ADAPTIVE MULTISCALE PREDICTIONS AT EXASCALE

Assad A. Oberai
Professor
Associate Director
Scientific Computation Research Center (SCOREC)
Rensselaer Polytechnic Institute

Onkar Sahni
Assistant Professor
Scientific Computation Research Center (SCOREC)
Rensselaer Polytechnic Institute

Mark Shephard
Professor
Director
Scientific Computation Research Center (SCOREC)
Rensselaer Polytechnic Institute
MULTISCALE PROBLEMS IN MULTIPHASE FLOW

PHASTA (continuum) → LAMMPS (molecular) → LAMMPS (molecular) → PHASTA (continuum)

EXP – roughness (nano)
AFM image:
Δx, Δy ~ 5.2μm

EXP – film (micro)
Adsorbed film
Meniscus

EXP – bubbles (macro)
EXP – slug (macro)
EXP – annular (macro)

Liquid Flow
Bubbly
Coalesced
Slug
Annular flow

Bubble Nucleation
Moving Contact Line

Coalescence
Interfacial Instabilities

Dry-out
Thin/Adsorbed Film

Dry-out
A coupled continuum/molecular dynamics approach

**Concurrent Coupling [Robbins et al.]**
- Continuum and MD together
- MD and continuum simulations coupled over an overlap region
- Averages of MD quantities applied as continuum BCs.
- Continuum variables constrain MD
- **Nonlinear Additive Schwarz**!
- Adaptively select coupling domains in space-time-stochastic space

**Extension to Exascale**
- Reduce communication
- Fault tolerant
- Uncertainty quantification
- Dynamic load balancing
Algorithmic Improvement

The present approach is an overlapping domain, Schwarz method

- Can be viewed as a predictor-corrector with 1 corrector stage!
- For convergence the continuum time step has to be very small
Algorithmic Improvement

The present approach is an overlapping domain, Schwarz method

- Propose multi-stage corrector!
- Allows larger continuum time steps
Algorithmic Improvement

The present approach is an overlapping domain, Schwarz method

• Propose multi-stage corrector!
• Allows larger continuum time steps and selective MD steps!
A Selective Approach

In time

- Launch new MD calculation
- Allow statistics to heal
- Sample to compute averages
- And so on.....
- Similar to other multiscale ideas (Kevrekides, E, etc.), but stabilized by an interpolation
Inner (MD) – outer (Continuum) structure

- Lots of inner (MD) calculations
- Software-enabled high reliability and a sandbox model of reliability
- Highly reliable outer computations, less reliable inner iterations
- If changes in MD averages are larger than a specified tolerance, then reject and use previous values: *guarantees eventual convergence!*
- Similar approach to FTGMRES [Hoemmen & Heroux]
- Several local recovery detection and strategies within the MD region
- Fault-tolerant averaging operators
- Time integrators with high-frequency damping
Inner (MD) – outer (Continuum)

- Random variables (uncertainties) exist in both MD and continuum domains (i.e., $y_1 \in \mathbb{R}^{d_1}$ and $y_2 \in \mathbb{R}^{d_2}$)
- Combined stochastic space is very large (i.e., $d = d_1 + d_2$ is very large)
- Individual stochastic spaces are also large (i.e., both $d_1$ and $d_2$ are large)
- However, uncertainties are partitioned or localized. It is likely that not every random variable in the MD domain will (strongly) influence the continuum solution and vice-versa
- Additionally, uncertainty is transferred across scales through the overlap region

**y**: random variables

- $y_1 \in \mathbb{R}^{d_1}$
  - $d_1 \sim 0(100)$
  - (e.g., atomic-scale surface heterogeneities, ...)
- $y_2 \in \mathbb{R}^{d_2}$
  - $d_2 \sim 0(10-50)$
  - (e.g., liquid inflow, wall superheat, ...)

Continuum

overlapping region

MD
Hierarchic (stochastic) adaptivity

- Two levels (with inner/MD and outer/continuum structure):
  - In the overlap region (adaptively) construct reduced representation for transfer quantities in terms of random variables associated with MD and continuum domains \(\Rightarrow\) reduced communication
  - In individual stochastic spaces, use (adaptive) methods that avoid or abate curse-of-dimensionality

- Potential approaches:
  - Overlap region: low-rank separated representations (e.g., for partitioned uncertainties), e.g., Doostan et al. CMAME 13, arXiv 13; Chevreuil et al. CMAME 13
  - Individual stochastic space: adaptive (stochastic) approaches, e.g., Maitre et al. JCP 2004; Wan & Karniadakis SIAM JSC 2006; Ma & Zabaras JCP 2009; Agarwal & Aluru JCP 2009
  - Local or global or hybrid approaches

\[
\begin{align*}
\mathbf{y}_2 & \in \mathbb{R}^{d_2} \\
\mathbf{y}_2 & \sim O(10^{-50}) \\
\text{(e.g., liquid inflow, wall superheat, ...)} \\
\mathbf{y}_1 & \in \mathbb{R}^{d_1} \\
\mathbf{y}_1 & \sim O(100) \\
\text{(e.g., atomic-scale surface heterogeneities, ...)}
\end{align*}
\]
Parallel computation

• Many levels of adaptation:
  • Adaptive/selective concurrent modeling
  • Adaptive stochastic approaches
  • Adaptive physical discretization
• Require dynamic load balancing and need to optimize for multiple objectives (flops, comms, faults (inner/outer structure), energy/power, etc.)
• Furthermore, exascale platform will have a significant amount of hierarchy (that must be taken into account):
  • Cores, nodes, drawers, racks, rows, ...

Adaptive/selective concurrent modeling

Adaptive low-rank separated representation

\[ u_c(x, y) \approx \sum_{i=1}^{r} s_l u_0^i(x) u_1^i(y_1) u_2^i(y_2) \]
Hierarchic dynamic load balancing as a potential solution

- Use problem hierarchy to construct a hierarchy of graph/hypergraph for different “subproblems” at different levels. Graph nodes at a level define a graph of a subproblem (i.e., at one level below):
  - For local changes due to adaptivity, apply non-global dynamic load balancing (use graph hierarchy) and possibly an improvement step, e.g., as done for meshes in Zhou et al. SIAM JSC 2010, JofScomp 2012 => minimizes data movement in redistribution/migration
  - Map problem hierarchy to computer-system hierarchy (accounting for interconnect topology), e.g., as done in Charm++ => again, minimizes data movement during solve/analysis steps
Multiscale problems are by definition:

(a) Combinations of heterogeneous calculations
(b) Involve transfer of information across scales

Consequently:

- **Communication** can be reduced by focusing on (b)
- **Fault tolerance** can be achieved by recognizing (a) and a sandbox model
- **UQ** is hard due to (a) but can be simplified by recognizing (b)
- **Dynamic load balancing** is hard due to (a) and (b)

<table>
<thead>
<tr>
<th></th>
<th>Heterogeneity</th>
<th>Scale Transfer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Fault tolerance</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>UQ</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Dynamic load balancing</td>
<td></td>
<td>✔</td>
</tr>
</tbody>
</table>
Thank You