Uncertainty Quantification in Cloud Cavitation Collapse using Multi-Level Monte Carlo

Jonas Šukys
Computational Science and Engineering Laboratory
ETH Zurich, Switzerland

MCQMC 2016
Stanford, California, United States
August 16, 2016
Work in progress in collaboration with

Petros Koumoutsakos
Ursula Rasthofer
Panagiotis Hadjidoukas
Diego Rossinelli
Fabian Wermelinger
Cavitation phenomenon

\[ p + \frac{1}{2} \rho u^2 = \text{const.} \]

Image courtesy: C. Koumoutsakos

Image courtesy: C. Brennen
Single cavity bubble collapse
Destructive power of cavitation

**AVOID** to maintain performance
- turbines (hydroelectricity, pumps)
- high pressure fuel injectors
- high pressure pipes
- propellers

**HARNESS** for medical treatments
- ultrasonic drug delivery
- kidney shockwave lithotripsy
- collapse of cavities near stone surface

Image courtesy:

Image courtesy:
State of the art

EXPERIMENTS

- Cloud interaction parameter, collapse time to radius (Brennen et al.)
- Averaged quantities, damage assessments (Lohse, Keller, Bose et al.)
- Single/double bubble, proximity effects on jetting (Tomita and Shima)

THEORY/MODELS

- Single bubble, radial symmetry (ODE):

SIMULATIONS

- Single bubble (Colonius, Caltech), multiple bubbles with models
- Clouds 120 bubbles, under-resolving and coarse-graining (Adams, TUM)
- Clouds 80 bubbles, k-div terms, interface sharpening (Tiwari, et al., 2015)

UNCERTAINTY QUANTIFICATION IN CAVITATION

- Congedo, Goncalves, Rodio (2015)
- 2D, sDEM [Abgrall, 2015], forward UQ propagation
Governing equations [Kappila] [Masoni] [Allaire]

**Multiphase flow equations**

\[
\begin{align*}
(\alpha_1 \rho_1)_t + \nabla \cdot (\alpha_1 \rho_1 \mathbf{u}) &= 0, \\
(\alpha_2 \rho_2)_t + \nabla \cdot (\alpha_2 \rho_2 \mathbf{u}) &= 0, \\
(\rho \mathbf{u})_t + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p \mathbf{I}) &= 0, \\
E_t + \nabla \cdot ((E + p) \mathbf{u}) &= 0.
\end{align*}
\]

**Advection of phase volume fractions**

\[
(\alpha_2)_t + \mathbf{u} \cdot \nabla \alpha_2 = K(\alpha_{1,2}, \rho_{1,2}, c_{1,2}) \nabla \cdot \mathbf{u}.
\]

**Equation of state (water phase: stiffened)**

\[
\begin{align*}
E &= \frac{1}{2} \rho \mathbf{u}^2 + \Gamma p + \Pi, \\
\Gamma &= \frac{1}{\gamma - 1}, \\
\Pi &= \frac{\gamma p \rho_c}{\gamma - 1}.
\end{align*}
\]

\[
p = \frac{(E - \rho \mathbf{u}^2) - (\alpha_1 \Pi_1 + \alpha_2 \Pi_2)}{\alpha_1 \Gamma_1 + \alpha_2 \Gamma_2}, \\
\frac{1}{\rho c^2} = \frac{\alpha_1}{\rho_1 c_1^2} + \frac{\alpha_2}{\rho_2 c_2^2}.
\]
Problem setup

Generation of the cavity cloud

- **locations**: uniform distribution
- **radii**: log-Gaussian, 50 - 200 µm

<table>
<thead>
<tr>
<th></th>
<th>water</th>
<th>vapor</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho$ [kg/m$^3$]</td>
<td>1000</td>
<td>1</td>
</tr>
<tr>
<td>$p$ [bar]</td>
<td>100</td>
<td>0.0234</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>6.59</td>
<td>1.4</td>
</tr>
<tr>
<td>$p_c$ [bar]</td>
<td>4049</td>
<td>0</td>
</tr>
</tbody>
</table>
Finite Volume Solver

- Cell averages
- Semi-discrete formulation (ODE)
- High order reconstruction
- Approximate Riemann solver HLLC
- RK3 time stepping [Gottlieb, Shu, Tadmor]

\[
\partial_t U(x, t) + \text{div} \ F(U, x) = 0
\]

\[
U_j(t) \approx \frac{1}{|C_j|} \int_{C_j} U(x, t)dx
\]

\[
\frac{d}{dt} U_j(t) + \frac{1}{\Delta x} \left( F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}} \right) = 0
\]

WENO3 / WENO5 [Harten, Shu, Osher]

\[
F_{j+\frac{1}{2}} \approx F_{j+\frac{1}{2}}^{\text{HLLC}}(U^+, U^-)
\]

\[
U_j^n \rightarrow U_j^{n+1}
\]
CUBISM-MPCF

Peta-scale Multi-Phase Compressible Flow approximate Riemann solver

[Rossinelli, Hejazialhosseini, Hadjidoukas, Conti, Bergdorf, Wermelinger, Rasthofer, Šukys]

- Block-based memory layout (spatial locality)
- Instruction/data-level parallelism (Structure of Arrays for SSE/QPX vectorization)
- Domain decomposition MPI/OpenMP (dynamic loop scheduling) (non-blocking P2P communication) (asynchronous progress for C/T overlap)

- ACM Gordon Bell Prize: **14.4 Pflops (72% peak)** on Sequoia (IBM BlueGene/Q, 1.6M cores)
- Wavelet-based I/O compression | ~100x reduction | 1% overhead
- Fault-tolerance with restart mechanism | lossless compression ~10x reduction
Petascale simulations of cloud cavitation collapse
50 thousand cavities at 100 bar
0.5 billion mesh elements
25 thousand time steps
25x pressure amplification
Uncertainty quantification in cloud cavitation collapse
Collapse of two random clouds

2 clouds: different statistical realizations (RNG seeds) of the initial configuration

Spherical clouds of 100 equally sized (75µm) cavities

Uniformly distributed (random) cavity positions
Collapse of two random clouds

2 clouds: different statistical realizations (**RNG seeds**) of the initial configuration

**Maximum pressure**

samples of $p_{\text{max}}$ at level 1 of type 0

Spherical clouds of 100 equally sized (75µm) cavities

Uniformly distributed (random) cavity **positions**

800 - 1500 bar
Multi-Level Monte Carlo [Heinrich, 1999] [Giles, 2008]

Variance reduction technique using sampling on a hierarchy of mesh resolutions
Multi-Level Monte Carlo method

Variance reduction technique using sampling on a hierarchy of mesh resolutions

1. Generate i.i.d. samples of random input quantities for each resolution level $0\ldots L$

2. For each level and sample, solve for approximate solutions using Cubism-MPCF

3. Assemble MLMC estimator for statistics of quantities of interest:

$$
\mathbb{E}[q_L] = \mathbb{E}[q_0] + \sum_{\ell=1}^{L} (\mathbb{E}[q_{\ell}] - \mathbb{E}[q_{\ell-1}]) \approx \frac{1}{M_0} \sum_{i=1}^{M_0} q_0^i + \sum_{\ell=1}^{L} \frac{1}{M_{\ell}} \sum_{i=1}^{M_{\ell}} (q_{\ell}^i - q_{\ell-1}^i).
$$

Sampling error of the MLMC estimator is given in terms of level correlations:

$$
\varepsilon^2 = \frac{\mathbb{V}[q_0]}{M_0} + \sum_{\ell=1}^{L} \frac{\mathbb{V}[q_{\ell} - q_{\ell-1}]}{M_{\ell}} \approx \mathbb{V}[q] \left( \frac{1}{M_0} + 2 \sum_{\ell=1}^{L} \frac{1 - Cor[q_{\ell}, q_{\ell-1}]}{M_{\ell}} \right).
$$
Optimal control variate coefficients

- Each level in MLMC estimator is a special case of control variate with coefficient 1
  \[
  \mathbb{E}[q_\ell] \approx \alpha \mathbb{E}[q_{\ell-1}] + \left( \mathbb{E}[q_\ell] - \alpha \mathbb{E}[q_{\ell-1}] \right).
  \]

- **Optimal** coefficient is given in terms of correlations between two levels
  \[
  \alpha = \frac{\text{Cov}[q_\ell, q_{\ell-1}]}{\text{Var}[q_{\ell-1}]} \approx \text{Cor}[q_\ell, q_{\ell-1}].
  \]

- This argument can be extended to the telescoping sum of the MLMC estimator
  \[
  \mathbb{E}[q_L] = \alpha_0 \mathbb{E}[q_0] + \sum_{\ell=1}^{L} \left( \alpha_\ell \mathbb{E}[q_\ell] - \alpha_{\ell-1} \mathbb{E}[q_{\ell-1}] \right).
  \]

- Related independent work for reused sampling on coarser levels: [Peherstorfer, Willcox, Gunzburger, 2015]
Optimal control variate coefficients

Minimizes variance reduction costs for weakly correlated resolution levels

- **Total computational work-weighted** variance over all levels is given by

\[
C[q^*_L] = \alpha_0^2 \mathbb{V}[q_0]W_0 + \sum_{\ell=1}^{L} \left( \alpha_\ell^2 \mathbb{V}[q_\ell] + \alpha_{\ell-1}^2 \mathbb{V}[q_{\ell-1}] - 2\alpha_\ell \alpha_{\ell-1} \text{Cov}[q_\ell, q_{\ell-1}] \right)W_\ell.
\]

- **Minimization** of the above pertains to solving linear system of equations,

\[
\frac{\partial}{\partial \alpha_\ell} C[q^*_L] = 0, \quad \ell = 0, \ldots, L - 1.
\]

- Linear system can be written in a form of a **diagonally dominant matrix**

\[
\begin{bmatrix}
\sigma_0^2 (W_1 + W_0) & -\sigma_{1,0}^2 W_1 \\
-\sigma_{1,0}^2 W_1 & \ddots & \ddots \\
& \ddots & \ddots & \ddots \\
& \ddots & -\sigma_{L-1,L-2}^2 W_{L-1} & -\sigma_{L-1,L-2}^2 W_{L-1} \\
& \ddots & \ddots & \ddots \\
& \ddots & \ddots & \ddots \\
-\sigma_{L-1,L-2}^2 W_{L-1} & \ddots & \ddots & \sigma_{L-1}^2 (W_L + W_{L-1})
\end{bmatrix}
\begin{bmatrix}
\alpha_0 \\
\alpha_1 \\
\vdots \\
\alpha_{L-2} \\
\alpha_{L-1}
\end{bmatrix}
= 
\begin{bmatrix}
0 \\
0 \\
\vdots \\
0 \\
\sigma_{L,L-1}^2 W_L
\end{bmatrix}
\]
Example and comments

For two levels

- For two levels of resolution (i.e. $L=1$), optimal control variate coefficient is

$$\alpha_0 = \frac{W_1}{W_1 + W_0} \frac{\sigma_{1,0}}{\sigma_0}.$$ 

- Coarsest level already available — classical control variate coefficient is recovered

- Significantly more expensive and **strongly** correlated finer level — classical MLMC

**WARNING:** significantly more expensive but **weakly** correlated finer level with

$$\rho_{1,0} < \frac{1}{2} \frac{W_1 + W_0}{W_1}.$$ 

leads to **variance increase** in MLMC, unless optimal control variates are used
Optimized number of samples [Giles, 2008]

Using empirical estimators for variances and measurements of computations work

- Sampling error of the MLMC estimator is given in terms of level variances:

\[
\varepsilon^2 = \frac{\text{Var}[\alpha_0 q_0]}{M_0} + \sum_{\ell=1}^{L} \frac{\text{Var}[\alpha_\ell q_\ell - \alpha_{\ell-1} q_{\ell-1}]}{M_\ell} \approx \frac{\tilde{\sigma}_0^2}{M_0} + \sum_{\ell=1}^{L} \frac{\tilde{\sigma}_\ell^2}{M_\ell}.
\]

Optimization problem

Given a required tolerance $\tau$ and variances $\sigma^2_\ell$ each level, minimize computational work and find optimal number of samples such that tolerance is attained: $\varepsilon \leq \tau$.

Remark: an analogous result is available for a prescribed computational budget (instead of tolerance).
PyMLMC example

solver setup

```
from solver import Solver

class MySolver(Solver):
    cmd = './mysolver --N=%(N)d --coef=%(coef)d --cores=%(cores)d'

    def run (args, discretization, seed, params):
        args ['N'] = discretization
        random.seed (seed)
        args ['coef'] = random.gauss (mu=params.mu, sigma=params.sigma)
        launch (args)

    def load (directory):
        outputfile = open (directory + 'output.dat', 'r')
        return outputfile.readlines ()

    def work (discretization):
        return discretization ** 3
```

target setup

```
from pymlmc import *

config.discretizations = [128, 256, 512, 1024]
config.solver = MySolver ()
config.samples = Estimated (budget=1e7, tolerance=1e-2)
config.scheduler = Static (cores=32768, walltime=8)

mlmc = MLMC (config)
mlmc.simulation()

statistics = [ NumPy_Stat ('mean'), Deviations (factor=1) ]

mlmc.assemble (statistics)
plot = MatPlotLib (mlmc)
plot.stats_mlmc ()
```

modules for solver, sampling and scheduling

any programming language
no code modifications required
automated job scheduling on clusters
status and progress monitoring for each sample

script

GIT repository: pymlmc.sukys.lt

For a more detailed and accurate examples, please refer to the 'doc/examples' directory in the PyMLMC repository
Insight to inner workings of MLMC

Majority of samples computed on lowest levels of resolution - reduced budget

adaptive number of warmup samples

observed speedup: 230x
Results of MLMC

Uncertainty quantification (i.e. mean, confidence intervals) for QoIs

vapor volume

- No significant uncertainty

pressure sensor

- Wide 90% confidence interval
  - 100 MPa - 600 MPa
Results of MLMC

Secondary cavitation observed at the epicenter immediately after the final collapse

"secondary cavitation" region after the final cloud collapse

average pressure in spherical shells

"secondary cavitation" region after the final cloud collapse

Pressure peak reflects at cloud center sharp drop - "secondary cavitation"
Optimal control variate coefficients

**Speedup**

<table>
<thead>
<tr>
<th>Method</th>
<th>Level samples</th>
<th>Budget in CPU hours</th>
<th>Total speedup</th>
<th>Relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC</td>
<td>$\infty$</td>
<td>3.7 billion</td>
<td>-</td>
<td>1E-02</td>
</tr>
<tr>
<td>MLMC</td>
<td>358, 64, 12, 4</td>
<td>16 million</td>
<td>101.3</td>
<td>1E-02</td>
</tr>
<tr>
<td>OCV-MLMC</td>
<td>214, 19, 9, 2</td>
<td>6 million</td>
<td>230.1</td>
<td>1E-02</td>
</tr>
</tbody>
</table>

higher OCV-MLMC speedup is expected for even less correlated levels
Summary and outlook

- **OCV-MLMC** for uncertainty quantification in multiple sensors for pressure, density, etc.
  - instead of a single value, **confidence intervals for peak pressures** are provided

- **Optimal control variate coefficients** for weakly correlated levels without sample “recycling”

- **Fault tolerance**: if some samples fail, the rest are used to assemble estimators [Pauli, Schwab, Arbenz]

**OUTLOOK**

- discrete optimization for the number of samples [Pauli, Arbenz]
- use Bayes to incorporate assumed prior on the convergence rate [Collier, Haji-Ali, Nobile, Schwerin, Tempone]
- unbiased estimator using randomised resolution levels [Rhee, Glynn, 2015]
- investigate the effect of uncertain and inhomogeneous vapor pressures inside cavities
- investigate the effect of uncertain cloud geometry (e.g. small surface perturbations in a sphere)
HPC resources

**CSCS allocation**
- Project s500
- **Piz Daint**
  - Cray XC30
  - 42,176 cores
  - 5,272 GPUs
  - 7.8 PFlops
  - Switzerland

**PRACE allocation**
- Jülich Research Center
- **JUQUEEN**
  - BlueGene/Q
  - 458,752 cores
  - 5.9 PFlops
  - Germany

**INCITE allocation**
- Argonne National Labs
- **MIRA**
  - BlueGene/Q
  - 786,432 cores
  - 10 PFlops
  - United States

**PRACE allocation**
- CINECA
- **FERMI**
  - BlueGene/Q
  - 163,840 cores
  - 2.1 PFlops
  - Italy
Thanks to

ETH Zürich

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

Argonne National Laboratory

Fonds National Suisse
Schweizerischer Nationalfonds
Fondo Nazionale Svizzero
Swiss National Science Foundation

U.S. Department of Energy
INCITE
Leadership Computing

IBM

PRACE

Jülich Forschungszentrum
THANK YOU
Index
Acceleration of UQ simulations using MLMC

Using empirical estimators for variances and measurements of computational work

- Speedup of the MLMC against plain MC can be estimated as follows

\[
\text{speedup} = \left( \frac{\mathbb{V}[q_L]}{\varepsilon^2} \text{Work}_L \right) / \left( M_0 \text{Work}_0 + \sum_{\ell=1}^{L} M_{\ell} \left( \text{Work}_\ell + \text{Work}_{\ell-1} \right) \right).
\]

- Sampling error of the MLMC estimator is given in terms of level variances:

\[
\varepsilon^2 = \frac{\mathbb{V}[q_0]}{M_0} + \sum_{\ell=1}^{L} \frac{\mathbb{V}[q_\ell - q_{\ell-1}]}{M_{\ell}} \approx \frac{\sigma_0^2}{M_0} + \sum_{\ell=1}^{L} \frac{\sigma_\ell^2}{M_{\ell}}.
\]
Cloud interaction parameters $\alpha$ and $\beta$

**Average radius**

Average radius can be calculated as:

$$R_{\text{avg}} = \frac{1}{N} \sum_{i=1}^{N} R_i$$

**Equivalent radius**

The equivalent radius is given by:

$$R_{\text{eq}} = \sqrt[3]{\sum_{i=1}^{N} R_i^3}$$

**Vapor volume fraction**

The vapor volume fraction is defined as:

$$\alpha = \frac{1}{V_c} \sum_{i=1}^{N} \frac{4}{3} \pi R_i^3$$

**Cloud interaction parameter**

The cloud interaction parameter is expressed as:

$$\beta = \alpha \left( \frac{R_{\text{eq}}}{R_{\text{avg}}} \right)^2$$

(Brennen et al.)
Governing equations [Shyue 1998]

**Euler equations for two phase flows**

\[
\begin{align*}
\rho_t + \text{div}(\rho \mathbf{u}) &= 0, \\
(\rho \mathbf{u})_t + \text{div}(\rho \mathbf{u} \otimes \mathbf{u} + p \mathbf{I}) &= 0, \\
E_t + \text{div}((E + p)\mathbf{u}) &= 0.
\end{align*}
\]

**Stiffened equation of state**

\[
E = \frac{1}{2} \rho \mathbf{u}^2 + \Gamma p + \Pi, \quad \Gamma = \frac{1}{\gamma - 1}, \quad \Pi = \frac{\gamma p c}{\gamma - 1}.
\]

**Advection of phase-dependent fields**

\[
\begin{align*}
\Gamma_t + \mathbf{u} \cdot \nabla \Gamma &= 0, \\
\Pi_t + \mathbf{u} \cdot \nabla \Pi &= 0.
\end{align*}
\]

water, vapor cavities, density $\rho$, velocity vector $\mathbf{u}$, pressure $p$
Problem setup

<table>
<thead>
<tr>
<th></th>
<th>water</th>
<th>vapor</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho ,[\text{kg/m}^3]$</td>
<td>1000</td>
<td>1</td>
</tr>
<tr>
<td>$p ,[\text{bar}]$</td>
<td>100</td>
<td>0.0234</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>6.59</td>
<td>1.4</td>
</tr>
<tr>
<td>$p_c ,[\text{bar}]$</td>
<td>4049</td>
<td>0</td>
</tr>
</tbody>
</table>

Volume fraction ‘$\alpha$’ for mixture values

\[
q = \alpha q_1 + (1 - \alpha)q_2, \\
\gamma = 1 + \frac{(\gamma_1 - 1)(\gamma_2 - 1)}{\alpha(\gamma_1 - 1) + (1 - \alpha)(\gamma_2 - 1)}.
\]

Generation of the cavity cloud

- **locations**: uniform distribution
- **radii**: log-Gaussian, 50 - 200 $\mu$m
Comparison of Eulerian and multiphase codes
Observed **limitations** of MLMC in cloud cavitation collapse
Observed limitations of MLMC

Accuracy requirements limit the number of coarsening levels

Semi-transparent lines correspond to the same sample (realization) but on a coarser resolution.

Differences between two consecutive levels do NOT decrease.

No proper convergence for increasing resolutions.
State of the art
2K bubbles
Piz Daint, 32K cores
64 Billion cells

Babak, Diego, et al.
2 bubbles, $t = 0\ldots2$
JUQUEEN, 8K cores, 4h
Analysis of cavity deformations
Isosurface extraction
\[ \psi = \frac{\pi^{\frac{1}{3}} (6V)^{\frac{1}{3}}}{S} \]

\[ \phi = \frac{V}{V_h} \]
Averages | non-sphericities and non-porosities

propagation of non-spherical cavities from cloud surface to center

propagation of non-porous cavities from cloud surface to center
Holes in the cavities from re-entrant micro-jets

caused by re-entrant micro-jet

http://1.bp.blogspot.com/-uSYxEFf2_sw/Ukw-iDDivXI/AAAAAAAAlj8/X1Vm8thHLhA/s1600/5.+Stages+in+bubble+collapse.png
Topology of the cavities: genus (# of holes)

- genus 0
- genus 1
- genus 2
- (genus 3)

caused by re-entrant micro-jet
Holes in the cavities from re-entrant micro-jets occur omnipresent before final collapse.

Propagation of cavities with holes from cloud surface to center.
Speed of Sound

\[ \phi = \frac{1}{\gamma - 1} \quad \psi = \frac{\gamma \Pi_\infty}{\gamma - 1} \]

\[ p = (\gamma - 1)(E - \frac{1}{2}\rho|\mathbf{u}|^2) - \Pi_\infty \]

\[ c = \sqrt{\gamma RT} = \sqrt{\gamma \frac{R \cdot e_t}{\rho c_v}} \]

\[ = \sqrt{\gamma(\gamma - 1)} \frac{e_t}{\rho} = \sqrt{\frac{1}{\rho} \left( \frac{1}{\Gamma} + 1 \right)(p + \frac{\Pi_\infty}{\Gamma})} \]

\[ e_t = E - \frac{1}{2}\rho|\mathbf{u}|^2 = \rho c_v T \]

\[ \gamma = c_p/c_v \]

\[ R = c_p - c_v \]