**BE-Simulation Framework: Looking Under the Hood**

Nalini Kumar, Ph.D. Candidate
Center for Compressible Multiphase Turbulence (CCMT)
ECE Department, University of Florida

---

**Co-Design Using Behavioral Emulation**

**HW/SW co-design**
- Algorithmic & architectural design-space exploration
- Coarse-grained BE simulation
  - Balance of simulation speed & accuracy for rapid design-space evaluation
**BE Framework Automation**

- Designed a framework to simplify and automate benchmarking
  - Job scripts can be generated for different machine/app combinations
  - Easy to extend benchmarking to new applications & systems
- Leveraging similar framework for CMT-nek batch runs as well

**Application to AppBEO Conversion**

Original Application  Call Graph

Instrumentation & Profiling

Original Application  Call Graph

Computation Block
Communication Block
Computation and Communication Block
**BE-SST Simulator**

- **SW Platform**: BE-SST created by extending Sandia’s SST
- **HW Platform**: Novo-G FPGA-based cluster

---

**BE-SST v1.0 Features**

- **System Configuration**
  - Component("vulcan. core", "Core", "BGQ-Core")
  - Component("vulcan. node", "none")
  - Component("vulcan. node. network", "eth")
  - Offspring("system", (Torus, "vulcan. node", "vulcan. node. network", [2, 2, 2, 2]))
  - Root("system")

- **Operations Configuration**
  - Operation("Core-name", "dr", "compute-dr.csv", Loiter(usage, =", 0.0),
    Modify(usage, 1.0),
    Dawdle(AnyOutput()),
    Modify(usage, 0.0))

- **AppBEO (.adl file)**
  - obtain(mpi.commRk)
  - comm(send, 16384, mpi.commRk - 1, 0)
  - call(cpu, wait, 16384, mpi.commRk + 1, 0)
  - call(cpu, fft, 512, 128)

- **Python Interface for Parameter Generation**
- **BE SST Component & Link Instantiation – allows for custom definitions**
- **Distribution of components and data structures**
- **BE Component**
- **Discrete time and discrete event simulation**
- **Clock and event queue management**

---
**BE-SST model enhancements**

- Communication model enhancements:
  - Dynamic generation of network routes provide better performance scaling and parallel performance
  - Overhead modeling at both sender/receiver endpoints

- Interpolation API allows easy switching between different interpolation methods for better accuracy

- Improved scalability and robustness of software product:
  - Reduced storage by deleting handled events from queues
  - Switched from static routes to runtime routing

**Interpolation Interface**

- Interpolation interface allows easy switching between different interpolation methods for better accuracy
  - Available schemes: linear, polynomial, lagrangian, kriging

1. Define default in operation configuration file

```python
Operation("$\text{appConv}$", "compareConv", "vulcan-convert-conv.csv", "\text{linear}\" "\text{elementsSize}\",
  Loader{usage", """, 0.0},
  Modify{usage", 1.0 },
  DelayOut{preps}(0, 0),
  DelayOut{usage", 0.0 })
```

2. Overwrite with the AppBEO instructions

```python
    call($\text{appConv}$, $\text{prepareConv}$, linear, $\text{appElements}$, $\text{appElementsPyRoman}\;

    call($\text{appConv}$, $\text{prepareConv}$, linear, $\text{appElements}$, $\text{appElementsPyRoman}\;
```
Memory Footprint of BE-SST

- BE simulations of different size systems with 3d-mesh topology running CMT-bone-BE
- Observations:
  - Memory used per BEO decreases because each core of HiPerGator is simulating more than one BEO
  - Memory usage has considerably reduced from previous version of BE-SST

BE-SST running on 64 cores of HiPerGator@UF

Results – Parallel Performance of BE-SST

- BE-SST demonstrates good performance scalability
  - Simulation of System A with 131,072 cores executing CMT-bone-BE
  - 385,025 network links in 3D Mesh topology [64, 64, 32]
  - Peak speedup of 33.7x was achieved on 128 MPI ranks
Parallel Performance of BE-SST

- BE simulations of 3d-mesh system running CMT-bone-BE
  - Simulations of machines with upto a million cores
  - With dynamic routing
- System config build time is not a bottleneck for large simulations

**Graph:**

- BE-SST running on 64 cores of HiPerGator@UF

**Axes:**

- X-axis: Simulated system size (in cores)
- Y-axis: Time (s)

**Legend:**

- Event simulation time (s)
- System config build time (s)
- Total simulation time - 1 timestep (s)

*Do you have any questions?*
Energy Benchmarking Infrastructure

**Tangible Result**
- Processor Granularity
  - Energy/Power Consumption Data

**Tool for Data Visualization**
- Visualization using Vampir
  - Allows easy visualizing of trace data generated by Score-P

**Output Data**
- Calibration Data Generation

**Auto Instrumentation**
- Score-P
  - Is open source
  - Generates profiling and tracing data
  - Is portable across HPC systems
  - Is scalable to large, HPC code

**API Wrapper**
- X86Energy Plugin
  - Libmsr
    - Intel : RAPL

**Open Source Plugins**
- Are configurable to vary overhead and API settings
- Allow extensions to a variety of hardware counters

---

Energy Benchmarking Infrastructure

**Tangible Result**
- Processor Granularity
  - Energy/Power Consumption Data
- Processor Granularity
  - Thermal Sensor Data
- Node Granularity
  - Energy/Power Consumption Data

**Tool for Data Visualization**
- Visualization using Vampir

**Output Data**
- Calibration Data Generation

**Auto Instrumentation**
- Score-P

**API Wrapper**
- X86Energy Plugin
  - Libmsr
    - Intel : RAPL
- Libsensors Plugin
  - Libsensors
    - Thermal Sensors
- PowerInsight Plugin
  - API
    - PowerInsight Board

**CCMT**
Validation of BE simulation on Vulcan

- Design of experiment
  - Element size (ES) = 5, 9, 13, 17, 21
  - Elements per processor (EPP) = 8, 32, 64, 128, 256
  - Number of processors (NP) = 16, 256, 2048, 16384

- Calibration data obtained through in situ benchmarking

- Observation: % error in emulation varies between -3% to 5%

In situ vs. Micro-kernel Benchmarking

- Micro-kernel benchmarking for calibration to support notional exploration
  - E.g., no source available

- Simulation of 5d torus system with 2048 cores
  - Error is comparable with in situ benchmarking methods for this initial study

![Graph showing validation of BE simulations on Vulcan with error in emulation plotted against element size and MPI ranks.](image)

![Graph comparing in situ vs. micro-kernel benchmarking with average execution time vs. element size.](image)
BE Simulations of CMT-bone-BE on Vulcan

- Average % error between CMT-bone-BE simulation and execution time is 4% and max. error is 9%

Particle Laden Simulations in CMT-nek

David Zwick
PhD student
Motivation & Setup

- While not fully resolved, Eulerian-Lagrangian simulations can simulate a realistic number (billions) of particles
- ASU experiment has $O(10^9)$ particles

One-Way Coupled Results (ASU)

Largest one-way simulations CMT-nek:

<table>
<thead>
<tr>
<th>Year</th>
<th>Ranks</th>
<th>Elements</th>
<th>Grid pts.</th>
<th>Particles</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>8k</td>
<td>60k</td>
<td>20 million</td>
<td>50 million</td>
</tr>
<tr>
<td>4</td>
<td>131k</td>
<td>131k</td>
<td>250 million</td>
<td>20 billion</td>
</tr>
<tr>
<td>4</td>
<td>131k</td>
<td>524k</td>
<td>905 million</td>
<td>905 million</td>
</tr>
</tbody>
</table>
Collaboration with BE Team

- Full spectral interpolation is costly (normal barycentric algorithm)
- Reduced barycentric interpolation is cheaper
  - 5x speedup on average over full barycentric approach
  - 8x speedup for higher polynomial orders

<table>
<thead>
<tr>
<th>Ranks</th>
<th>Elements/rank</th>
<th>Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>4096</td>
<td>64</td>
<td>15</td>
</tr>
</tbody>
</table>

Ranks Elements/rank Particles/grid pt.

<table>
<thead>
<tr>
<th>Ranks</th>
<th>Elements/rank</th>
<th>Particles/grid pt.</th>
</tr>
</thead>
<tbody>
<tr>
<td>4096</td>
<td>64</td>
<td>1</td>
</tr>
</tbody>
</table>

*Times are total time per time step

Governing Equations

**Fluid Momentum**

\[
\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{u} = \mathbf{f}
\]

**Particle Momentum**

\[
M_P \frac{d\mathbf{V}}{dt} = \mathbf{F}_{hyd}
\]

**Particle Forces**

\[
\mathbf{F}_{hyd} = \mathbf{F}_{qs} + \mathbf{F}_{un} + \mathbf{F}_{iu}
\]

\[
\mathbf{F}_{qs} = \frac{Re_P M_P}{24 \tau_p} (\mathbf{u} - \mathbf{V}) C_D
\]

\[
\mathbf{F}_{un} = M_P \frac{D\mathbf{u}}{Dt}
\]

\[
\mathbf{F}_{iu} = \frac{1}{2} V_p \left( \rho_g \frac{D\mathbf{u}}{Dt} - \frac{d(\rho_g \mathbf{V})}{dt} \right) C_M
\]
Computational Hurdles in Coupling

Each particle’s influence extends beyond its element/rank

Back coupling with gas for two-way coupling

Particle-particle interaction

Nearest-neighbor search

MPI Rank 0  MPI Rank 1

Search  Search  Search

MPI Rank 2  MPI Rank 3

Ghost Particle Algorithm

Idea:
- If a particle is near a MPI rank edge, it will create a copy of itself called a ghost particle
- Sending perspective rather than receiving perspective

GP steps:
1. Create GP
2. Send GP
Scaling of Ghost Particle Algorithm

Scalability of algorithm:
- We consider a test case of an expansion wave over a bed of particles
- Particles are randomly distributed and advected by the wave

<table>
<thead>
<tr>
<th>MPI Ranks</th>
<th>Elements</th>
<th>Polynomial</th>
<th>Grid points</th>
<th>Particles</th>
<th>Time steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>8k–65k</td>
<td>432,000</td>
<td>5</td>
<td>54 million</td>
<td>27 million</td>
<td>2000</td>
</tr>
</tbody>
</table>

Strong Scaling

\[ \delta = 0.5(L) \]

*Times are total time per time step per MPI rank

Back Coupling with Gas

- Accomplished through the use of a filtering kernel \( g_m \) for mollification
- Gaussian kernel is used for \( g_m \)
- For example, to transfer particle property \( A \) to the grid as \( a \):

\[
a(x) = \sum_{i=1}^{N_p} A(X^{(i)}) g_m(|x - X^{(i)}|) V_p^{(i)}
\]

<table>
<thead>
<tr>
<th>( A )</th>
<th>( a )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>particle volume fraction</td>
</tr>
<tr>
<td>( F_{av} / V_p^{(i)} )</td>
<td>particle hydrodynamic force</td>
</tr>
</tbody>
</table>

Fluid Momentum

\[
\frac{\partial u}{\partial t} + \nabla \cdot h = f
\]

For an element in weak form

\[
\int_{V_\Omega} v_k f_k dV_\Omega \rightarrow v_k^T B f_k
\]
**Momentum Conservation**

Particles are placed at rest in a uniform flow and subjected to Stokes drag
- Particles speed up and fluid slows down
- Total momentum remains constant

<table>
<thead>
<tr>
<th>Domain</th>
<th>Elements</th>
<th>Polynomial</th>
<th>Particles</th>
<th>MPI Ranks</th>
</tr>
</thead>
<tbody>
<tr>
<td>x,y,z = [-1,1]</td>
<td>64</td>
<td>11</td>
<td>200,000</td>
<td>64</td>
</tr>
</tbody>
</table>

**Application to ASU Experiment**

Initial gas properties
- Inertial tail
- Rarefaction wave
- Contact discontinuity
- Shock wave

**Gas Initial Conditions**

- Pressure (Pa)
- Temperature (K)
- Velocity (m/s)

Particles & Gas → Gas Only
Simulation Setup

Elements | Grid points | MPI Ranks | SPL
---|---|---|---
8,192 | 1,000,000 | 8,192 | 76

Simulation Results

Fluid Velocity [m/s] | Particle Volume Fraction
**Simulation Results**

**Strong Scaling**

- Time [μs]: 850.0, 876.3, 884.8, 892.4
- Bed Height
- **Void Formation**

**Future work**

- Larger simulations on Mira
- Improved force modeling
  - PIP
  - Volume filtering correction
  - Effects on validation metrics
- Load balancing
  - Conference paper in preparation with Computer Science group
- UQ and V&V
Do you have any questions?

Filtering Examples

- Using ghost particle approach, a mollification kernel is used to spread properties to the Eulerian grid

Force from a single particle

Volume from a cloud of particles
A short time after detonation, there is pure explosive product (red) and pure air (green)

At this point, use real gas equations in red cells and ideal gas equations in green cells

At later time, there are still red and green cells but also a mixing layer shown by the orange cells

Here, both equations of state must be used
### Standard Approaches for Mixed Cells

<table>
<thead>
<tr>
<th>Method of Tackling</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
</table>
| Mass fraction weighted averaging   | • Algebraic equations  
• Small data storage                                                      | • Does not place species equilibrium requirement                      |
| Tabulated data                     | • Interpolation is only operation  
• Can place species equilibrium requirement                                   | • Must store the tabulated data  
• Interpolation error                                                   |
| Polynomial regressions/curve fits  | • Algebraic equations  
• Small data storage  
• Can place species equilibrium requirement                                 | • Curve fit errors                                                      |
| Iterative schemes                  | • Small data storage  
• Can place species equilibrium requirement                                  | • Computationally expensive                                              |

- Rocflu uses a Broyden’s method iterative solver to enforce pressure and temperature equilibrium between air and the explosive products.

### Looking at the Big Picture

- What is the iterative scheme really doing?

  **Inputs:**
  \[ \rho_m e_m Y_{exp} \]

  **Outputs:**
  1) Common pressure/temperature for both EoS
  2) Internal energy attributed to the air and the products
  3) Density attributed to the air and the products

- If we compute the common pressures and temperatures outside of the code, a curve fit can be applied.
Models Implemented in Rocflu

The Jones-Wilkins-Lee (JWL) equations of state are used to predict the pressures of high energy substances and are:

\[ P_{JWL}(\rho, e) = A\left(1 - \frac{\omega}{\rho R_1}\right)e^{-\frac{\rho}{\rho R_1}} + B\left(1 - \frac{\rho}{\rho R_2}\right)e^{-\frac{\rho}{\rho R_2}} + \omega pe \]
\[ T_{JWL}(P, \rho) = \left(\frac{1}{\rho c_w}\right)(P - A\rho R_1e^{-\frac{\rho}{\rho R_1}} - B\rho R_2e^{-\frac{\rho}{\rho R_2}}) \]

where \( V = \frac{\rho e}{\rho} \) and \( \rho, A, B, C, R_1, R_2 \) and \( \omega \) are parameters for the substance.

<table>
<thead>
<tr>
<th>One Equation Model</th>
<th>Iterative Method</th>
<th>Multi-Fidelity Surrogate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advantages</td>
<td>Speed</td>
<td>Accuracy and Speed</td>
</tr>
<tr>
<td></td>
<td>- Algebraic equation</td>
<td>Problem independent</td>
</tr>
<tr>
<td>Disadvantages</td>
<td>- Uncertainty and error</td>
<td>May be slow to converge</td>
</tr>
<tr>
<td></td>
<td>- JWL + ideal gas case only</td>
<td>Computationally expensive</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Uncertainty and Equation of State specific (problem specific)</td>
</tr>
</tbody>
</table>

Surrogate Model - Development

- Models for mixed cell pressure and temperature generated by Kriging with 200 sampling points by UQ team
- Domain Space: \( \rho_m \rightarrow (1, 1.77) \)
  \( e_m \rightarrow (1.25e5, 1.5e7) \)
  \( Y_{exp} \rightarrow (0, 1) \)
- Average relative errors:
  - Pressure: 1.298%
  - Temperature: 0.132%
Flowchart – Iterative vs. Surrogate

IF Y < 0.001
Loop over all cells in domain
IF Y > 0.999
ELSE
Ideal Gas Equations
JWL Equations

Broyden’s Method Iteration
Inputs: $\rho_m$, $e_m$, $Y$

Calculate Initial guesses, values for $f_1$ to $f_4$ and
Jacobian matrix (4 exponential operations)

Invert Jacobian matrix

Update Solution (1 matrix multiply)

Update Inverse Jacobian
(2 matrix multiplies, 4 exponential operations)

IF converged
Break, Calculate
P and T

ELSE
Surrogate Model
Inputs: $\rho_m$, $e_m$, $Y$

Read in fit parameters from
data files

Call model operations twice
(4 matrix multiplies,
$n_{pts}$ exponential operations)

P and T are outputs of the
model operations

Surrogate Model Results – Computation Time

<table>
<thead>
<tr>
<th>Run</th>
<th>seconds/Time step (Iterative)</th>
<th>seconds/Time Step (Surrogate)</th>
<th>Ratio</th>
<th>Previous Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.4636</td>
<td>0.1739</td>
<td>2.6662</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>0.4314</td>
<td>0.1778</td>
<td>2.4262</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>0.4766</td>
<td>0.1754</td>
<td>2.7178</td>
<td>-</td>
</tr>
<tr>
<td>Avg.</td>
<td>0.4572</td>
<td>0.1757</td>
<td>2.6024</td>
<td>2.6313</td>
</tr>
</tbody>
</table>

Average time per iteration in computing 1000 time steps in Rocflu
Results averaged over three runs
Speed up ratio for 200 point model is about 2.60

Preliminary results from study on differing numbers of sampling points
Adding more points increases computation time
**Surrogate Model Results – Simulation**

- Maximum error in blast wave location for the 200 point model is 0.83%
- Maximum error in peak pressure for 200 point model is 8.51%
- Increasing number of sampling points reduces errors obtained using the surrogate

**Simulation Description**
- 2D, Quarter-Cylinder grid
- Outer radius = 0.3 m
- 400,000 cells
- Gas-Only

---

**Future Work**

**Effect of Sampling Points**
- Study of the effect that altering the number of sampling points has to the model error
- Goal is to verify that 200 sampling points hits ideal midpoint of accuracy and computing time

**Multiple Species**
- Suppose an experiment dealt with a detonation process that contained \( n \) species
- If using Rocflu's iterative solver, this means going from a \( 4 \times 4 \) system of equations to an \( 2n \times 2n \) system
- As long as the same number of test points are used to create a new surrogate, its computation time would remain roughly the same

**Implementation with Multiphase flows**
- Implement the model into simulations of flows with particles.
- Goal is to ensure the surrogate and the iterative scheme have minimal differences in particle trajectories, allowing for confident use in simulating experiments
Do you have any questions?

Microscale Simulations and Modeling

Yash Mehta
Outline

- Fully resolved simulations of shock interaction with randomly distributed bed of particles:
  - Varying volume fraction
  - Varying shock Mach number
  - 1-D Riemann model for interpreting/analyzing results from numerical simulations
  - Pair-wise Interaction Extended Point Particle (PIEP) model

Fully Resolved Simulations

- We have performed multiple fully resolved simulations of shock interaction with particles
- Particle arrangement, particle volume fraction and, shock Mach number was varied
- Different physical mechanisms occurring during shock particle interaction were identified
We have performed multiple simulations by varying the volume fraction and the shock Mach number: \( M_s = 1.22, 1.66 \) and \( 3.00 \); \( \varphi = 1.25, 10, 15, 20 \) and \( 25\% \). 
- Number of particles in the computational domain varied from 200 to 500 
- Effect of viscosity and particle motion was neglected in these simulations

Flow Properties

Mach number contour plot for \( \varphi = 10\% \); \( M_s = 3.0 \)

Pressure contour plot for \( \varphi = 25\% \); \( M_s = 3.0 \)

Normalized stream-wise averaged density for \( \varphi = 10\% \); \( M_s = 3.0 \)

Normalized stream-wise averaged velocity for \( \varphi = 25\% \); \( M_s = 3.0 \)
Forces on particles; \( M_S = 3.0 \)

- \( \varphi = 10\% \)
- \( \varphi = 15\% \)
- \( \varphi = 20\% \)
- \( \varphi = 25\% \)

There is a high variability for the peak drag and a clear downward trend.

1-D Riemann Model

- Shock interaction with bed of particles can be modeled as shock interaction with an area change in a 1-D context.
- This is a standard Riemann problem with an additional parameter - area change.

\[
\frac{\partial \psi}{\partial t} = 0
\]

We obtain a "steady" state solution for different volume fractions

Reflected shock travels upstream of the area change, there is a resonant expansion fan at the location of area change, contact discontinuity and, transmitted shock travel downstream

For a given shock Mach number and volume fraction there exists a unique solution

Flow properties obtained from 1-D model for $M_s = 3.0; \, t/\tau = 12$

Averaged flow properties obtained from numerical simulations for $M_s = 3.0; \, t/\tau = 12$
1-D Riemann Model Vs Numerical Simulation

<table>
<thead>
<tr>
<th>$\phi_1$</th>
<th>$M_r$</th>
<th>$M_\ell$</th>
<th>$P_s/P_\ell$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>1.41</td>
<td>3.00</td>
<td>1.000</td>
</tr>
<tr>
<td>15%</td>
<td>1.47</td>
<td>3.03</td>
<td>1.020</td>
</tr>
<tr>
<td>20%</td>
<td>1.52</td>
<td>3.05</td>
<td>1.035</td>
</tr>
<tr>
<td>25%</td>
<td>1.56</td>
<td>3.07</td>
<td>1.047</td>
</tr>
</tbody>
</table>

Flow properties obtained from 1-D model for $M_s = 3.0; \; t/\tau = 12$

<table>
<thead>
<tr>
<th>$\phi_1$</th>
<th>$M_r$</th>
<th>$M_\ell$</th>
<th>$P_s/P_\ell$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>1.41</td>
<td>2.75</td>
<td>0.874</td>
</tr>
<tr>
<td>15%</td>
<td>1.59</td>
<td>2.64</td>
<td>0.799</td>
</tr>
<tr>
<td>20%</td>
<td>1.67</td>
<td>2.43</td>
<td>0.718</td>
</tr>
<tr>
<td>25%</td>
<td>1.72</td>
<td>2.34</td>
<td>0.678</td>
</tr>
</tbody>
</table>

Averaged flow properties obtained from numerical simulations for $M_s = 3.0; \; t/\tau = 12$

Pairwise Interaction Extended Point Particle Model (PIEP)

- Force on a single sphere is given by the Generalized Faxen’s Theorem ($F_{\text{flow}}$).

- Force due to presence of other particles ($F_{\text{diff}}$):
  \[ F_{\text{diff}} = \sum_{j=1}^{N-1} F_{j-i} \]
**PIEP – Incompressible Flow**

DNS | DEM w/ PIEP | DNS
---|---|---

**PIEP – Compressible Flow**

- Shock interaction with transverse array of particles

\[ F_R = F_{\text{shock}} + 4 \times F_{1-R} + 4 \times F_{2-R} \]

- Drag force on the center particle
Future Work

- Post processing the simulation data
- Viscous simulations for shock interaction with particles
- Developing PIEP model for compressible flows

Do you have any questions?
Predicting Execution Times of CMT-nek using Multi-fidelity Surrogate

Yiming Zhang

Accomplishments

- Coordinated the multi-fidelity error reduction for CMT codes at large-scale runs
  - Designed the experiments for comparing CMT-nek (Jason Hackl), CMT-bone (Tania Banerjee), CMT-bone BE (Aravind Neelakantan) and BE emulation (Aravind Neelakantan)

- Predicted execution times of CMT-nek using multi-fidelity surrogate
  - Proposed a multi-fidelity surrogate to fit a few CMT-nek runs together with BE emulations. The multi-fidelity prediction had 4% root-mean-square difference (RMSD) with CMT-nek
  - Submitted to IEEE Cluster 2017 titled as “Using Multi-fidelity Surrogates of Skeleton Apps for Predicting Application Performance”
Design of Experiments at Large-Scale Runs

- Execution times of a typical no-particle test (on Vulcan) using up to 130k processors, 34 million \((131072 \times 256)\) elements and 311 billion \((21^3 \times 131072 \times 256)\) computational grid points
- 125 \((5^3)\) runs for BE emulation & CMT-bone BE, with subsets for CMT-nek & CMT-bone

CMT codes vs. BE emulation

- Similar trend, but large difference
  - Effective emulation (low-fidelity model) could predict the overall trend well of CMT-nek runs (high-fidelity model)
  - Emulations of CMT-bone BE simplify the calculations, which were much smaller than the execution times of CMT-nek
  - Using multi-fidelity surrogate translates emulations (low-fidelity model) to predict CMT-nek runs (high-fidelity model)
Predictions using Multi-fidelity Surrogate

Translate BE emulation against a few CMT-nek runs by polynomial function

Form of translation function
Selected a popular form with a scale factor and a discrepancy surrogate

\[ \hat{f}_{\text{nek}}(x) = \rho \hat{f}_{\text{BE}}(x) + \delta(x) \]

Schemes to determine surrogate parameters
Developed a multi-fidelity surrogate for improved robustness and accuracy
- Bayesian vs. Deterministic
- Spatial distribution vs. Residual error
- Sequential vs. Simultaneous

Test Plan To Evaluate Multi-fidelity Predictions

- Selected 10 runs (out of 22) of CMT-nek as validation runs to examine the accuracy of emulation
- The overall difference (root-mean-square) between original BE emulation and CMT-nek was 74% at the 10 validation points
- Adopted 2 to 12 remaining runs of CMT-nek as samples to translate BE emulations using the multi-fidelity surrogate
- For each number of samples, repeated random selection 20 times to account for the effect of sampling plan
Multi-fidelity Predictions of CMT-nek Runs

- Overall difference (root-mean-square) is less than 10% with 6 or more CMT-nek samples for correction
- Max error is less than 20% (at the 10 validation runs) with 9 or more CMT-nek samples for correction
- The proposed multi-fidelity surrogate was more accurate than other approaches

![Graph showing multi-fidelity predictions](image)

Ongoing Work

- Investigating the capability of corrected BE emulation to extrapolate the execution time of CMT-nek runs towards exascale computing
- Sampling strategy for extrapolating the execution time of CMT-nek runs while incorporating varying cost of runs
Do you have any questions?

Deeper look into Collapsed Pipeline Approach

Carlo Pascoe
Center for Compressible Multiphase Turbulence (CCMT)
NSF Center for High-Performance Reconfigurable Computing (CHREC)
ECE Department, University of Florida
Simulations of CMT-bone-BE on Vulcan

Simulation Method | Num. of MPI Ranks | Num. of Timesteps | Num. of Events | % Logic Utilization | FPGA Clk Frequency | Latency to First Output | Mega Sims/Sec
--- | --- | --- | --- | --- | --- | --- | ---
BE-SST | 128 | 1 | 5,952 | NA | NA | NA | NA | 0.00000152
FEP | 128 | 1 | 5,952 | 66 % | 280 – 330 MHz | 56 | 280 – 330
CP | 128 | 1 | 5,952 | 2 % | 315 – 355 MHz | 458 | 1.68 – 1.99

- BE software and BE FPGA simulations produce similar results.
- Fully-expanded approach provides greatest performance at cost of more resources.
- Collapsed approach allows for greatly reduced resource utilization and increased scalability at cost of slightly reduced performance.

CCMT

Outline

- Motivation
  - Fully-expanded v Collapsed Pipeline Approach
  - Simple Motivating Example
- Collapsed Pipeline Single-FPGA Performance/Scalability
- Collapsed Pipeline Multi-FPGA Performance/Scalability
- Moving Forward
  - Validation of multi-FPGA predictions
  - Increased performance with Stratix 10
  - Partially-collapsed pipeline approach
**Fully-Expanded v Collapsed Approach**

**Fully-Expanded Pipeline**

- **Advantages:**
  - Superior performance in terms of simulation throughput and latency

- **Limitations:**
  - Resources scale linearly with both MPI Ranks and number of timesteps
  - Scaling across multiple FPGAs expected to be ineffective when considering exascale simulations

**Collapsed Pipeline**

- **Advantages:**
  - Resources scale linearly with timesteps, but sublinearly with MPI Ranks
  - Better scaling on single and multiple FPGAs

- **Limitations:**
  - Lower simulation throughput and longer initial latency, but still more than sufficient for rapid design-space exploration

---

**Simple Motivating Example**

**Fully-expanded Approach**

**Collapsed Approach**

---

Page 38 of 54
As number of ranks increase:

- additional logic per rank approaches zero
- simulation throughput reduced by a factor of “ranks”
- event throughput remains proportional to instantiated event hardware

Pipelines scale linearly with length of simulation, however blockram eventually becomes limiting resource

- e.g., only fit 2 TS for 1 million ranks on Stratix V
- Motivation to explore partially-collapsed pipeline approach

Increased performance expected with Stratix 10

- Collapsed hardware specifically designed to exploit new Stratix 10 architecture

### Collapsed Pipeline Single-FPGA Performance/Scalability

<table>
<thead>
<tr>
<th>Ranks</th>
<th>TS</th>
<th>% LU</th>
<th>Num. of Events</th>
<th>Lat to First Out (cycles)</th>
<th>Mega Sims/second*</th>
<th>Giga Events/second*</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>32</td>
<td>44</td>
<td>43,008</td>
<td>7,532</td>
<td>10.5</td>
<td>450</td>
</tr>
<tr>
<td>64</td>
<td>32</td>
<td>46</td>
<td>92,160</td>
<td>10,252</td>
<td>5.23</td>
<td>482</td>
</tr>
<tr>
<td>128</td>
<td>32</td>
<td>47</td>
<td>190,464</td>
<td>10,316</td>
<td>2.62</td>
<td>498</td>
</tr>
<tr>
<td>256</td>
<td>32</td>
<td>67</td>
<td>393,216</td>
<td>13,516</td>
<td>1.31</td>
<td>515</td>
</tr>
<tr>
<td>512</td>
<td>32</td>
<td>84</td>
<td>811,008</td>
<td>20,684</td>
<td>0.65</td>
<td>531</td>
</tr>
<tr>
<td>1K</td>
<td>32</td>
<td>84</td>
<td>1,664,592</td>
<td>21,196</td>
<td>0.33</td>
<td>539</td>
</tr>
<tr>
<td>32K</td>
<td>32</td>
<td>84</td>
<td>55,443,456</td>
<td>241,868</td>
<td>1.02E-2</td>
<td>567</td>
</tr>
<tr>
<td>128K</td>
<td>16</td>
<td>46</td>
<td>111,673,344</td>
<td>333,932</td>
<td>2.56E-3</td>
<td>285</td>
</tr>
<tr>
<td>1M</td>
<td>2</td>
<td>5</td>
<td>112,459,776</td>
<td>1,148,056</td>
<td>3.19E-4</td>
<td>36</td>
</tr>
</tbody>
</table>

Ranks TS % LU Num. of Events Lat to First Out (cycles)* Mega Sims/second* Giga Events/second*

| 32    | 32 | 44   | 43,008         | 7,532                    | 10.5              | 450                 |
| 64    | 32 | 46   | 92,160         | 10,252                   | 5.23              | 482                 |
| 128   | 32 | 47   | 190,464        | 10,316                   | 2.62              | 498                 |
| 256   | 32 | 67   | 393,216        | 13,516                   | 1.31              | 515                 |
| 512   | 32 | 84   | 811,008        | 20,684                   | 0.65              | 531                 |
| 1K    | 32 | 84   | 1,664,592      | 21,196                   | 0.33              | 539                 |
| 32K   | 32 | 84   | 55,443,456     | 241,868                  | 1.02E-2           | 567                 |
| 128K  | 16 | 46   | 111,673,344    | 333,932                  | 2.56E-3           | 285                 |
| 1M    | 2  | 5    | 112,459,776    | 1,148,056                | 3.19E-4           | 36                  |

### Collapsed Pipeline Multi-FPGA Performance/Scalability

*Predicted collapsed performance of CMT-Bone-BE with varied MPI ranks and simulation timesteps on a single Stratix V S5GSMD8K1F40C2 @ 335MHz

Pipelines scale linearly with the length of simulation, but as a single, unidirectional pipe

- Partitioned easily/predictably across any number of connected FPGAs

Given a desired # of simulated ranks, timesteps, granularity (events per timestep), we can predict performance & # of FPGAs

Main point: FPGAs achieve similar scale as BE-SST, but orders of magnitude faster

- 1M ranks, 300+ of sims/second

*See poster from Carlo Pascoe

**Novo-GF**

- 64 GiDEL ProceV (Stratix V)
- 4x4x2 3D torus or 5D hypercube
- 6 Rx-Tx links per FPGA
- Measured 32 Gbps per link
- 150 ns latency across links

- Require 64bits/cycle between FPGAs
- 335MHz, 21.4 Gbps, 51 additional lat
- 345MHz, 28.8 Gbps, 68 additional lat
- 500 MHz before link BW saturation

* See poster from Carlo Pascoe
Moving Forward

- Validation of multi-FPGA predictions
- Increased performance with Stratix 10
- Partially-collapsed pipeline approach

Do you have any questions?

Simulation-Motivated Forensic Uncertainty Quantification of Eglin Microscale Experiments

Kyle Hughes
Graduate Student

April 21, 2017
**Forensic UQ Perspective**

- Significant time lag is present between the microscale experiments and the simulations
  - Not unusual in validation exercises
- As time grows, it becomes steadily more difficult to decrease the epistemic (or lack of knowledge) uncertainty of the experiments
  - Experimental setup or samples are often unavailable for analysis and must be recreated
- Uncertainty may be recognized and reduced through **forensic uncertainty quantification** (UQ) by an independent investigator
- Forensic UQ of the past experiments can be analogous to performing forensics at a crime scene
  - Document all the experimental details (crime scene)
  - Clarify details with experimentalists (witnesses)
  - Quantify uncertain inputs and prediction metrics (forensics lab)
  - Compare to simulation to identify inconsistencies (culprits)
- **Goal:** Apply forensic UQ to the Eglin microscale explosive experiments

**Validation Hierarchy**

- CCMT employs a physics-based validation hierarchy
- Eglin conducts experiments at all three scales of interest to CCMT
- **Microscale** ($O(1-10^4)$ particles)
  - Validation of microscale drag models
  - Explosive tests conducted October 2014 and February 2015
- **Mesoscale** ($O(10^6)$ particles)
  - Examination of instabilities, volume fraction effects
  - Explosive tests conducted November 2015
  - Gas gun tests in planning stages
- **Macroscale** ($O(10^9)$ particles)
  - Validation of full simulation
  - Explosive tests (AFRL blastpad) in planning stages
## Microscale Experiments

**Pl: Don Littrell**

Side view photograph showing the concave pressure probe array.

Summary of test shots performed in October 2014 and February 2015.

<table>
<thead>
<tr>
<th>Test #</th>
<th>Particle(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct14-1</td>
<td>Single (2 mm dia) Tungsten sphere</td>
</tr>
<tr>
<td>Oct14-2</td>
<td>Ring of 7 (2 mm dia) Tungsten spheres</td>
</tr>
<tr>
<td>Oct14-3</td>
<td>Grid of 16 (2 mm dia) Tungsten spheres</td>
</tr>
<tr>
<td>Feb15-1</td>
<td>Single (2mm dia) Tungsten sphere</td>
</tr>
<tr>
<td>Feb15-2</td>
<td>Single (2mm dia) Tungsten sphere</td>
</tr>
<tr>
<td>Feb15-3</td>
<td>Diamond of 4 (2 mm dia) Tungsten spheres</td>
</tr>
</tbody>
</table>

Sources: Myles Delcambre, Internal CCMT Fall 2014 presentation  
Black, Littrell, and Delcambre, Eglin internal written report, 3/7/2015

## Validation and UQ Framework - Microscale

Experiments

- Measured Input
- Measurement Uncertainty

Measured Prediction Metrics

- Sampling Uncertainty
- Measurement Uncertainty
- Systematic Uncertainty

Explosive length
Explosive diameter
Explosive density
Casing outer diameter
Casing inner diameter
Particle diameter
Particle density
Particle initial location(s)
Ambient pressure
Ambient temperature

Particle trajectory
Contact line location

Limited # of x-ray exposures
Limited framerates
User error
Instrument resolution
Apparatus misalignment

\( T_0 \)
Measured Input: Particle Diameter

- Particle diameter was not measured a priori – initial uncertainty was assumed +/- 10% nominal diameter
- Statistical study of tungsten particles was conducted after the experiments to quantify their uncertainty
- Eglin provided 52 tungsten spheres that were randomly measured by two users

![Histograms of particle diameter measurement](Source: Don Littrell, 2/26/2015)

- Final data shows uncertainty has been **reduced** to 2-3% of nominal diameter compared to prior belief of 10%
- Smaller range of diameters will also decrease number of runs necessary to construct the surrogate

Measured Input: Particle Density

- The sample provided by Eglin was additionally used to quantify the uncertainty in the particle density
- Manufacturer cites the particle density as 17 g/cm³
- All 52 spheres were weighed and then their volume repeatedly measured in a helium gas pycnometer
- Densities are significantly different than those cited by the manufacturer
- Possible causes for this discrepancy are still being investigated
- Energy dispersive x-ray spectrometry may identify the specific alloy

<table>
<thead>
<tr>
<th>Run</th>
<th>Volume [cm³]</th>
<th>Density [g/cm³]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2401</td>
<td>15.92</td>
</tr>
<tr>
<td>2</td>
<td>0.2416</td>
<td>15.82</td>
</tr>
<tr>
<td>3</td>
<td>0.2470</td>
<td>15.48</td>
</tr>
<tr>
<td>4</td>
<td>0.2546</td>
<td>15.02</td>
</tr>
<tr>
<td>5</td>
<td>0.2462</td>
<td>15.53</td>
</tr>
<tr>
<td>6</td>
<td>0.2457</td>
<td>15.56</td>
</tr>
<tr>
<td>7</td>
<td>0.2429</td>
<td>15.74</td>
</tr>
<tr>
<td>8</td>
<td>0.2428</td>
<td>15.75</td>
</tr>
<tr>
<td>9</td>
<td>0.2491</td>
<td>15.35</td>
</tr>
<tr>
<td>10</td>
<td>0.2489</td>
<td>15.36</td>
</tr>
<tr>
<td>11</td>
<td>0.2480</td>
<td>15.42</td>
</tr>
<tr>
<td>12</td>
<td>0.2460</td>
<td>15.54</td>
</tr>
</tbody>
</table>

Average 0.2461 15.54
Std Dev 0.00394 0.25
Microscale Simulations $T_0$

- Large assumptions currently made in the simulations:
  - High density quiescent gas is representative of the explosive after reaction
  - Requires time shift to align the experiment and simulation
- To align with experiments need to account for both time of detonator to act and the time for the explosive to burn:
  - Manufacturer reported detonator function time: $5.38 \pm 0.125 \mu s$
  - Assuming constant detonation velocity through the explosive: $4.31 \mu s$
  - Total delay: $\sim 9.7 \mu s$

![Cross Section View](image)

Timing Uncertainty

- 10 MHz X-ray o-scope data for Oct14-1
- 1 MHz X-ray o-scope data for Feb15-2

CCMT
**Metric: Single Particle Position**

- Horizontal error bars are determined from oscilloscope records
- Vertical error bars are determined based on spread in data at repeated times

![Multiple exposure x-ray of single particle](image1)

![Plot of single particle position from x-ray data](image2)

**Shock Time of Arrival**

- Initially expected negligible effect on the shock time of arrival based on the small number of particles present
- Shock time of arrivals show an increase in the mean as the number of particles is increased from 1 to multiples
- Evidence that the flow is strongly coupled with particles
- A possible explanation is to examine the volume fraction in the initial plane of the particles:

<table>
<thead>
<tr>
<th>Number of Particles</th>
<th>Volume Fraction [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.65</td>
</tr>
<tr>
<td>4</td>
<td>6.6</td>
</tr>
<tr>
<td>7</td>
<td>11.6</td>
</tr>
<tr>
<td>16</td>
<td>27</td>
</tr>
</tbody>
</table>

![Table of shock time of arrival](image3)
Conclusions and Future Work

Conclusions

- Uncertainty quantification effort for even this "simplified" problem requires extensive back and forth between UQ, simulation, and experimental teams (5+ months)
- Understanding of the simulation assumptions and inputs can help drive forensic investigation
  - Difficulty of determining T₀ resulted in significant investigation into oscilloscope records
- Forensic UQ found a surprise that neither experimentalist or simulationist expected: The small number of particles show significant coupling with the flow

Future Work

- Completion of the uncertainty quantification of remaining microscale uncertain inputs
  - The explosive parameters are especially critical but also challenging
- Construction of surrogates to propagate input uncertainties through simulation
- Comparison of the experimental prediction metrics with simulation results for validation of the drag force models for a single particle

Do you have any questions?
**Forensic UQ Work-Flow**

**Experiment**
- Identify Metrics
- Design of Experiment
- Data Collection
- Measurement of Uncertain Inputs
  - Prediction Metrics w/ Uncertainty

**Simulation**
- Identify Prediction Metrics
- Identify Uncertain Inputs
- Design of Simulation
- Verification
- Uncertainty Propagation
- Prediction Metrics w/ Uncertainty

- **2+ years separation!**
- Requires forensic investigation
- **Stop**
- Yes
- Satisfactory Validation?
- No

**Simulation Improvement**

---

**Out-of-Plane Motion: Witness Panel**

- Witness panel results provides some indication of out-of-plane movement of the particles by providing final position
- Single particle shots show very small angles of the particles
- Repeated particles show significant spread such that two of the particles did not impact the panel while the other two impacted fairly close to the shot line
- The small angles of the single shot results allows elimination of out-of-plane movement as a large possible source of uncertainty

---

**Angles obtained for February 2015 shots**

<table>
<thead>
<tr>
<th>Shot</th>
<th>( \theta_1 ) (deg)</th>
<th>( \theta_2 ) (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb15-1</td>
<td>-0.6494</td>
<td>-1.3718</td>
</tr>
<tr>
<td>Feb15-2</td>
<td>0.2392</td>
<td>0.7177</td>
</tr>
<tr>
<td>Feb15-3</td>
<td>2.8685</td>
<td>6.1961</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shot</th>
<th>( \theta_1 ) (deg)</th>
<th>( \theta_2 ) (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb15-4</td>
<td>-6.5337</td>
<td>-5.0452</td>
</tr>
</tbody>
</table>

---

**a)** Angle conventions used for the witness panel results

**b)** Raw witness panel results
Metric: Contact Line

- Obtained from the high speed Simacon images (maximum 16 frames)
- Contact line results are obtained by the selection of the furthest location from explosive casing
- Contact line is only visible in a couple of frames in October 2014 shots (excluded)
- Uncertainty quantification of the results is on-going

Metric: Average Particle Position

- Can be difficult to track each particle for the multiple particle shots
- An average of the particle packet is selected here to begin looking at the multiple particles
- Variance seems reduced by this choice of an average, very similar trends observed with respect to a single particle
Simulation Capability

- Clear discussion of simulation capabilities is necessary for design of a validation experiment and forensic UQ.
- Current simulation is missing a variety of physics:
  - For instance, it does not include effects from particle non-sphericity and deformability.
  - Experimentalists did an excellent job in minimizing this effect by using high-density, spherical tungsten particles.
- Experimental evidence for significant particle-fluid interaction was surprising for both experimentalists and simulationists.
- Challenging to implement 2-way coupling for such large particles.

Experimental Studies of Gas-Particle Mixtures Under Sudden Expansion at ASU

Heather Zunino
PhD Student
Dr. Ronald Adrian
Advisor
Problem Statement and Goals

- Experimental multi-phase studies involving compressible flow are complicated
  - Air and solid particles may move separately
  - Particles generate turbulence
- Need for a simple 1D flow experiment that can be used for early validation of the computational codes developed by the PSAAP center.
- Simpler physics involved than the PSAAP capstone experiment
- Reduce the scatter in current data (Chojnicki, et al.)
- Perform experiments on existing shock tube setup
- Determine improvement points and weaknesses
- Design an improved, simple 1D compressible multi-phase flow shock tube experiment
- Examine expansion fan, flow structures, turbulence, and instabilities
- Provide data for early-stage validation of computational codes developed by the PSAAP Center

Experiment Description

- 1 meter glass tube
  - Cylindrical footprint
  - Inner diameter: 2”
- Particle bed
- Diaphragm
  - Tape
- High-speed camera
- Measurements
  - Contact line velocity
  - Gas velocity
  - Particle volume concentration
  - Particle interface
- Parameters: particle size and pressure ratio
Experiment Description

Stages of Experiment
Cloud Evolution

Reflected Shock Wave
Interface Rise

2cm 5cm 5.4cm

Hardware Additions

- 8 pressure sensors
  - 4 locations
  - 90º or 180º apart
- Data Acquisition
  - 8 simultaneous channels
  - Over 150kHz each channel
- Synchronization
  - Camera
  - Pressure sensors
  - Laser light sheet
Summary

- Cloud evolution
  - Interframe appearance
  - Measure gas velocity escape
  - Gas escapes from edges of particle bed first
- Reflected shock wave
  - Use cloud droplets as gas tracer
  - Particles seem unaffected
- Interface rise
  - Different bed heights
  - Different initial volume fraction
- Hardware additions
  - Upgraded pressure sensors
  - Better temporal resolution
  - Synchronized data acquisition

Do you have any questions?