA) for software support.

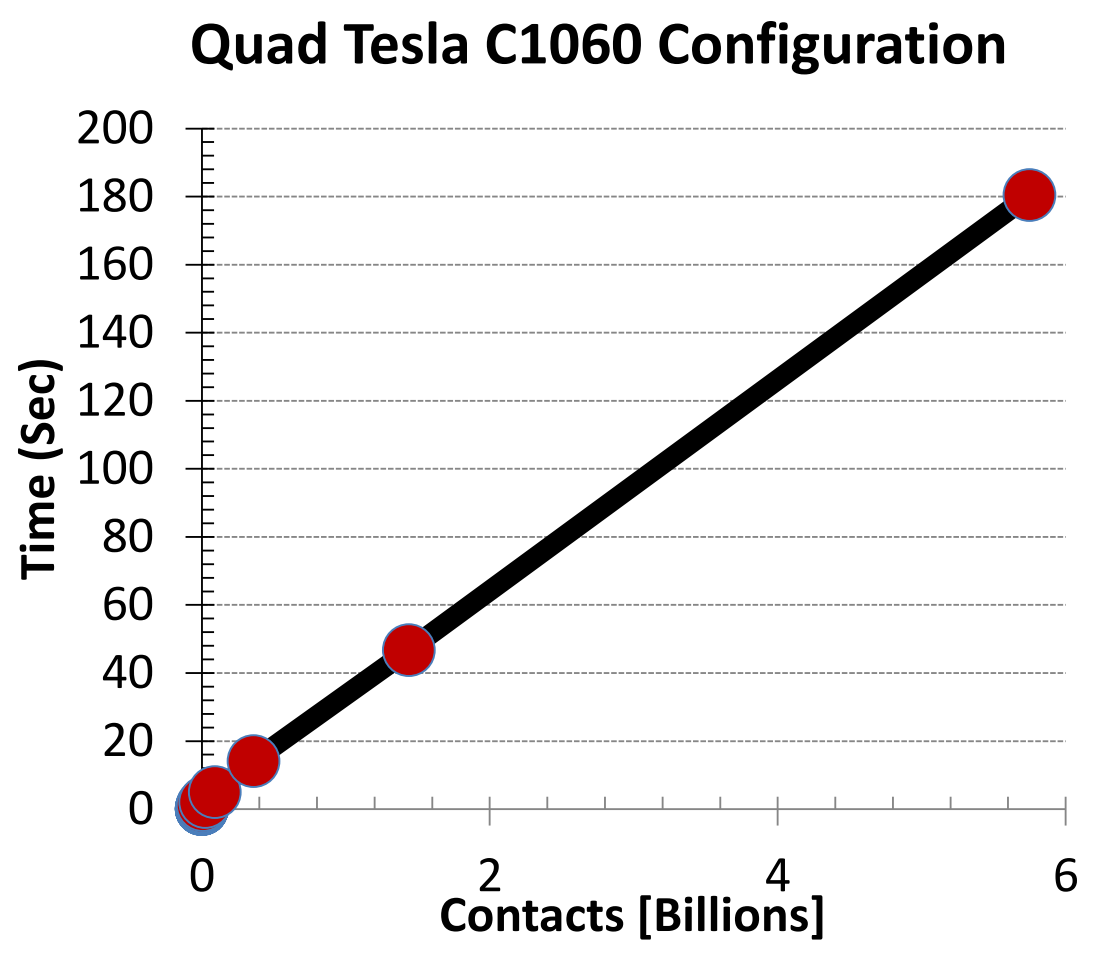


Fig.2 Contact Detection Using 4 GPUs

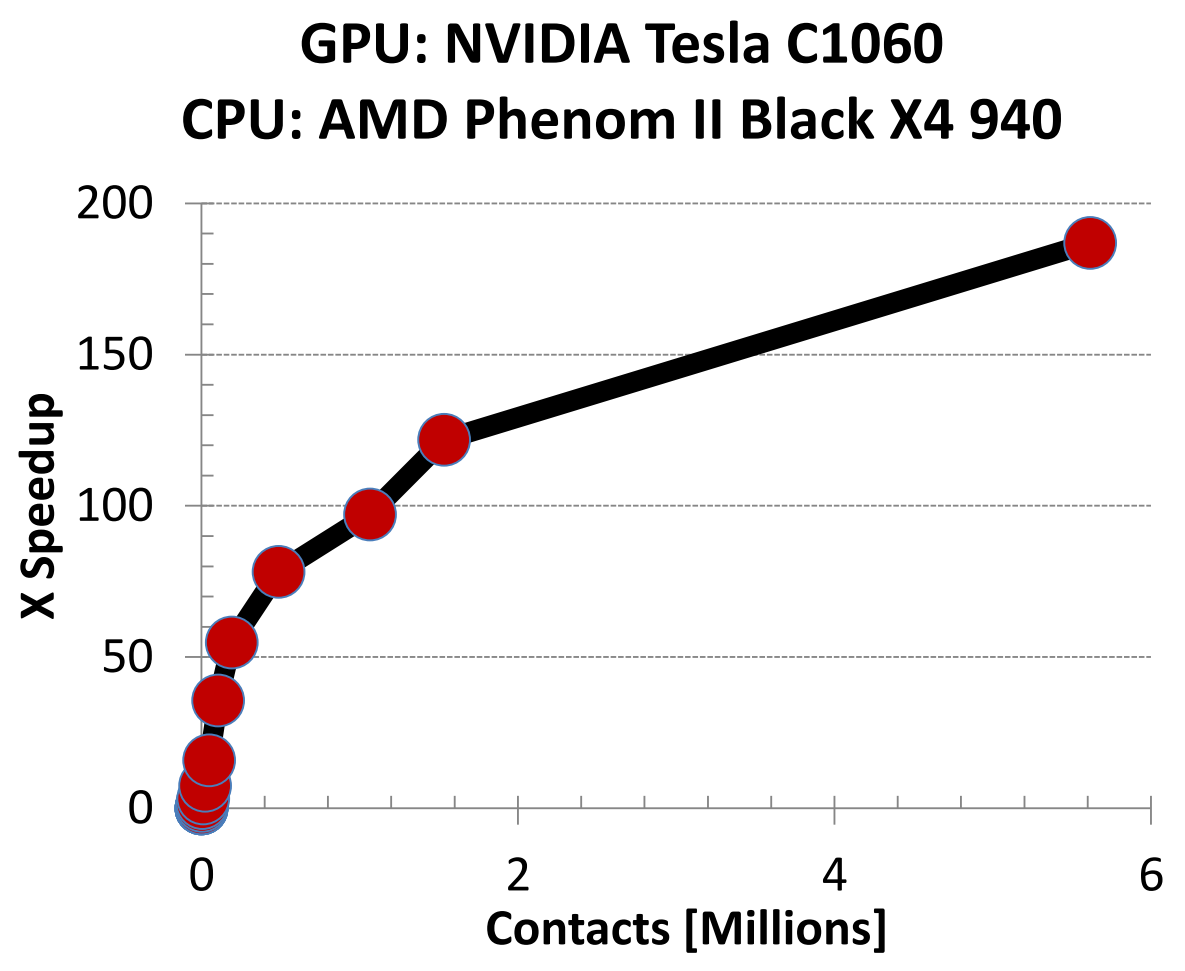


Fig.3 Speedup of Algorithm when compared to Bullet Physics

6. Technology Demonstration: Tracked Vehicle Simulation on Granular Terrain

The multibody dynamics solver and collision detection have been implemented to execute on 240 parallel Scalar Processors on one Tesla C1060 NVIDIA GPU. Examples of a double tracked vehicle operating on granular terrain were run with close to 0.5 million pebble elements. The method used for collision detection called for a spherical decomposition of the entire vehicle geometry, resulting in more than 3 million spheres being used in the collision model (Fig.6). The terrain was represented as a collection of discrete particles that matched a given surface profile and had a variable fidelity with depth.

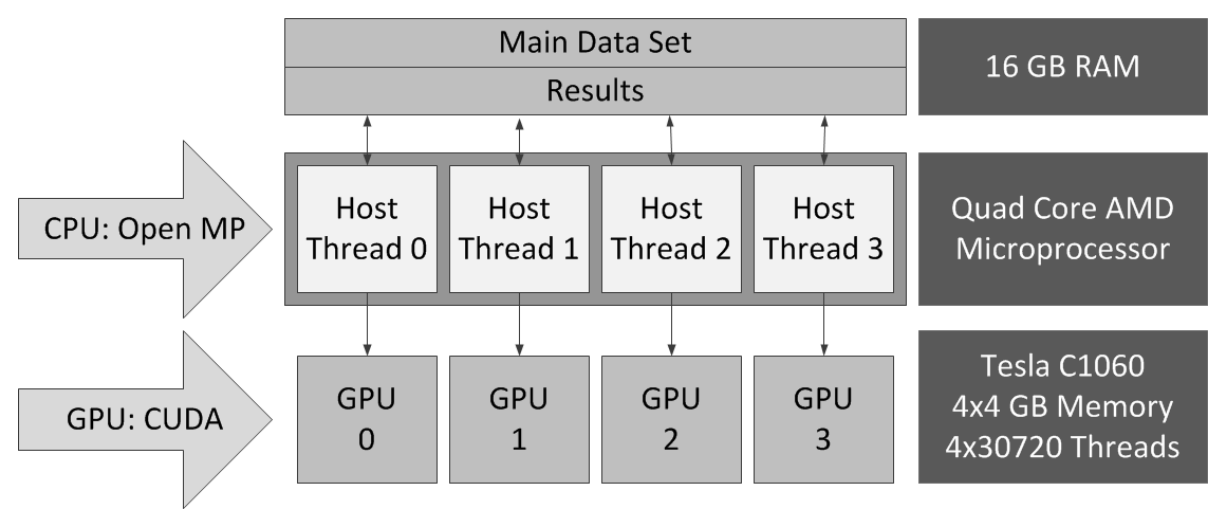


Fig.4 Hardware and Software Stack used for multi-GPU computation

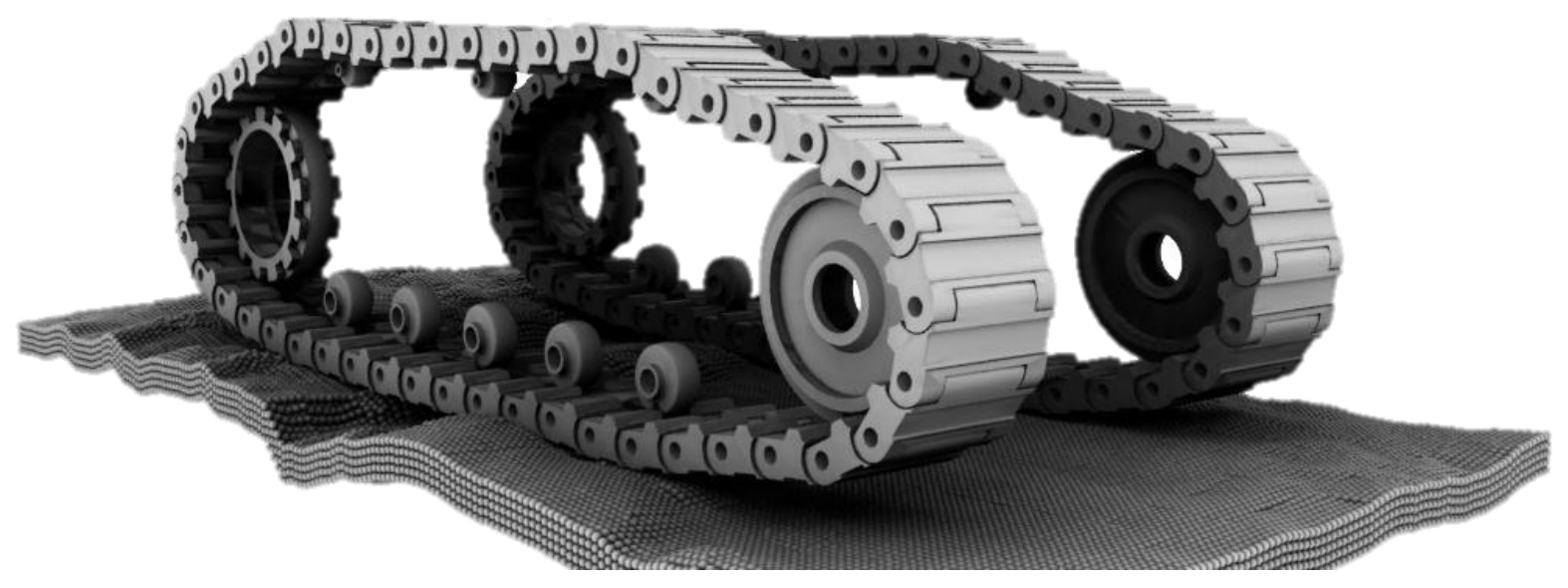


Fig.5 Double-Track on Deformable Terrain

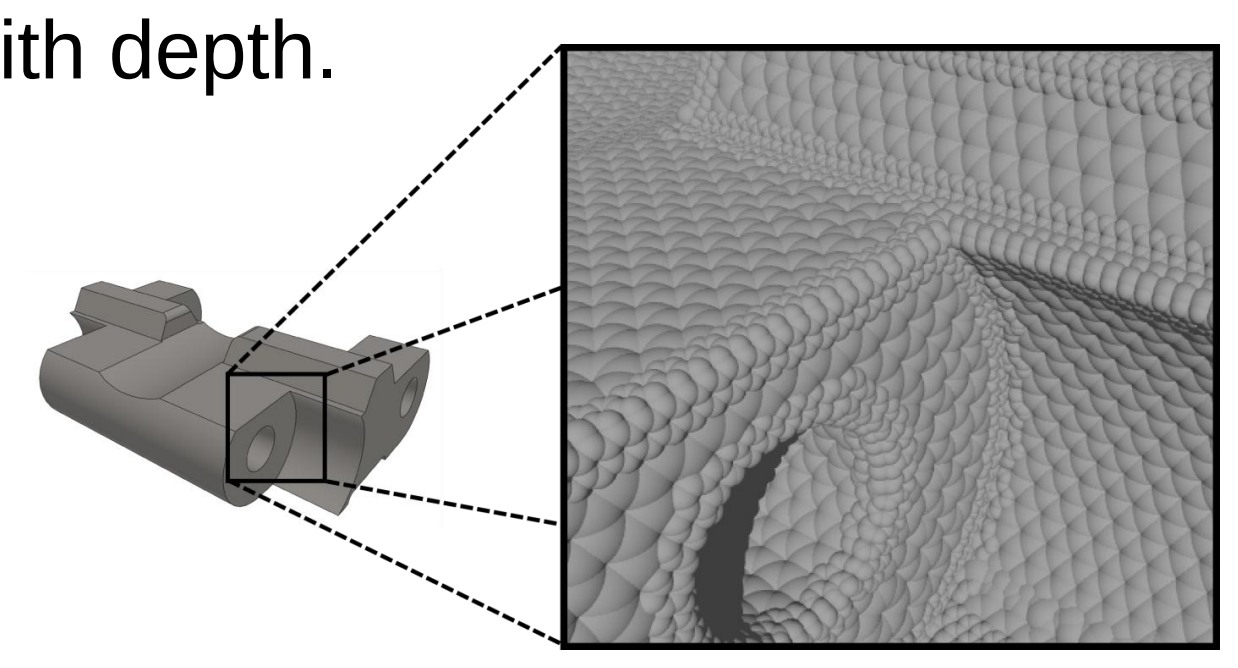


Fig.6 Spherical Decomposition of Track Shoe

7. Computational Dynamics: Fluid-Solid Interaction Analysis

Fluid-Solid Interaction Analysis relies on the Smoothed Particle Hydrodynamics (SPH) method. In this Lagrangian method, each particle carries field variables (density, internal energy, velocity, etc.). A kernel-function approach defines the influence area of each particle. Field variables and their derivatives can be approximated as outlined in Fig.8. The approach is embarrassingly parallel since each computational thread can independently process one particle. The original partial differential problem is reduced to a ordinary differential problem by expressing the right-hand side using SPH approximations for field function derivatives and an appropriate state equation.

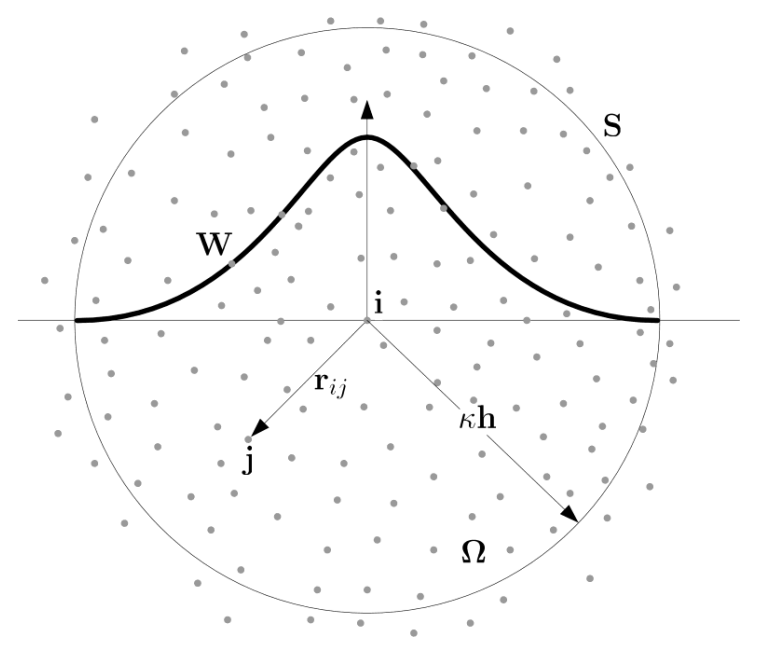


Fig.7 Illustration of the kernel, W, and its support domain, Ω, which is bordered by S. For 3D problems the support domain is a sphere with radius κh, where κ is a constant associated with the kernel and h is the smoothing length.

$$\langle f(x)\rangle = \int_\Omega f(x')W(x-x',h)dx' \approx \sum_{j=1}^{N} f(x_j)W(x-x_j,h)\Delta V_j = \sum_{j=1}^{N}\frac{m_j}{\rho_j}f(x_j)W(x-x_j,h)$$

$$\langle \nabla f(x)\rangle = \int_\Omega [\nabla f(x')]W(x-x',h)dx' = -\int_\Omega f(x')\cdot\nabla W(x-x',h)dx' \approx -\sum_{j=1}^{N}\frac{m_j}{\rho_j}f(x_j)\nabla W(x-x_j,h)$$

$$\text{Conservation of mass:} \quad \frac{d\rho}{dt} = -\rho\frac{\partial v^\beta}{\partial x^\beta} \quad \Rightarrow \quad \frac{d\rho_i}{dt} \approx \sum_j m_j \mathbf{v}_{ij}\cdot\nabla_i W_{ij}$$

$$\text{Conservation of momentum:} \quad \frac{dv^\alpha}{dt} = \frac{1}{\rho}\frac{\partial\sigma^{\alpha\beta}}{\partial x^\beta} + \frac{f^\alpha}{\rho} \quad \Rightarrow \quad \frac{d\mathbf{v}_i}{dt} \approx -\sum_j m_j\left(\frac{p_j}{\rho_j^2}+\frac{p_i}{\rho_i^2}+\Pi_{ij}\right)\nabla_i W_{ij} + \frac{\mathbf{f}}{m_i}$$

$$\text{Conservation of energy:} \quad \frac{du}{dt} = \frac{\sigma^{\alpha\beta}}{\rho}\frac{\partial v^\alpha}{\partial x^\beta} \quad \Rightarrow \quad \frac{du_i}{dt} \approx \frac{1}{2}\sum_j m_j\left(\frac{p_j}{\rho_j^2}+\frac{p_i}{\rho_i^2}+\Pi_{ij}\right)\mathbf{v}_{ij}\cdot\nabla_i W_{ij}$$

Fig.8 Discretized Equations Governing Smooth Particle Hydrodynamics

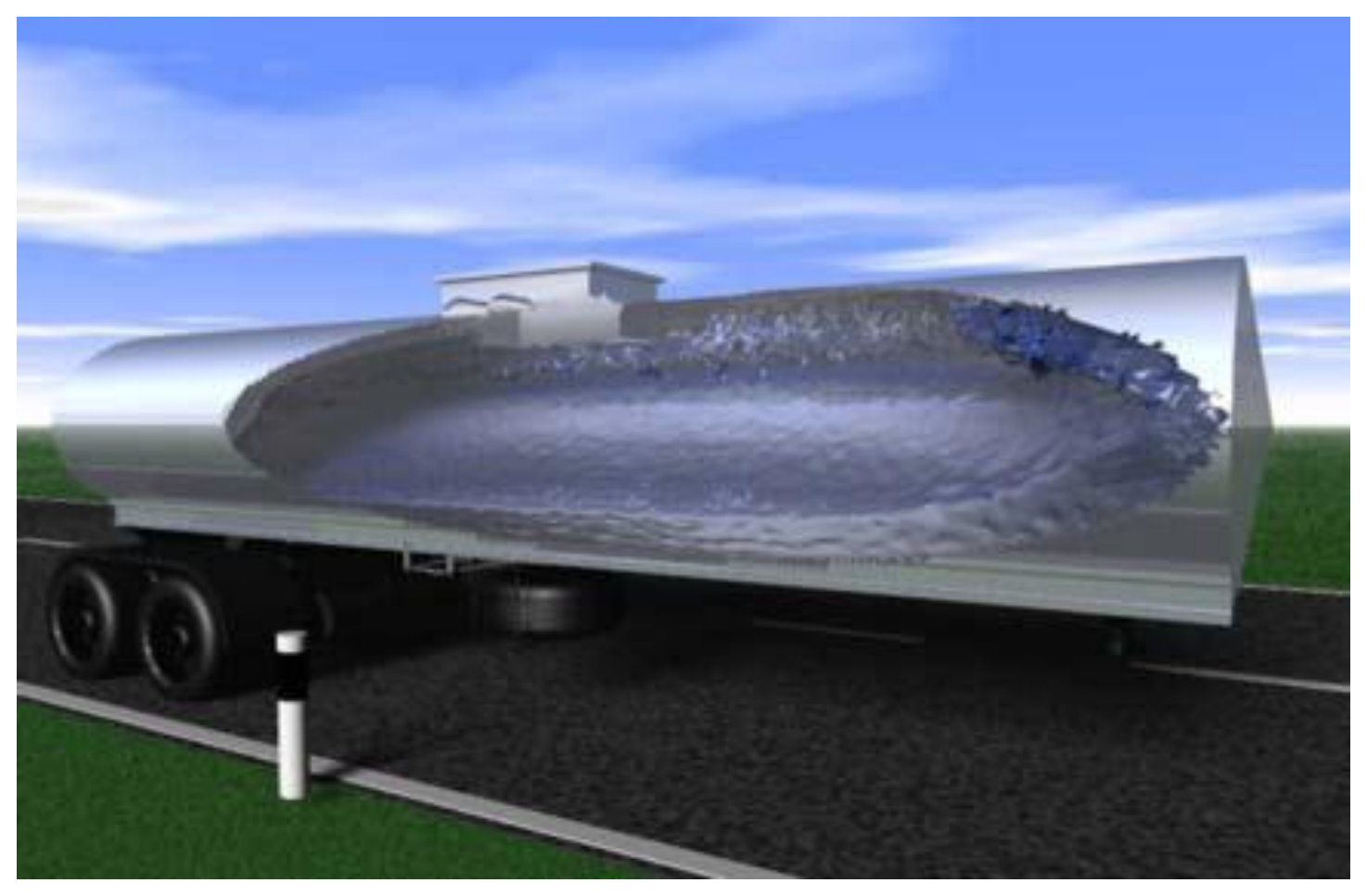


Fig.9 Tanker with simulated fluid

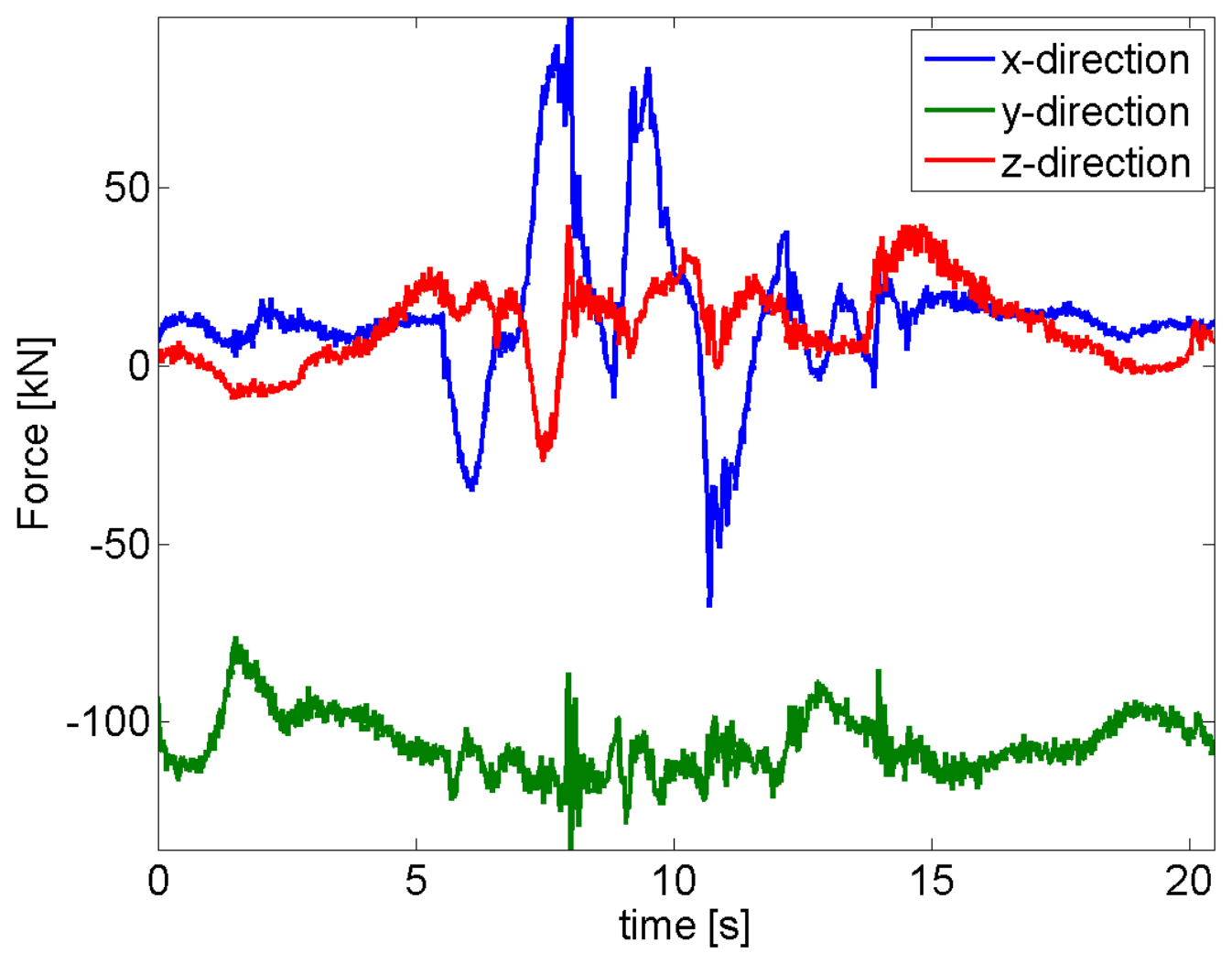


Fig.10 Forces Exerted by Fluid on Moving Tank

Fig.11 Fluid-Solid Simulation Of Highly Viscous Material Being Scooped By Digger

8. Technology Demonstration: Fluid Sloshing in Moving Tanker

A classic example of fluid structure interaction is the tank fluid sloshing problem (Fig.9). Approximately 230,000 points define the tanker boundary geometry and 300,000 SPH particles discretize the liquid. After the fluid particles have settled, the tank is set into a motion that mimics a typical driving maneuver and the fluid starts to slosh around in the tank. Figure 10 reports the sloshing forces, in three directions, that act on the tanker. Gravity acts in negative y-direction. The tanker starts to accelerate in positive z-direction and drives through curves in positive and then negative x-direction. The simulation was performed in approximately 19 hours using a NVIDIA GeForce 8800 GTX graphics card with 128 Scalar Processors.

9. Conclusion

The key observation that motivates this effort is that commodity hardware available today has tremendous compute power. What is missing is the solution methods and software that harnesses these heterogeneous supercomputers. The long-term expected outcome of this project is a software infrastructure that can be used by any researcher interested in using heterogeneous CPU/GPU hardware for HPC in Computational Dynamics applications. Preliminary results obtained with an early version of this software infrastructure show one to two orders of magnitude reductions in simulation times for real-life dynamics applications.

Contact Information

Dan Negrut
Simulation-Based Engineering Laboratory
Dept. of Mechanical Engineering
University of Wisconsin-Madison

Acknowledgements:

The results reported have been obtained using hardware donated by NVIDIA®, which along with financial support from US Army TARDEC and Microsoft®, made possible the assembly of a 5,760 Scalar Processor cluster at the University of Wisconsin-Madison.

Collaborators, Institutions

US Army TARDEC
Jet Propulsion Lab
FunctionBay, S. Korea
Caterpillar

Contributors, Individuals

Alessandro Tasora
Mihai Anitescu
Hammad Mazhar
Toby Heyn
Philipp Hahn
Arman Pazouki
Andrew Seidl