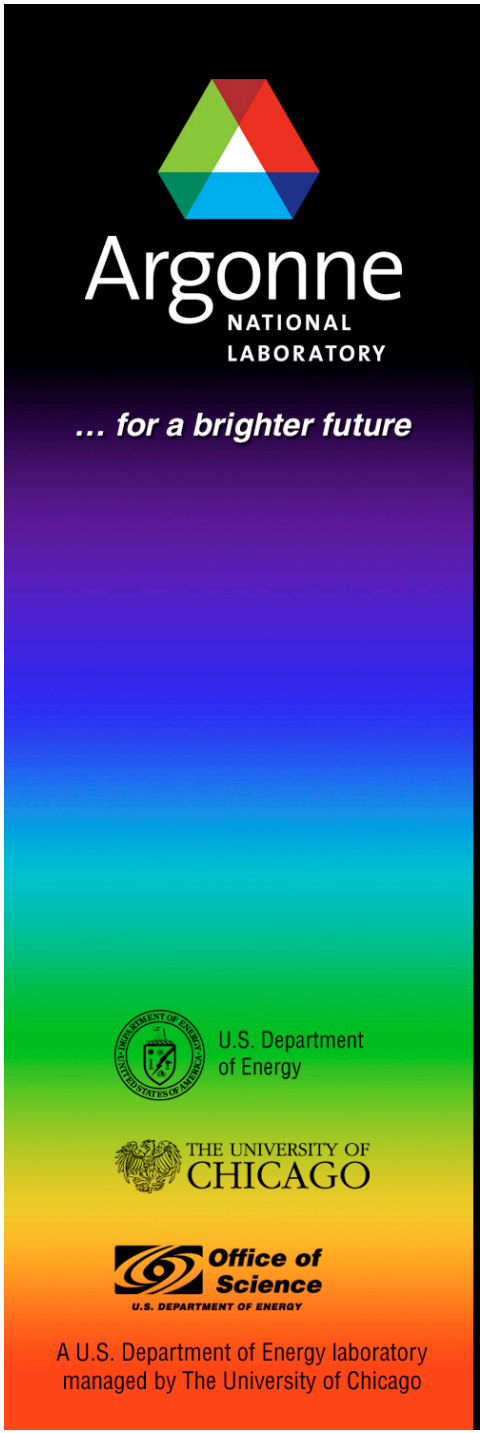


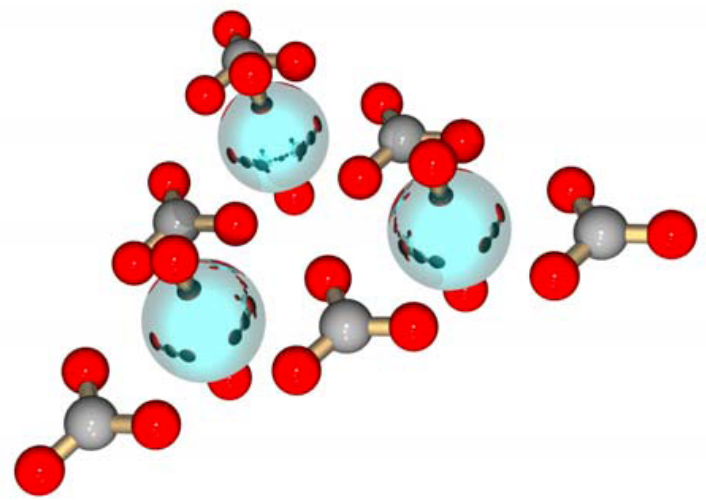
Optimization at Argonne

Mihai Anitescu

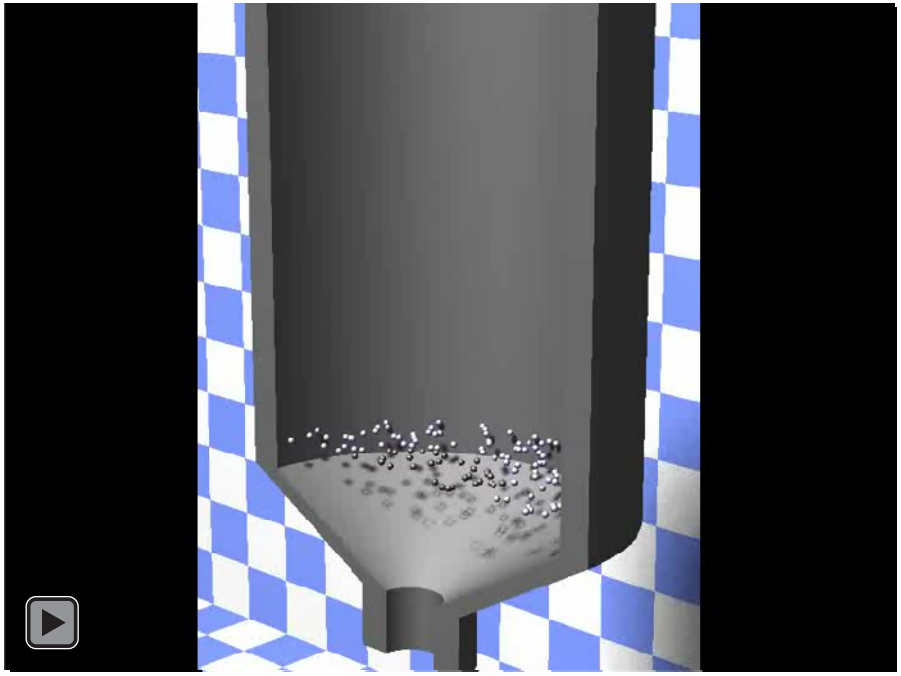
Optimization-Areas of research

- Continuous optimization
  - Complementarity constraints; Diff variational inequality
  - PDE-constrained optimization
  - Derivative-free optimization
  - Multiscale Optimization
- Mixed integer nonlinear programming (MINLP)
- Automatic Differentiation.
- Uncertainty quantification
  - Stochastic Optimization
  - Gaussian Processes

Applications



- Computational Chemistry.
- Pebble Bed Nuclear Reactors.
- Energy Markets/Energy Distribution.
- Nuclear Physics.
- Climate/Economics/Energy Models.
- Nuclear Waste Reprocessing Plants.
- Off-road vehicle design.
- ….



Tools

- We specialize primarily in continuous optimization.

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & g(x) \le 0 \\ & h(x) = 0 \end{aligned}$$

- Software:
  - MINPACK, PATH, filterSQP—serial codes but widely used.
  - Toolkit for advanced optimization (TAO)—highly scaleable , built on PETSC
  - The NEOS server (Beale-Orchard-Hayes prize)

Do we solve big optimization problems? …

- Yes.

Toolkit for advanced optimization (TAO)

- Scalable Optimization Tools for Optimization
- Primarily for PDE-constrained Optimization

Processors Used	BLMVM Iterations	Execution Time	Percentage of Time		
			AXPY	Dot	FG
8	996	1083.8	31	9	60
16	991	538.2	30	10	60
32	966	267.7	29	11	60
64	993	139.5	27	13	60
128	987	72.4	25	15	60
256	996	39.2	26	18	56
512	1000	21.6	23	22	53

Table 1: Scalability of BLMVM on Obstacle Problem with 2,560,000 variables.

Using the Computational Grid ..

- People envision a “Computational Grid” much like the national power grid
- Users can seamlessly draw computational power whenever they need it
- Many resources can be brought together to solve very large problems
- Gives application experts the ability to solve problems of unprecedented scope and complexity, or to study problems which they otherwise would not.
- It is ideally suited for sampling-based applications where the vast majority of the computing can be done asynchronously.

The problem

The World's Largest LP

- Storm – A stochastic cargo-flight scheduling problem (Mulvey and Ruszczyński)
- We aim to solve an instance with 10,000,000 scenarios
- $x \in \Re^{121}, y_k \in \Re^{1259}$
- The deterministic equivalent LP is of size

$$A \in \Re^{985{,}032{,}889 \times 12{,}590{,}000{,}121}$$

The "supercomputer"

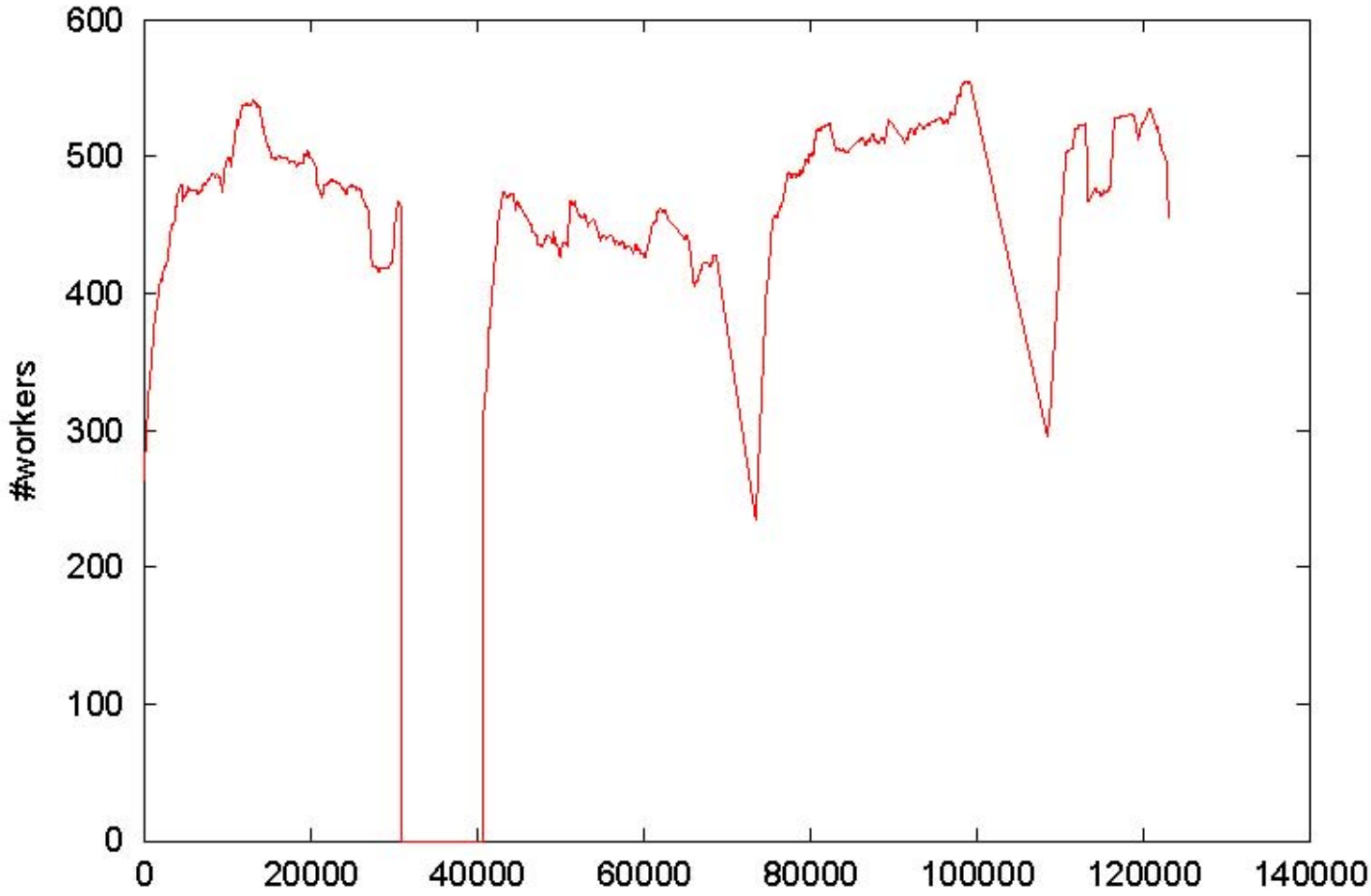
- An ad-hoc entity created with help of grid middleware (CONDOR-GLOBUS)

Number	Type	Location
184	Intel/Linux	Argonne
254	Intel/Linux	New Mexico
36	Intel/Linux	NCSA
265	Intel/Linux	Wisconsin
88	Intel/Solaris	Wisconsin
239	Sun/Solaris	Wisconsin
124	Intel/Linux	Georgia Tech
90	Intel/Solaris	Georgia Tech
13	Sun/Solaris	Georgia Tech
9	Intel/Linux	Columbia U.
10	Sun/Solaris	Columbia U.
33	Intel/Linux	Italy (INFN)
1345		

Statistics

Wall clock time	31:53:37
CPU time	1.03 Years
Avg. # machines	433
Max # machines	556
Parallel Efficiency	67%
Master iterations	199
CPU Time solving the master problem	1:54:37
Maximum number of rows in master problem	39647

Worker performance



NEOS Server

- In addition to creating software, or solving specific applied math problems, we support the NEOS server.
- http://www-neos.mcs.anl.gov
- Optimization Software Developers connect their own solvers to NEOS.
- Users can try multiple solvers.
- Submission is supported by web interface or as remote library in either modeling language (AMPL, GAMS) or programming language input (MATLAB, FORTRAN).
- Supports differentiable constrained optimization, mixed integer nonlinear programming, global optimization, complementarity problems, network` programming …

Mathematical Programming with Equilibrium/ Complementarity Constraints (MPEC/MPCC)—Definition and context

- MPEC are important model problems in structural engineering, economics, data analysis, and have generated substantial activity in NLP algorithmic research.
- They appear (for example) if the multipliers of a lower-level optimization problem are primal variables in an embedding optimization problem (bilevel optimization) .

$$
\begin{aligned}
\min_{x,y,w,z} \quad & f(x,y,w) & \\
\text{sbj.to} \quad & g(x,y,w) & \le 0 \\
& h(x,y,w) & = 0 \\
& y, w & = 0 \\
(y^T w = 0) \quad & y^T w & \le 0
\end{aligned}
$$

Challenges in the algorithmics of MPEC

- The difficulty of this problems arises from the fact that, at a solution, the gradients of the active constraints are *linearly dependent,* and that, even worse, *the Lagrange multipliers are unbounded.*
- Initial investigation with classical nonlinear programming algorithms showed a 50% failure rate.
- It was long believed that these problems can be reliably solved only by specialized techniques such as bundle trust region methods and disjunctive technique, both of which may have exponential complexity.

The elastic mode approach

- The elastic mode: a classical safeguarding technique.
- Key: for the *generic* case, for c sufficiently large but fixed, the elastic mode problem has the same solution as the MPEC, but bounded multipliers (2005, in print).

$$
\begin{aligned}
\min_{x,y,w,z} \quad & f(x,y,w) + c(e_h^T \xi_h + e_g^T \xi_g + \xi_c) \\
\text{sbj.to} \quad & g(x,y,w) && \le \xi_g \\
-\xi_h \le \quad & h(x,y,w) && \le \xi_h \\
& y, w && = 0 \\
& y^T w && \le \xi_c
\end{aligned}
$$

Results/Impact

- We have shown that the elastic mode solves the generic case and achieves superlinear convergence (2005, in print).
- We have obtained global convergence results for bilevel optimization problems (2005, in print), as well as for the general MPEC, while solving *inexact subproblems* (2005, submitted).
- Sven Leyffer, in addition of investigating the benefits of using a filter approach, has created the MacMPEC test set, which is the de facto standard for MPEC algorithm testing.
- 3 of our initial papers have 40, 25, 24 Google citations.
- Elastic mode has been implemented, beyond SNOPT, by LOQO and KNITRO and successfully applied to MPEC.

Can I simulate efficiently systems with many switches …

- Such as granular dynamics. Normally attacked by DEM

- Yes.

Nonsmooth contact dynamics—what is it?

- Differential problem with variational inequality constraints – DVI

Newton Equations

Non-Penetration Constraints

$$M\frac{dv}{dt} = \sum_{j=1,2,\ldots,p}\left(c_n^{(j)}n^{(j)} + \beta_1^{(j)}t_1^{(j)} + \beta_2^{(j)}t_2^{(j)}\right) + f_c(q,v) + k(t,q,v)$$

$$\frac{dq}{dt} = \Gamma(q)v$$

Generalized Velocities

$$c_n^{(j)} \geq 0 \quad \perp \quad \Phi^{(j)}(q) \geq 0, \qquad j = 1,2,\ldots,p$$

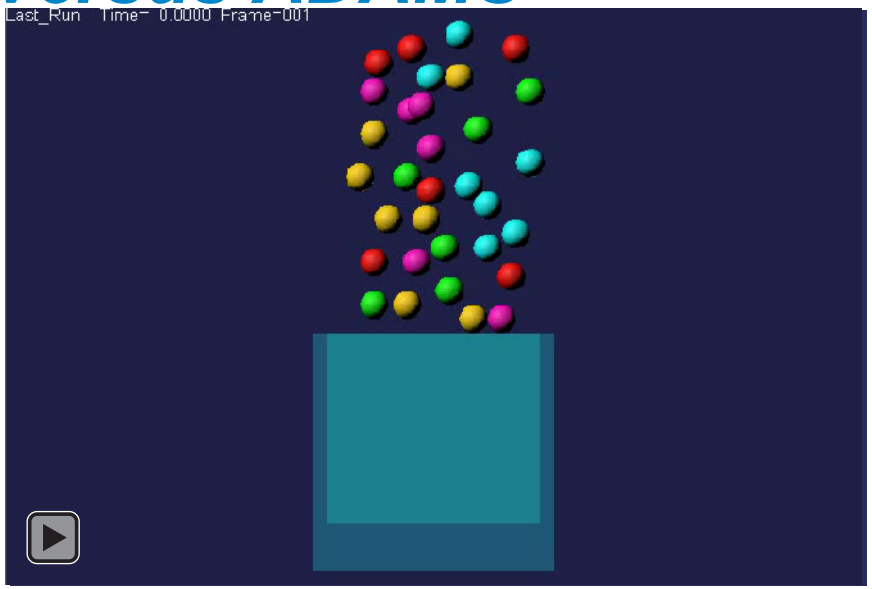
$$\left(\beta_1^{(j)}, \beta_2^{(j)}\right) = \operatorname{argmin}_{\mu^{(j)}c_n^{(j)} \geq \sqrt{\left(\beta_1^{(j)}+\beta_2^{(j)}\right)^2}} \left[\left(v^T t_1^{(j)}\right)\beta_1 + \left(v^T t_2^{(j)}\right)\beta_2\right]$$

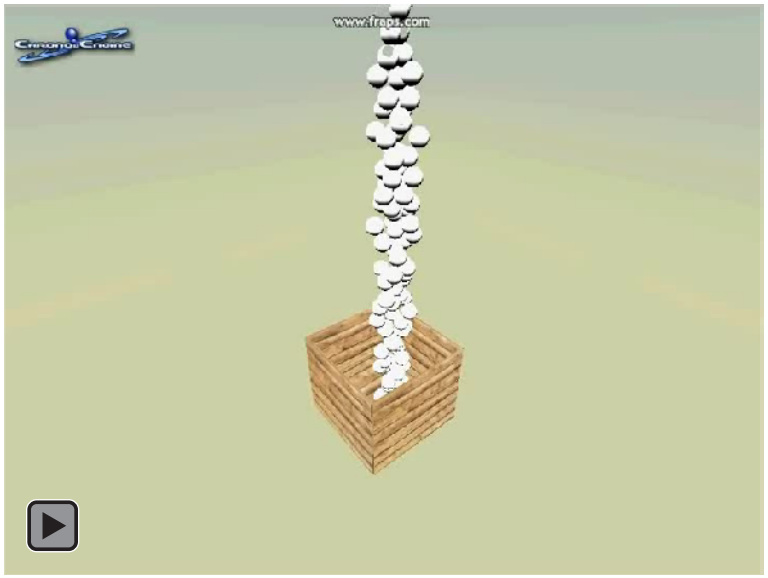
Friction Model

- Truly, a Differential Problem with Equilibrium Constraints

Differential Optimization versus ADAMS

- ADAMS is the workhorse of engineering dynamics.
- ADAMS/View Procedure for simulating.
- Spheres: diameter of 60 mm and a weight of 0.882 kg.
- Forces:smoothing with stiffness of 1E5, force exponent of 2.2, damping coefficient of 10.0, and a penetration depth of 0.1 … even if it is not yet accurate enough!





ADAMS versus ChronoEngine

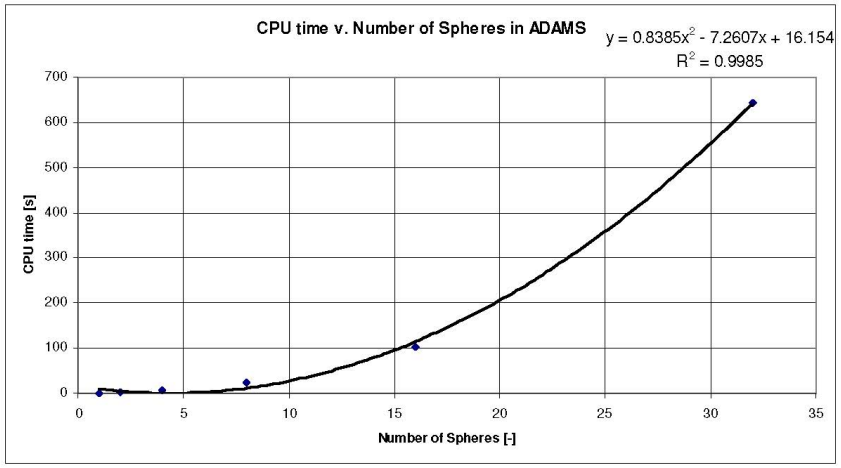
Table 1: Number of rigid bodies v. CPU time in ADAMS

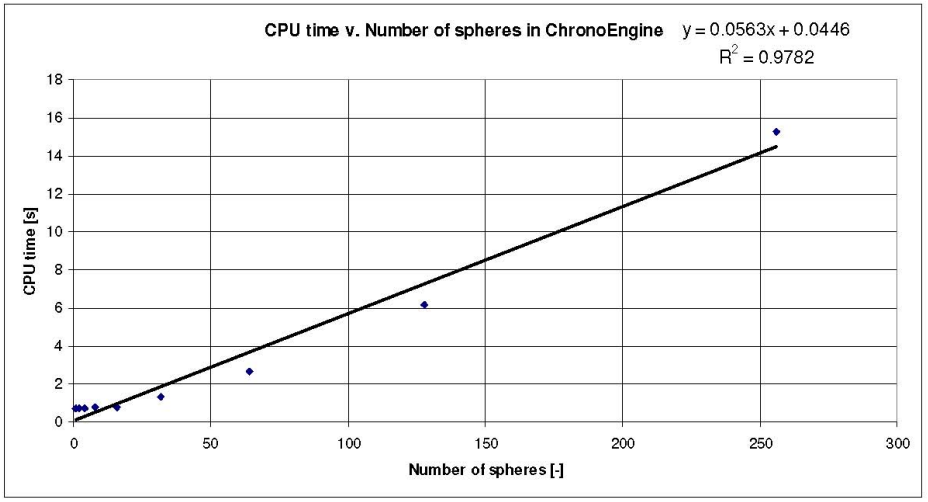
Number of Spheres	Max Number of Mutual Contacts [-]	CPU time (seconds)
1	1	0.41
2	3	3.3
4	14	7.75
8	44	25.36
16	152	102.78
32	560	644.4

The following graph shows the nonlinear increase in the CPU time as the number of colliding bodies increases.

Table 2: Number of rigid bodies v. CPU time in ChronoEngine

Number of Spheres	Max Number of Mutual Contacts [-]	CPU time (seconds)
1	1	0.70
2	3	0.73
4	14	0.73
8	44	0.76
16	152	0.82
32	560	1.32
64	2144	2.65
128	8384	6.17
256	33152	15.30





Conclusion 1: Often, time stepping is more promising,

Hard Constraint (time stepping) Simulations

- Different from hard particle, since they do not necessarily stop at collisions, and do not suffer from the strong time step limitation of penalty (spring and dashpot) approaches.
- At every step they solve a Linear Complementarity Problem.
- Our contributions have included the definitions of scheme with solvable subproblems (1997), accommodation of stiffness (2001), constraint stabilization (2004), fixed time step (2004).
- Method is widely used in robotics and gaming.
- Unfortunately, the subproblem may be nonconvex (2004), which may be a considerable obstacle for large scale applications.

Optimization-based simulation of nonsmooth dynamics.

- We have defined a scheme that solves only convex subproblems, and takes a fixed time step.
- We proved that the method converges to the same weak formulation as the original approach (2005, in print).
- We have simulated segregation of granular materials with 10^4 larger time step than spring dashpot models.

$$v^{(l+1)} = \operatorname{argmin}_{\hat{v}} \quad \frac{1}{2}\hat{v}^T M \hat{v} + k^{(l)T}\hat{v}$$

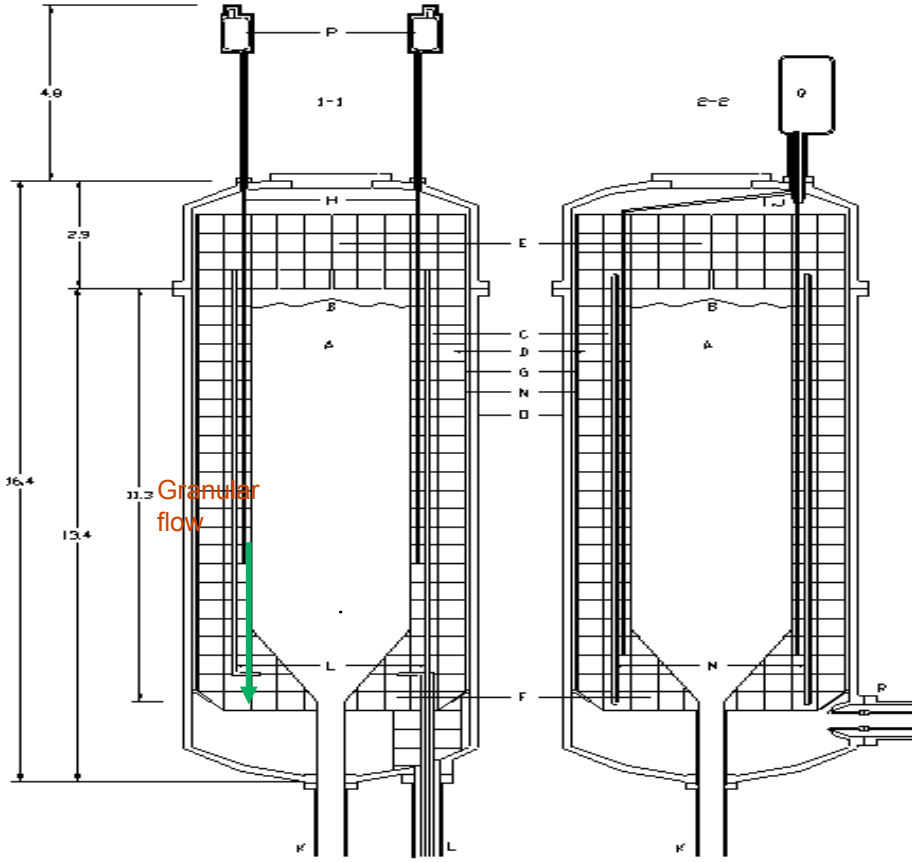
$$\text{subject to} \quad \nabla\Phi^{(j)T}\hat{v} - \mu^{(j)}\sqrt{\left(t_1^{(j)T}\hat{v}\right)^2 + \left(t_2^{(j)T}\hat{v}\right)^2}$$

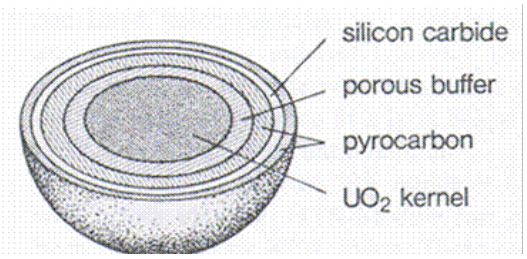
$$\frac{1}{h}\Phi^{(j)}(q^{(l)}) \geq 0$$

$$j \in A(q^{(l)}, \varepsilon), \quad k = 1, 2, \ldots, m^{(j)}$$

Simulating the PBR nuclear reactor

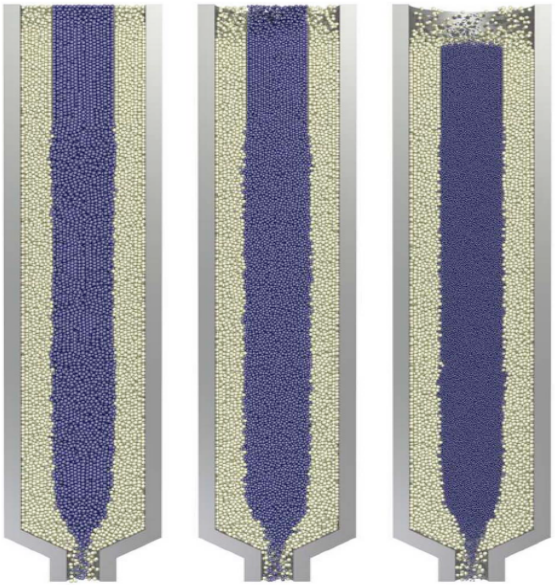
- The PBR nuclear reactor:
  - Fourth generation design
  - Inherently safe, by Doppler broadening of fission cross section
  - Helium cooled > 1000 °C
  - Can crack water (mass production of hydrogen)
  - Continuous cycling of 360’000 graphite spheres in a pebble bed





Simulating the PBR nuclear reactor

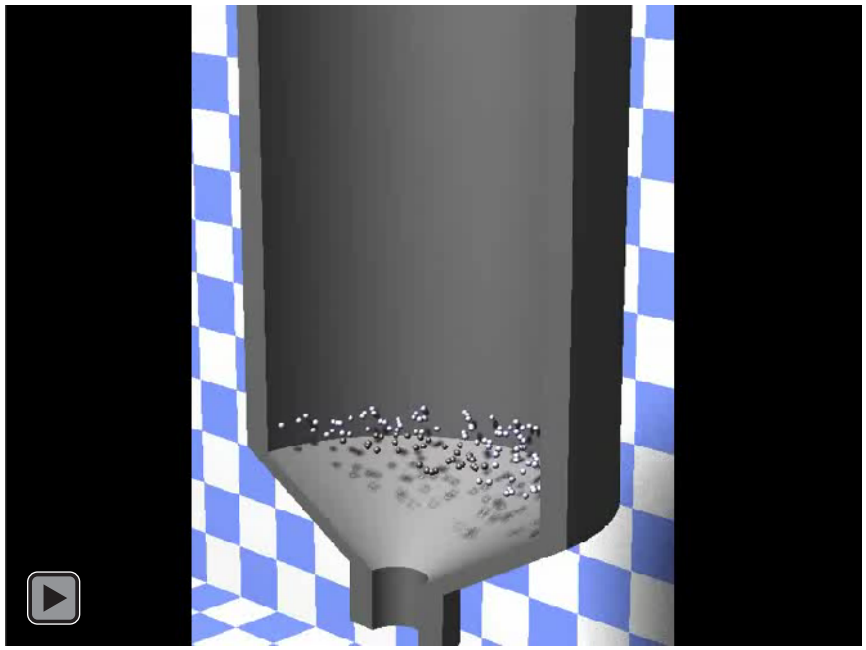
- Problem of bidisperse granular flow with dense packing.
- Previous attempts: DEM methods on supercomputers at Sandia Labs regularization)
- 40 seconds of simulation for 440,000 pebbles needs 1 week on 64 processors dedicated cluster (Rycroft et al.)



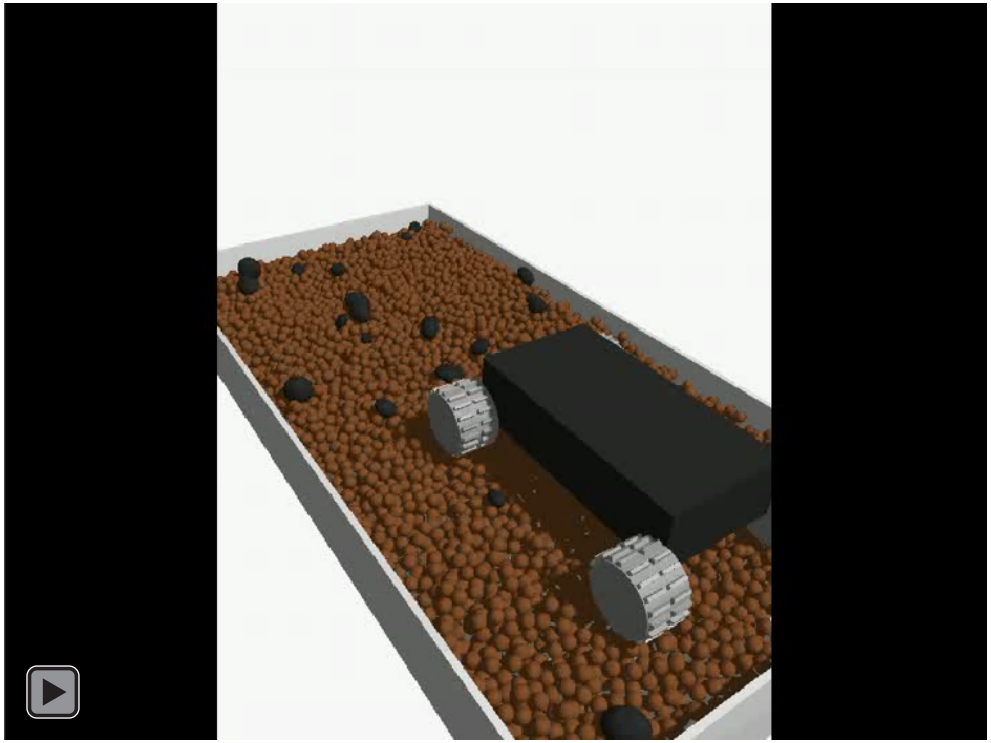
model a frictionless wall, $\mu_w = 0.0$. For the current simulations we set $k_t = \frac{2}{7} k_n$ and choose $k_n = 2 \times 10^5$ gm/d. While this is significantly less than would be realistic for graphite pebbles, where we expect $k_n > 10^{10}$ gm/d, such a spring constant would be prohibitively computationally expensive, as the time step scales as $\delta t \propto k_n^{-1/2}$ for collisions to be modeled effectively. Previous simulations have shown that

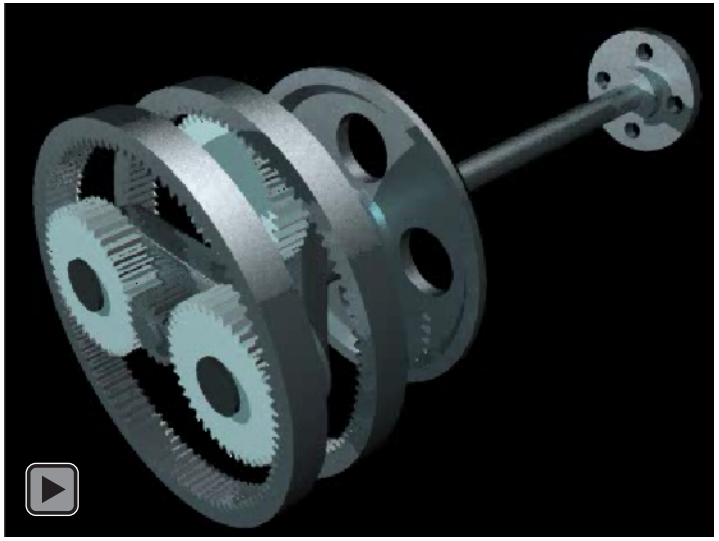
Simulating the PBR nuclear reactor

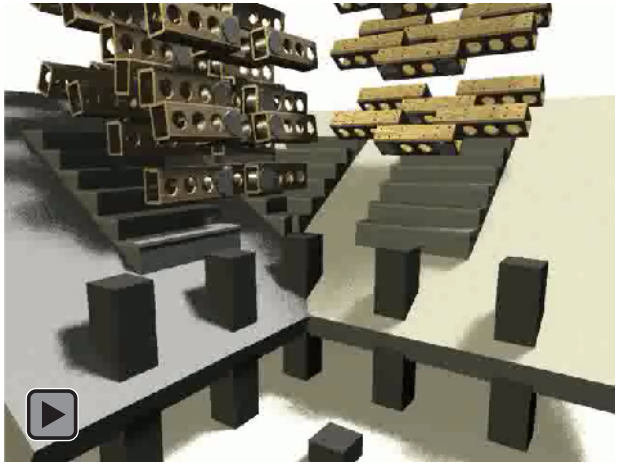
- 160’000 Uranium-Graphite spheres, 600’000 contacts on average
- Two millions of primal variables, six millions of dual variables
- *1 day on a Windows station…*
- But we are limited by the 2GB user mode limit, 64 bit port in progress—but linear scaling..
- We estimate 3CPU days, compare with 450 CPU days for an incomplete solution in 2006 !!!
- Answer 3: Our approach is efficient for large scale!!



In addition, we can approach efficiently approach many engineering problems (see website for papers)

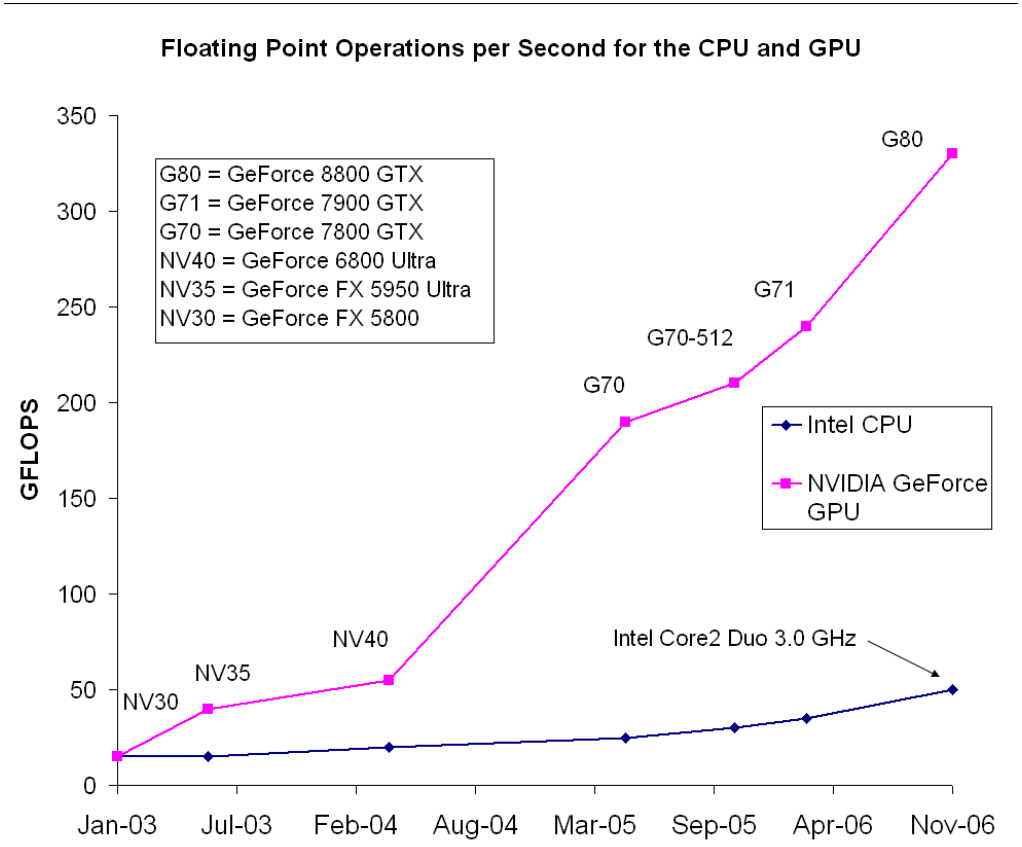






New computational opportunity: GPU

- Entry BG/L model: 1024 dual core nodes
- 5.7 Tflop (compare to 0.5 Tflop for NVIDIA Tesla GPU)
- Dedicated OS
- Dedicated power management solution
- Require dedicated IT support
- Price (2007): $1.4 million
- Same GPU power (2008): 7K!!!



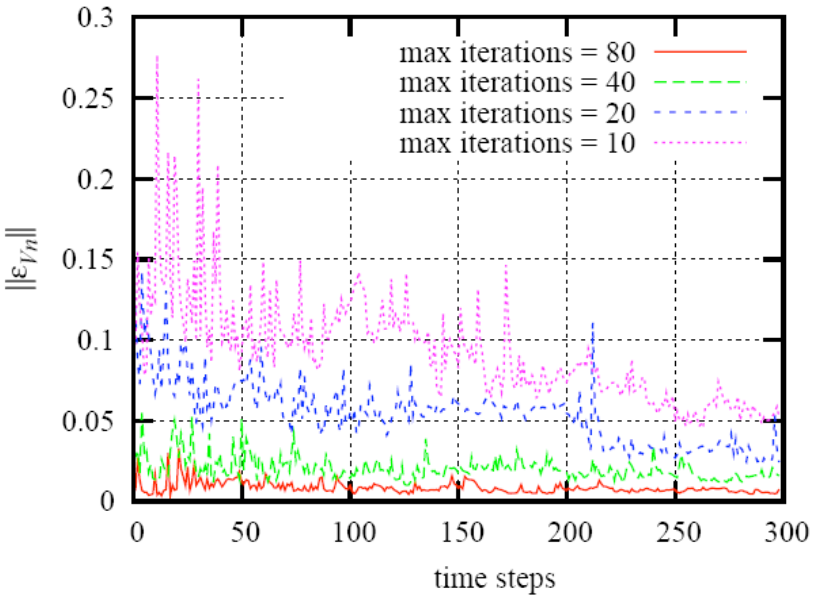
Brick Wall Example…

- Times reported are in seconds for one second long simulation
- GPU: NVIDIA GeForce 8800 GTX

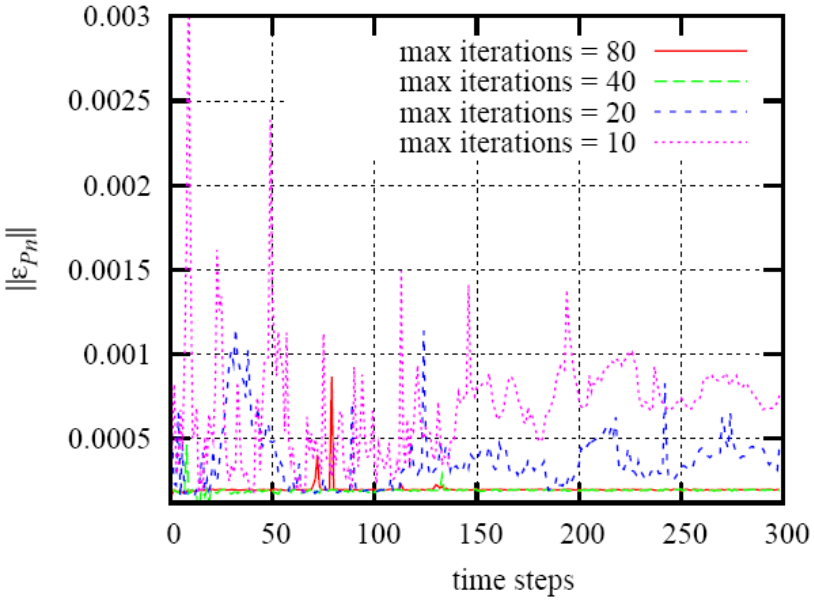
Bricks	Sequential Version	GPU Co-processing Version
1000	43	6
2000	87	10
8000	319	42

Tests

- Feasibility accuracy increases with number of iterations:



Speed violation in constraints

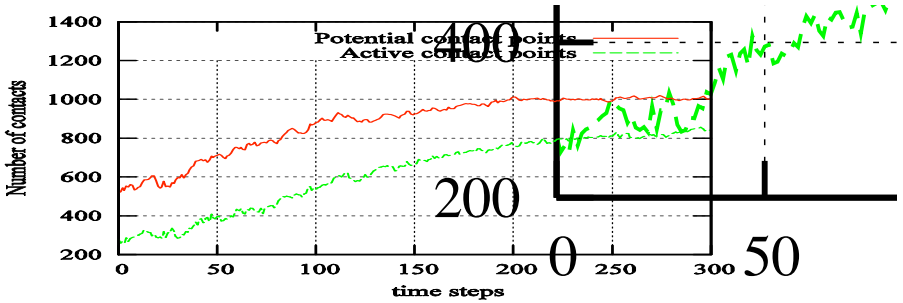


Position error in constraints (penetration)

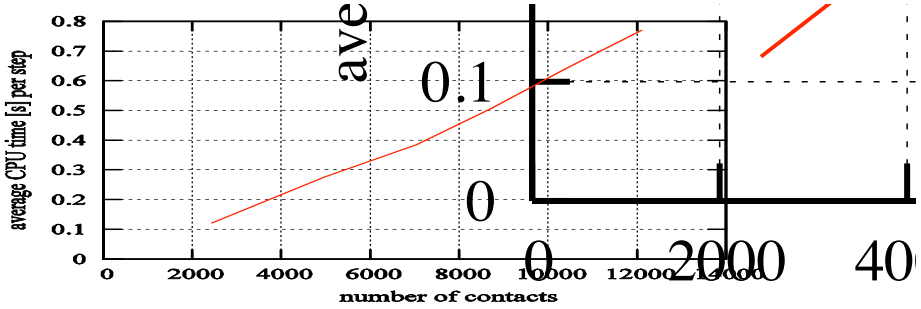
(with example of 300 spheres in shaker)

Tests: Scalability

- CPU effort per contact, since our contacts are the problem variables.
- Penetration error was uniformly no larger than 0.2% of diameter.



Number of contacts in time, 300 spheres



CPU time per step for 300-1500 spheres

Multiscale and Optimization

- If I reduce an optimization problem by a multiscale ansatz, will it be well posed? …

- Sometimes.

Multiscale Approaches for Problems in Material Science.

- … inspired by the quasicontinuum approach (Tadmor et al.).
- High resolution model

$$(O)\quad \begin{array}{ll} \min & f(x_1, x_2) \\ \text{sbj. to} & g(x_1, x_2) = 0. \end{array}$$

- **Representative (coarse-scale) DOF,** $x_1, \dim(x_1) \ll \dim(x_2).$
- **Key observation: at the solution of the problem we have** $x_2 \approx T x_1$ **, where** T **is an interpolation operator.**
- **Replace *(O)* with *(RE),* of much smaller dimension**

$$(RE)\quad \begin{aligned} \nabla_{x_1} f(x_1, Tx_1) + \nabla_{x_1} g(x_1, Tx_1)\lambda &= 0, \\ g(x_1, Tx_1) &= 0. \end{aligned}$$

Electronic structure computations

- Electronic structure computation, a fundamental problem in chemistry and nanomaterials.

$$\min_{\rho \geq 0} \quad E(\rho, \{\mathbf{R}_A\})$$

$$\text{s.t.} \quad \int \rho = N$$

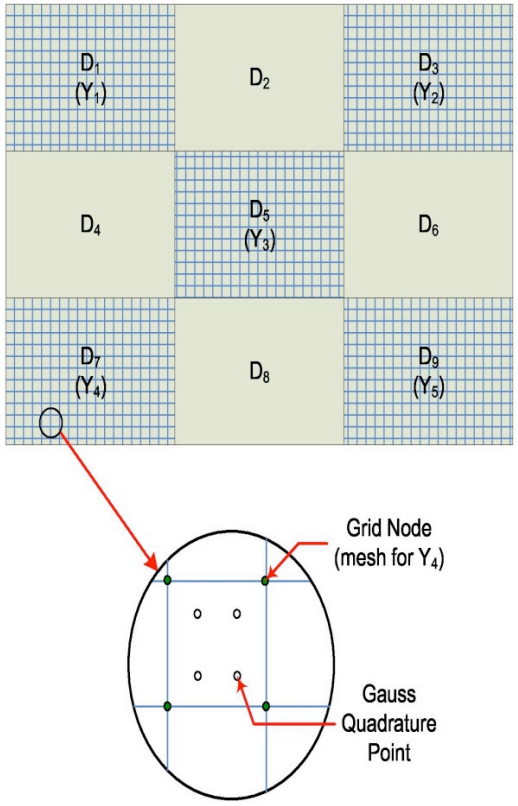
$$E\left[\rho, \{\mathbf{R}_A\}\right] = E_{ne}\left[\rho, \{\mathbf{R}_A\}\right] + J\left[\rho\right] + K\left[\rho\right] + T\left[\rho\right].$$

- **Functionals $E_{ne}\left[\rho, \{\mathbf{R}_A\}\right], J\left[\rho\right]$ are computed from classical electrostatic theory, but $K\left[\rho\right]$ and, especially, $T\left[\rho\right]$ (kinetic) need to be computed by Density Functional Theory (DFT).**
- **In orbital-free DFT (Carter et al.), the functionals are explicit, but not in Kohn-Sham DFT (which is more accurate).**

A multiscale approach for electronic density nanoscale simulations

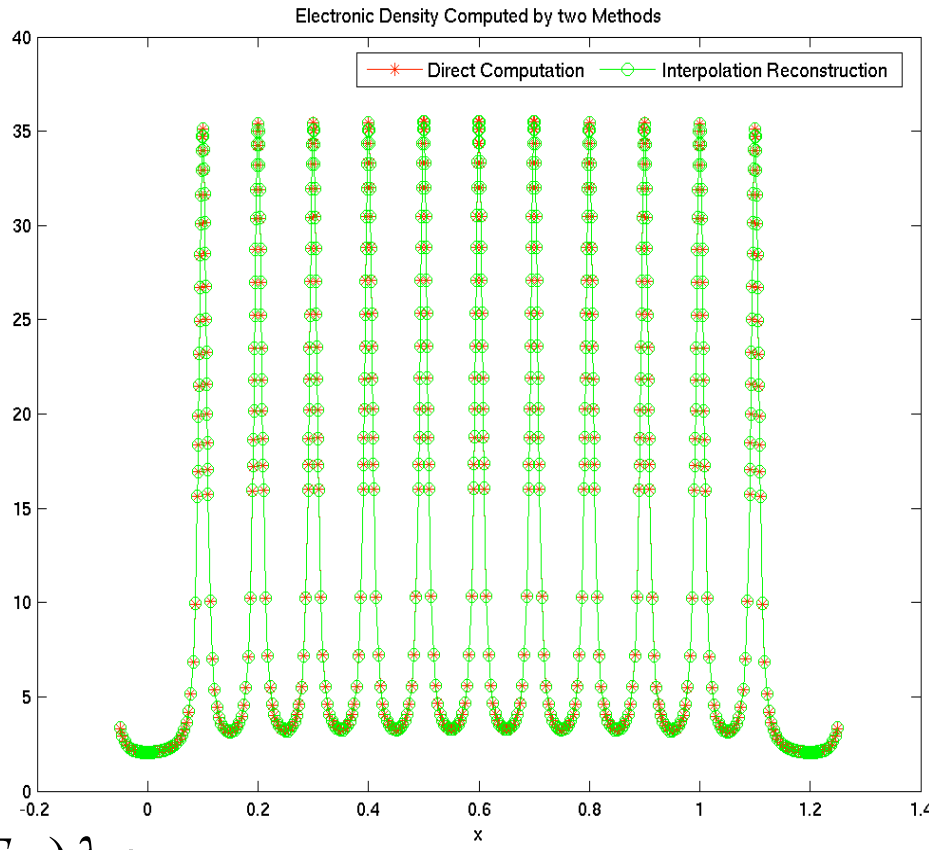
- **Kohn Sham can deal with ~10^3 atoms, whereas nanoparticles may have 10^7 atoms.**
- **Multiscale approach: reduces the degrees of freedom.**
- **Representative variables: The density in the representative domains** $Y_\alpha, \alpha = 1, 2, \ldots, p$.
- **The interpolation operator is constructed with respect to a reference crystalline mesh (2005, in print)**

$$\rho_i(\Phi(\mathbf{r}^0, t)) = \sum_{\alpha=1}^{p} \vartheta_\alpha(i)\rho_\alpha(\Phi(\mathbf{r}^0 + \mathbf{T}_{i\alpha}, t)).$$



The nonlinear variational inequality approach

- We substitute the interpolation operator in the optimality conditions, (VI).
- Example: Thomas-Fermi DFT on 11 Hydrogen atoms, using less than 50% degrees of freedom.



$$x_1 \geq 0, \quad r \geq 0$$

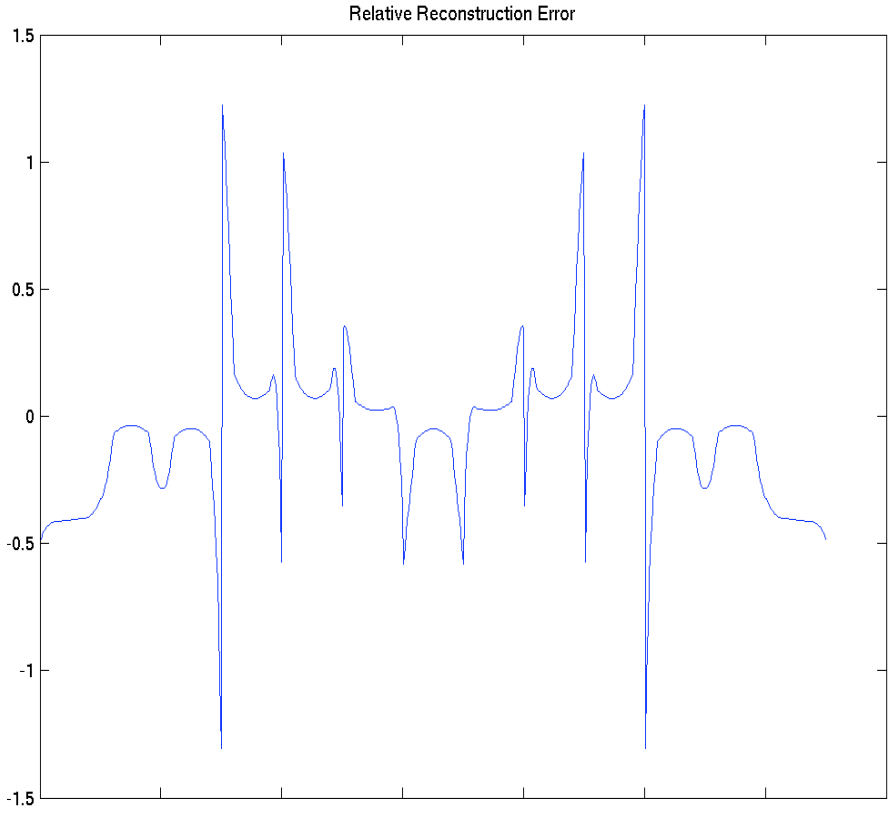
$$(VI)\ 0 = g(x_1, Tx_1)$$

$$0 = \nabla_{x_1} f(x_1, Tx_1) + \nabla_{x_1} g(x_1, Tx_1)\lambda + \eta.$$

The nonlinear optimization approach

- **"Interpolate and optimize" as opposed to "Optimize and Interpolate".**
- **Allows us to use optimization tools, with costlier setup but more robustness.**
- **We proved that *(RO)* gives solutions of the same quality as *(RE)* (2005, in preparation)**

$$(RO)\quad \begin{aligned} \min \quad & f(x_1, Tx_1) \\ \text{sbj. to} \quad & g(x_1, Tx_1) = 0, \\ & x_1 \geq 0. \end{aligned}$$



Interpolate and Optimize, one step further

- Interpolation gives assembly rule with precomputable kernels.

$$J(\rho)=\frac{1}{2}\sum_{\alpha=1}^{p}\sum_{\gamma=1}^{p}\int_{Y_\alpha^0}\int_{Y_\gamma^0}\tilde{K}_{\alpha\gamma}(\mathbf{r}^0,\mathbf{r}^{0'})\rho_\alpha(\Phi(\mathbf{r}^0,t))\rho_\gamma(\Phi(\mathbf{r}^{0'},t))d\mathbf{r}^0 d\mathbf{r}^{0'}$$

$$E_{ne}(\rho)=-\sum_{\alpha=1}^{p}\int_{Y_\alpha^0}\tilde{L}_\alpha(\mathbf{r}^0)\rho_\alpha(\Phi(\mathbf{r}^0,t))d\mathbf{r}^0,$$

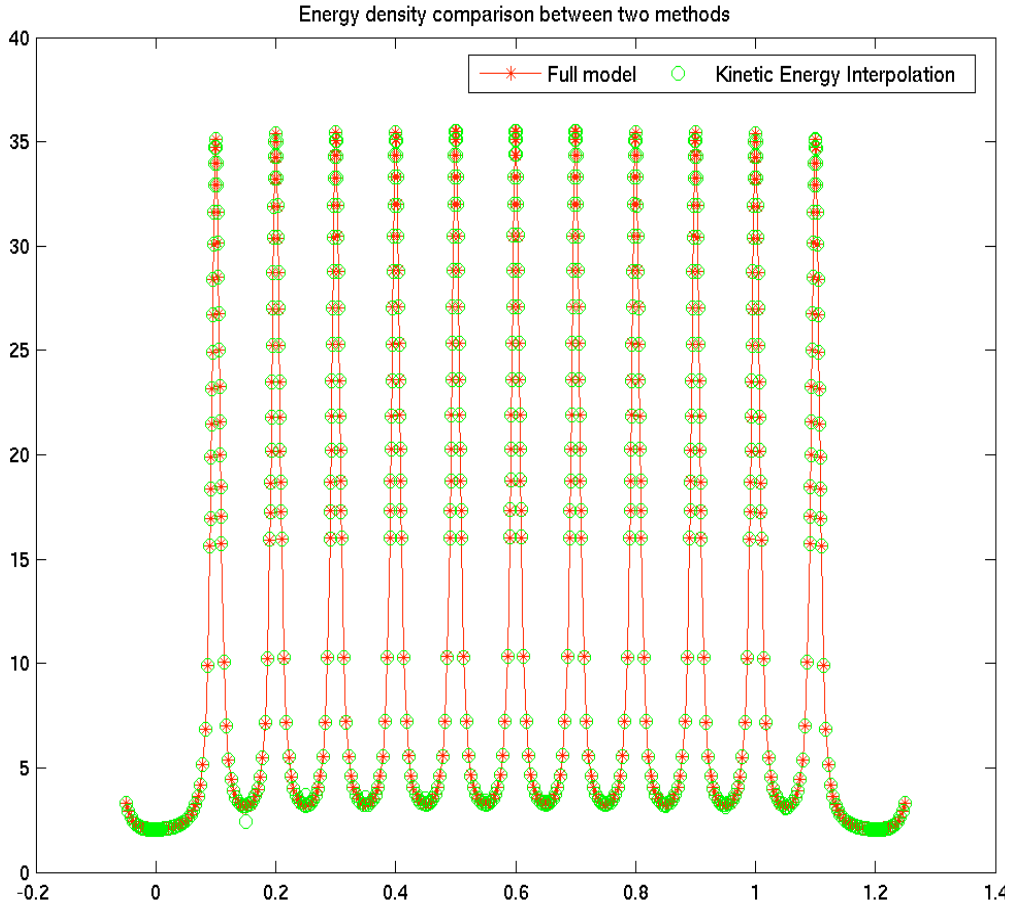
$$\int\rho d\mathbf{r}=\sum_{\alpha=1}^{p}\int_{Y_\alpha^0}\tilde{M}_\alpha(\mathbf{r}^0)\rho_\alpha(\Phi(\mathbf{r}^0,t))d\mathbf{r}^0.$$

- **By a separation of scales argument,**

$$T[\rho]+K[\rho]\approx\sum_{\alpha=1}^{p}\int_{Y_\alpha^0}\tilde{M}_\alpha(\mathbf{r}^0)\theta^1(\rho_\alpha,\Phi(\mathbf{r}^0,t))d\mathbf{r}^0\approx\sum_{\alpha=1}^{p}w_\alpha\left(T[\rho]+K[\rho]\right)_\alpha.$$

Results for the kinetic energy interpolation approach

- 11 Hydrogen atoms.
- There are a few domain boundary artifacts but do not exceed 2% of peak.
- Investigation in superior interpolation techniques is warranted .



Algorithms for rigid encapsulation of DFT

- Kohn-Sham encapsulations are not $\rho \to T[\rho], \partial_\rho T[\rho]$ but rather

$$\bar{V}_{Y_\alpha}(\bullet) \to \rho = \arg\min_{\rho_\alpha, \int \rho_\alpha = q_\alpha} E_\alpha\left[\rho, \bar{V}_{Y_\alpha}\right].$$

- **With "interpolate and optimize", we obtain the following expression, where L is assembled using precomputable kernels and acts as an external field**

$$\min_{\rho \geq 0} \quad \sum_{\alpha=1}^{p} w_\alpha E_\alpha(\rho, \{\mathbf{R}_A\}) + L\left(\{\rho_\alpha\}_{\alpha=1,p}\right)$$

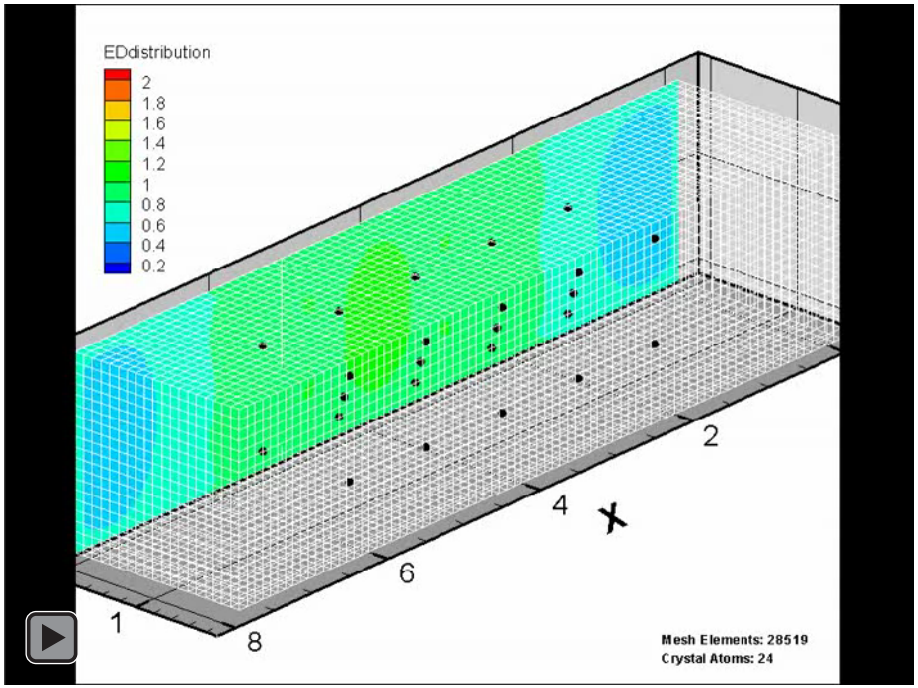
$$\text{s.t.} \quad \sum_{\alpha=1}^{p} \int_{Y_\alpha^0} \tilde{M}_\alpha(\mathbf{r}^0)\rho_\alpha(\Phi(\mathbf{r}^0, t))d\mathbf{r}^0 = N.$$

Algorithms for rigid encapsulation of DFT (II)

- **We can use a nested SOR Gauss-Seidell iteration which at every step uses the rigid Kohn-Sham encapsulation.**
- **We proved convergence for sufficiently small relaxation parameter (2005, in preparation).**
- **As more flexible Kohn-Sham implementations, that provide more local information, become available, the convergence can be accelerated.**

Status

- Project started in January, with LDRD support.
- We completed 1-dimensional test cases, and implementation of 3D Thomas-Fermi DFT w/o interpolation.
- Joint work with Peter Zapol (ANL/MSD) , Dan Negrut, Todd Munson, (ANL/MCS), Adrian Kopacz (NWU), 2 proceedings papers and 1 in progress.



What is the right frame of mind for decision making?

- Usually design (optimization) and risk analysis/ uncertainty are separate endeavors.

- We argue (complex systems call) that they are hard-to-separate elements of decision making.

Optimization under uncertainty

- Necessary for robust and cost-effective design.
- Perhaps not so widespread due to the fact that computational effort was a limitation until very recently.
- One possible formulation – stochastic optimization

$$\begin{aligned} \min_{x,(z)(\omega),(y)(\omega)_{\omega\in\Omega}} \quad & E_\omega\left[f(x, y(\omega), z(\omega), \omega)\right] \\ \text{subject to} \quad & c^1(x, y(\omega), s(\omega), z(\omega), \omega) \leq 0; \omega \in \Omega \\ & c^2(x, y(\omega), s(\omega), z(\omega), \omega) = 0; \omega \in \Omega \\ & c^3(x, y(\omega), s(\omega), z(\omega), \omega) = 0; \omega \in \Omega \\ & 0 \leq y_1(\omega) \perp s(\omega) = F(x, y(\omega), z(\omega), \omega) \geq 0; \omega \in \Omega, \end{aligned}$$

- Used for crew scheduling, portfolio optimization, investigated in nuclear reactor design, nuclear waste plant design ..

Design under uncertainty of nuclear reactors

- A huge, multiphysics, multiple time scales, stochastic optimization problem.
- The workhorse technique has been first-order sensitivity analysis with Gaussian pdf (Cacuci, 1982). Probably a precursor for many other fields – the favorite complex system example.
- It has been undertaken for more than 5 decades (even for ABR) … so is there something to still do here?
- To answer this, we look at the economics and the various scales of the problem.

Nuclear Power Economics 101

- US Nuclear Business in FY06 sales: $64B
- Annual costs for a **new** 1000 MWe power plant
  - Total = $391M (4.9 cents/KWh)
  - Capital = $254M (3.2 cents/KWh)
  - O&M = $101M (1.27 cents/KWh)
  - Fuel = $36M (0.45 cents/ KWh)

Source: Paul Turinsky – MCSNA 07

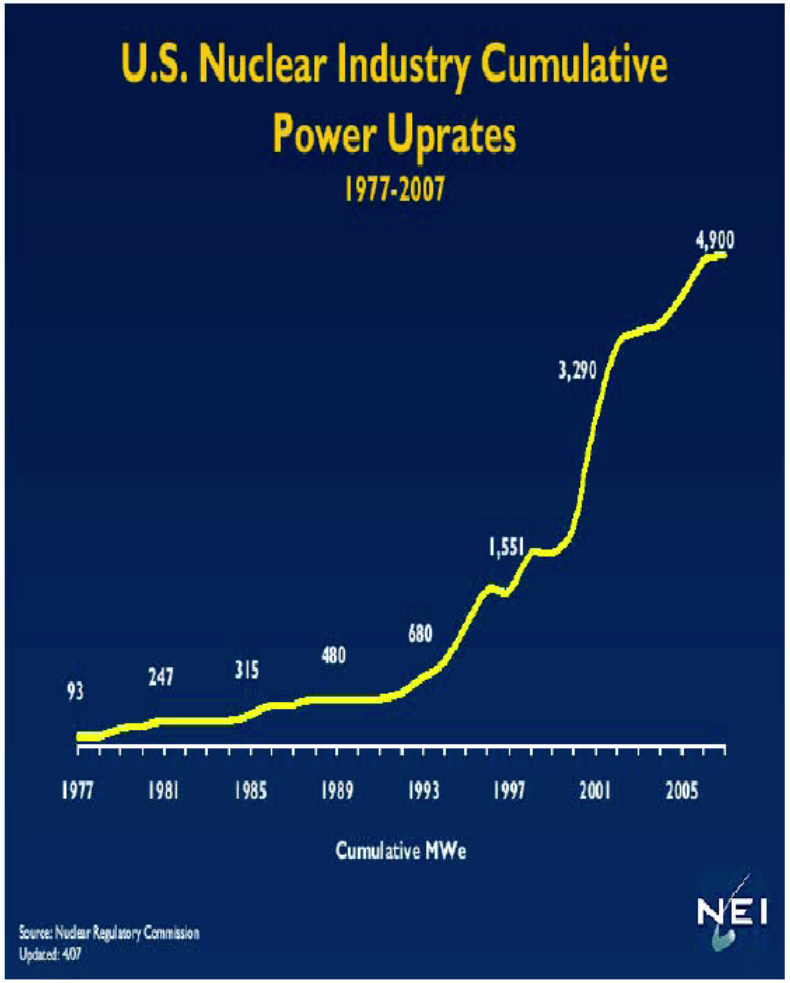
Nuclear Power Economics 101.

- Annual savings in bus-bar electrical energy cost for 1% power uprate of one 1000 MWe LWR (assuming annual capital and O&M costs fixed) = $3.55M/year
  - *OR*
- A 0.3% power uprate of one 1000 MWe LWR results in bus-bar electric energy cost savings of $1M/year

- Annual savings in bus-bar electrical energy cost for 1% reduction in fuel cost for one 1000 MWe LWR = $0.36M/year
  - *OR*
- A 2.8% reduction in fuel cost for one 1000 MWe LWR results in bus-bar electric energy cost savings of $1M/year

How are power uprates achieved?

- Modification of nuclear fuel design to extract higher power with the same thermal margins. design margin,e.g. BWR 7x7 lattice => 10x10 lattice = 15-20% power uprates
- The power uprates in the last 3 decades amount to about 5 new reactors!



Stochastic finite element method

- A surrogate model is an explicit approximation $\Psi(\alpha)$ in some basis $J \approx \hat{J} = \sum_q x_q \Psi_q$

- Stochastic Finite Element Method (SFEM):

- Choose a set of multi-variable orthogonal polynomials Ψ. Use some subset $\{\Psi_q\}$ to approximate the output function:

$$\int_\Omega \Psi_p \Psi_q d\pi = 0 \quad p \neq q$$

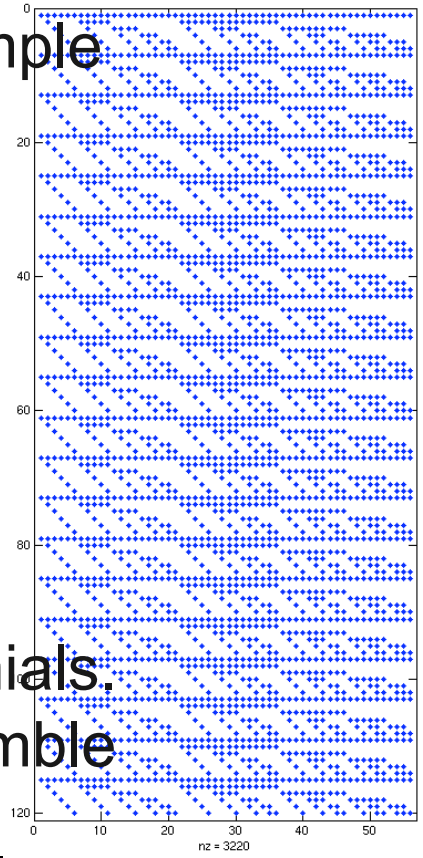
- The coefficients P in the definition of each polynomial are chosen to satisfy the orthogonality condition in some measure π:

- For Gaussian probability measure,

SFEM: Derivative-based Regression

- Collocation procedure: evaluate the basis polynomials at the sample points in the parameter space, run full model to compute the outputs S at the sample points, assemble the collocation system $\Psi x = S$:

$$\begin{pmatrix} \Psi(S_1) \\ \Psi(S_2) \\ \vdots \\ \Psi(S_m) \end{pmatrix} X = \begin{pmatrix} J(S_1) \\ J(S_{2)} \\ \vdots \\ J(S_m) \end{pmatrix}$$

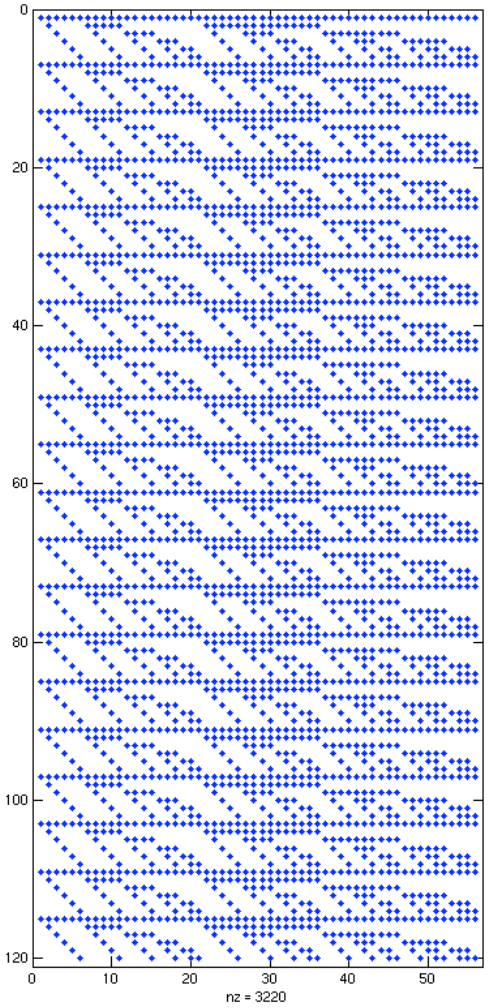


- Issue: we would like to use high-order polynomials. The number of sample points required to assemble Ψ grows rapidly.

- Suggestions:

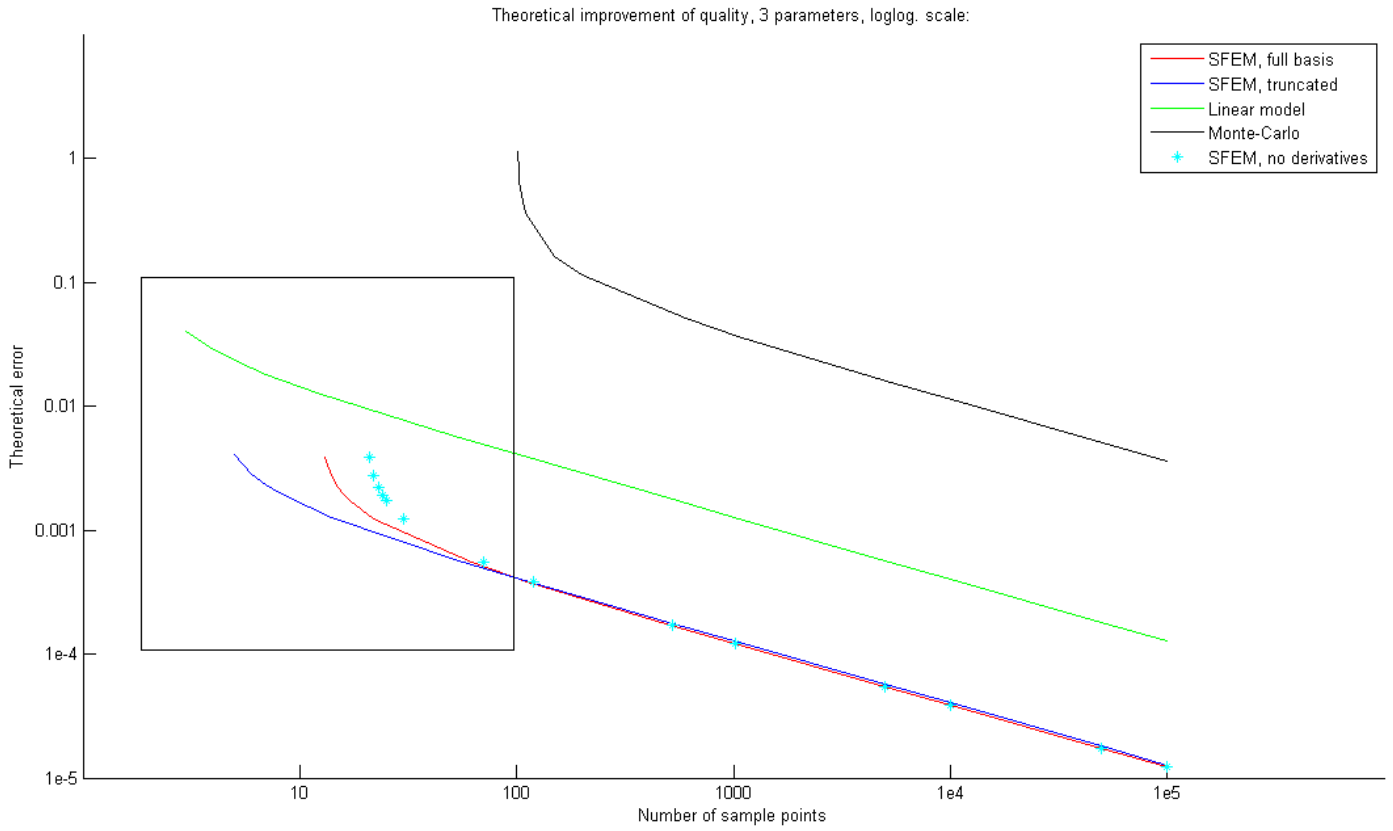
Using Derivatives

$$\begin{pmatrix} \Psi(S_1) \\ \frac{\partial}{\partial \alpha}\Psi(S_1) \\ \Psi(S_2) \\ \frac{\partial}{\partial \alpha}\Psi(S_1) \\ \vdots \\ \Psi(S_n) \\ \frac{\partial}{\partial \alpha}\Psi(S_n) \end{pmatrix} x = \begin{pmatrix} J(S_1) \\ \frac{\partial}{\partial \alpha}J(S_1) \\ J(S_2) \\ \frac{\partial}{\partial \alpha}J(S_1) \\ \vdots \\ J(S_n) \\ \frac{\partial}{\partial \alpha}J(S_n) \end{pmatrix}$$

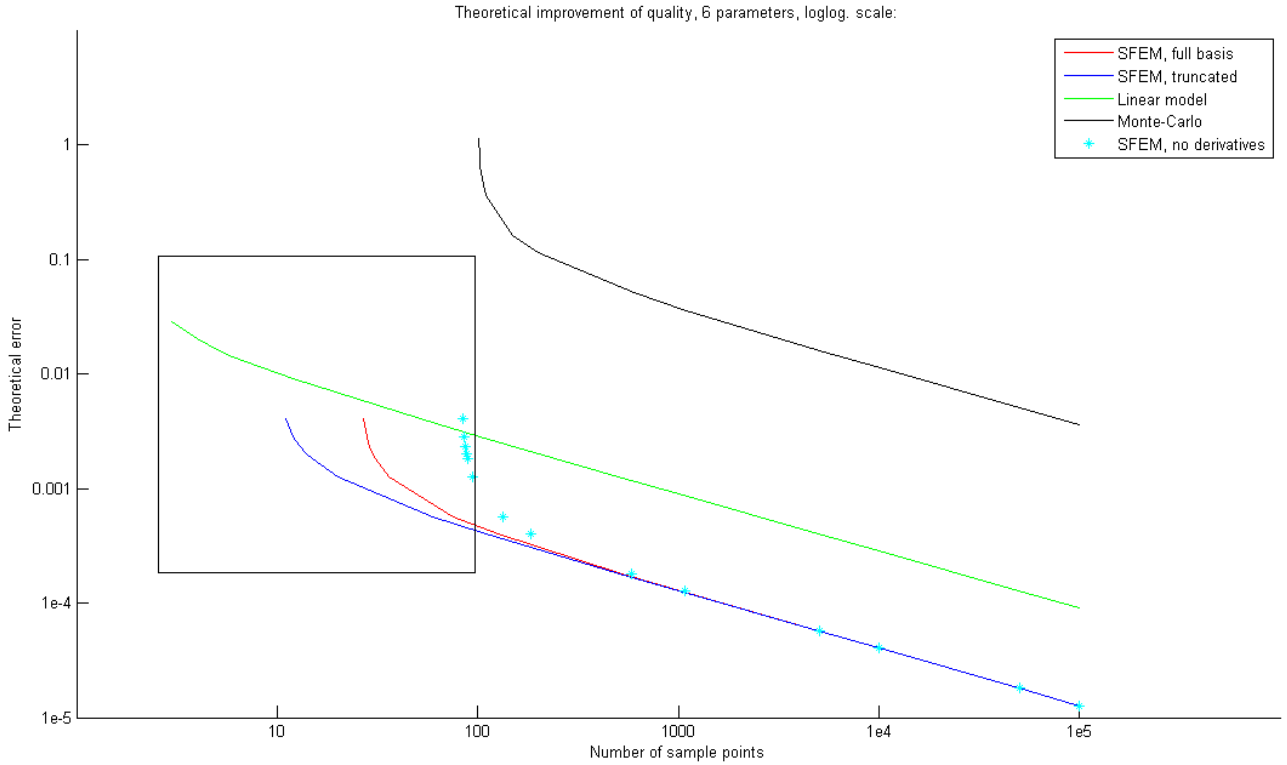


- The only interaction with the physics code is only by the right hand side.
- If we implement the adjoint wisely, we can get NP times more information for not even on extra function evaluation cost !!!

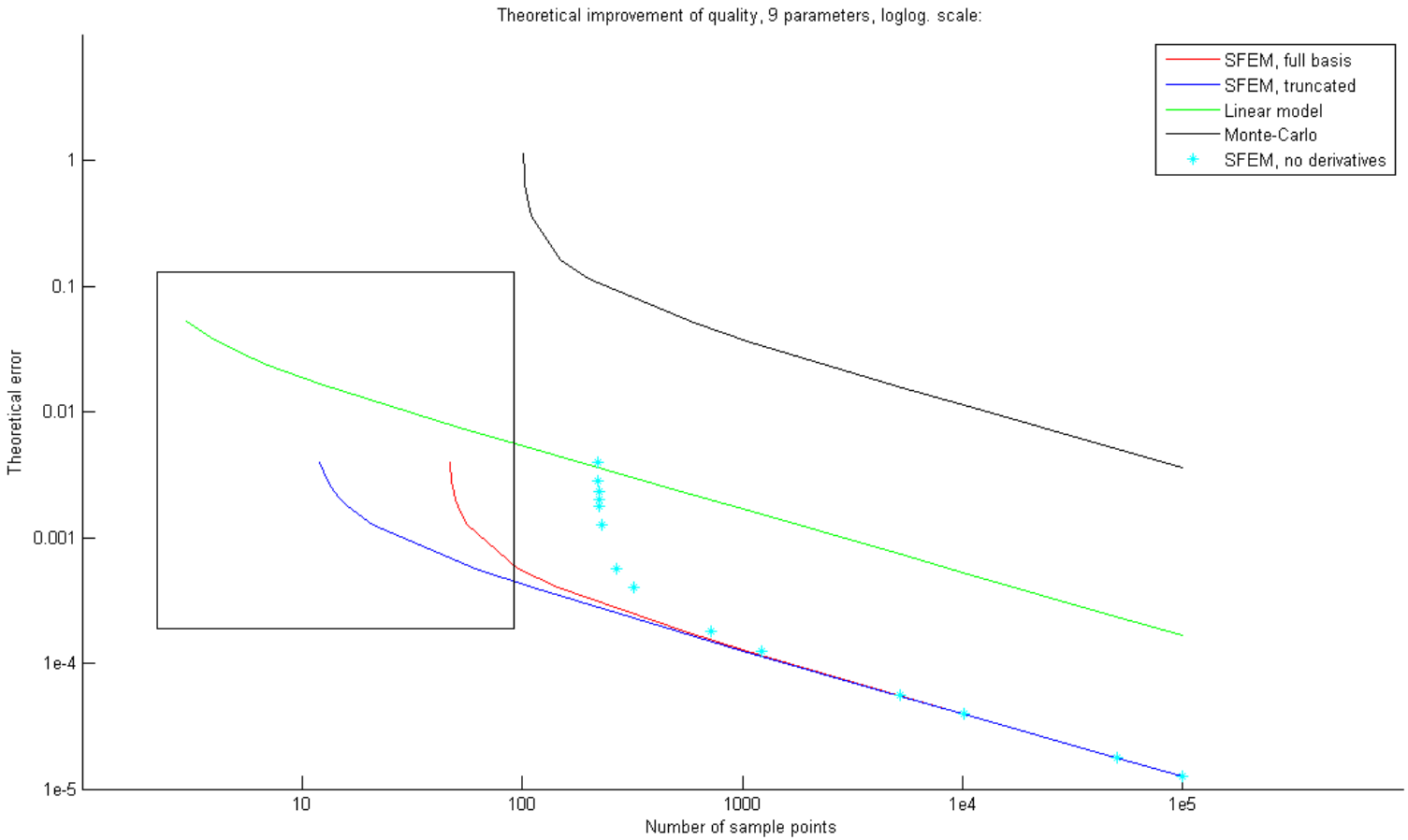
Control Variate Performance 3 Parameters



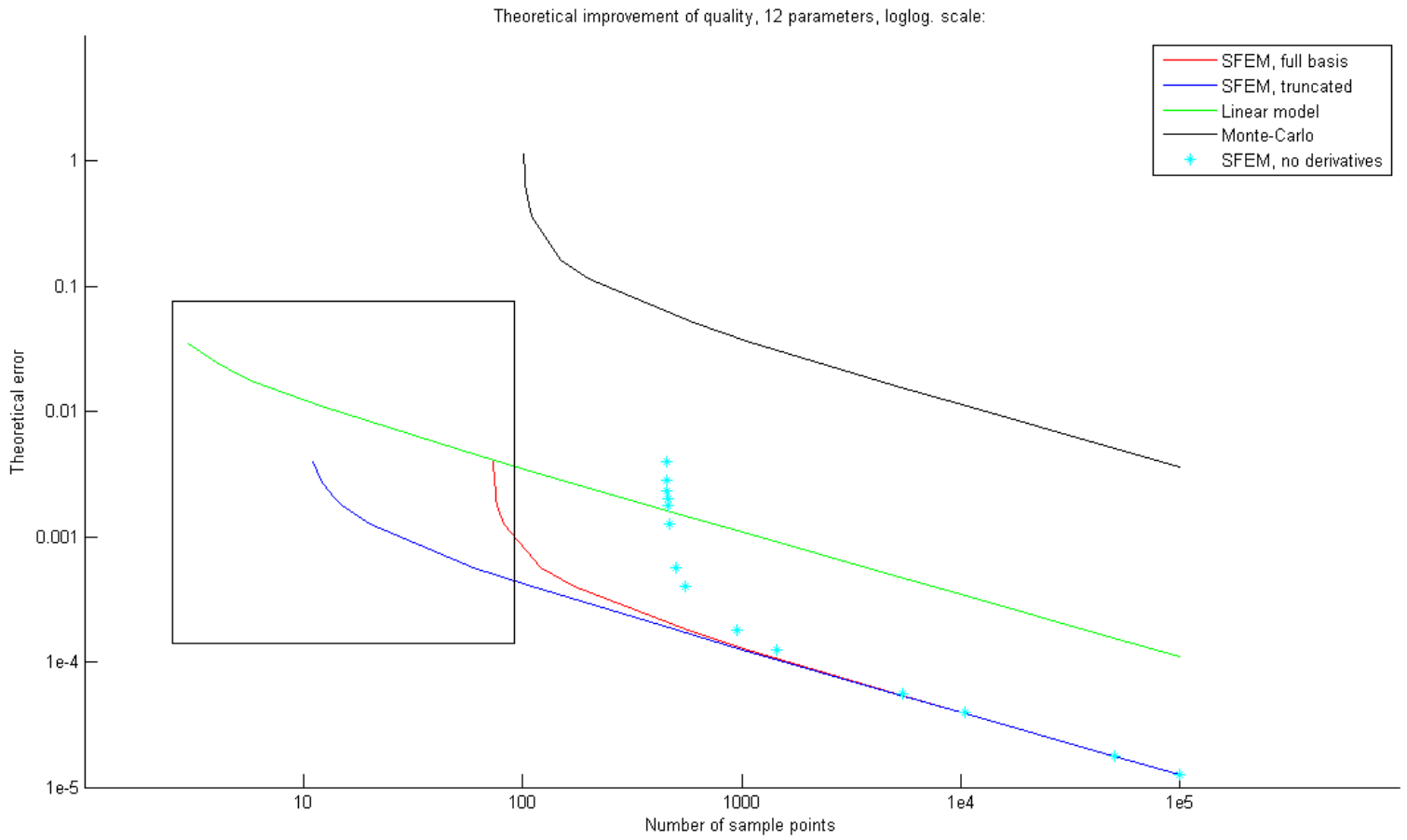
Control Variate Performance, 6 Parameters



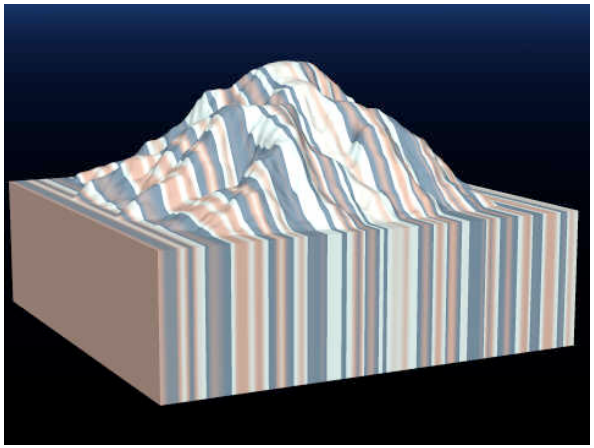
Control Variate, 9 parameters



Control Variate 12 parameters



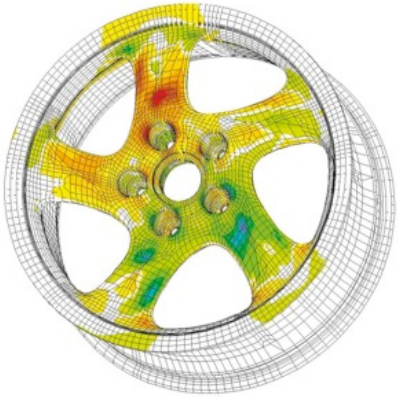
Spatial Uncertainty Calculations



Geostatistics

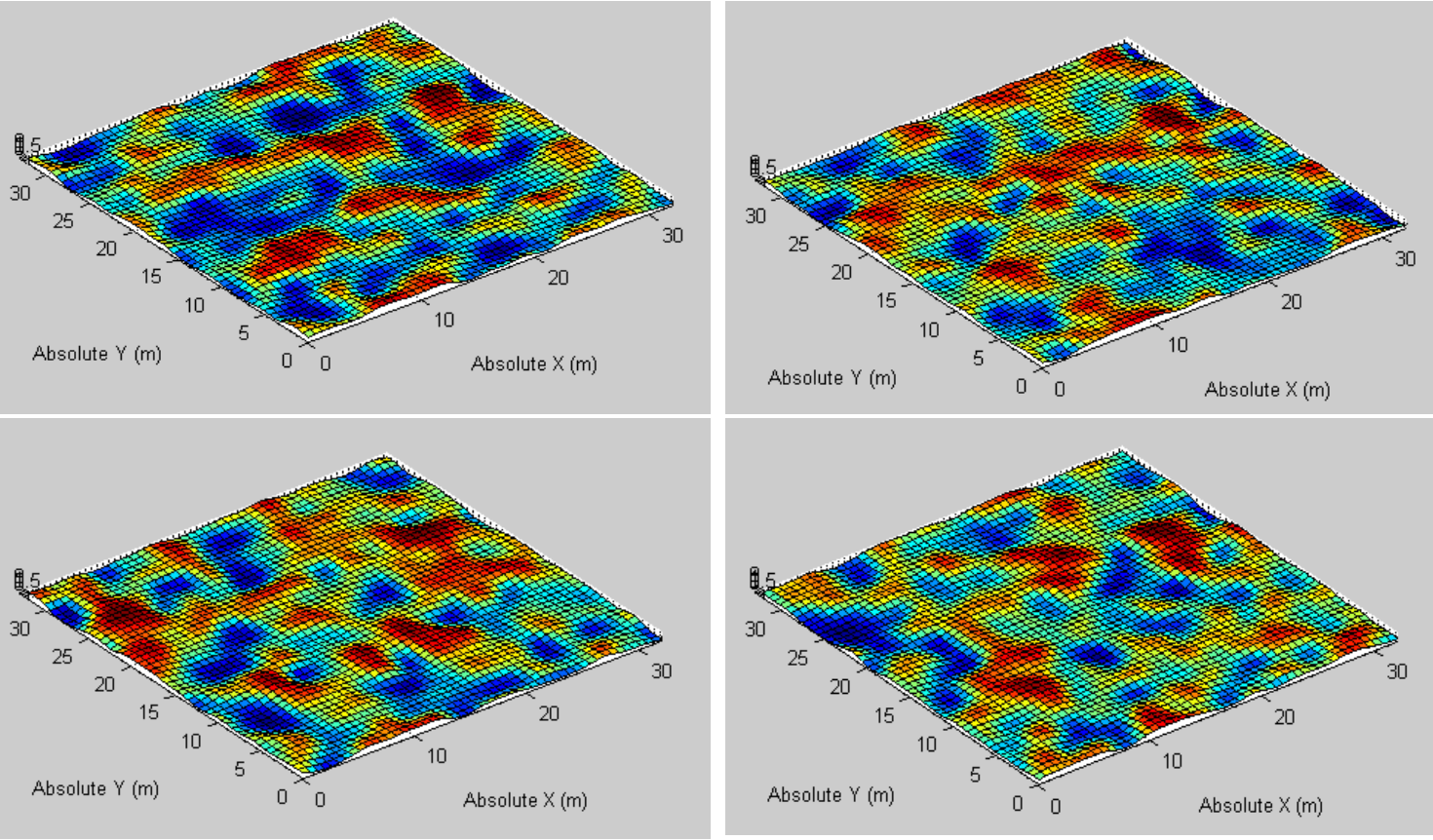
Dynamic Simulations

Meteorology

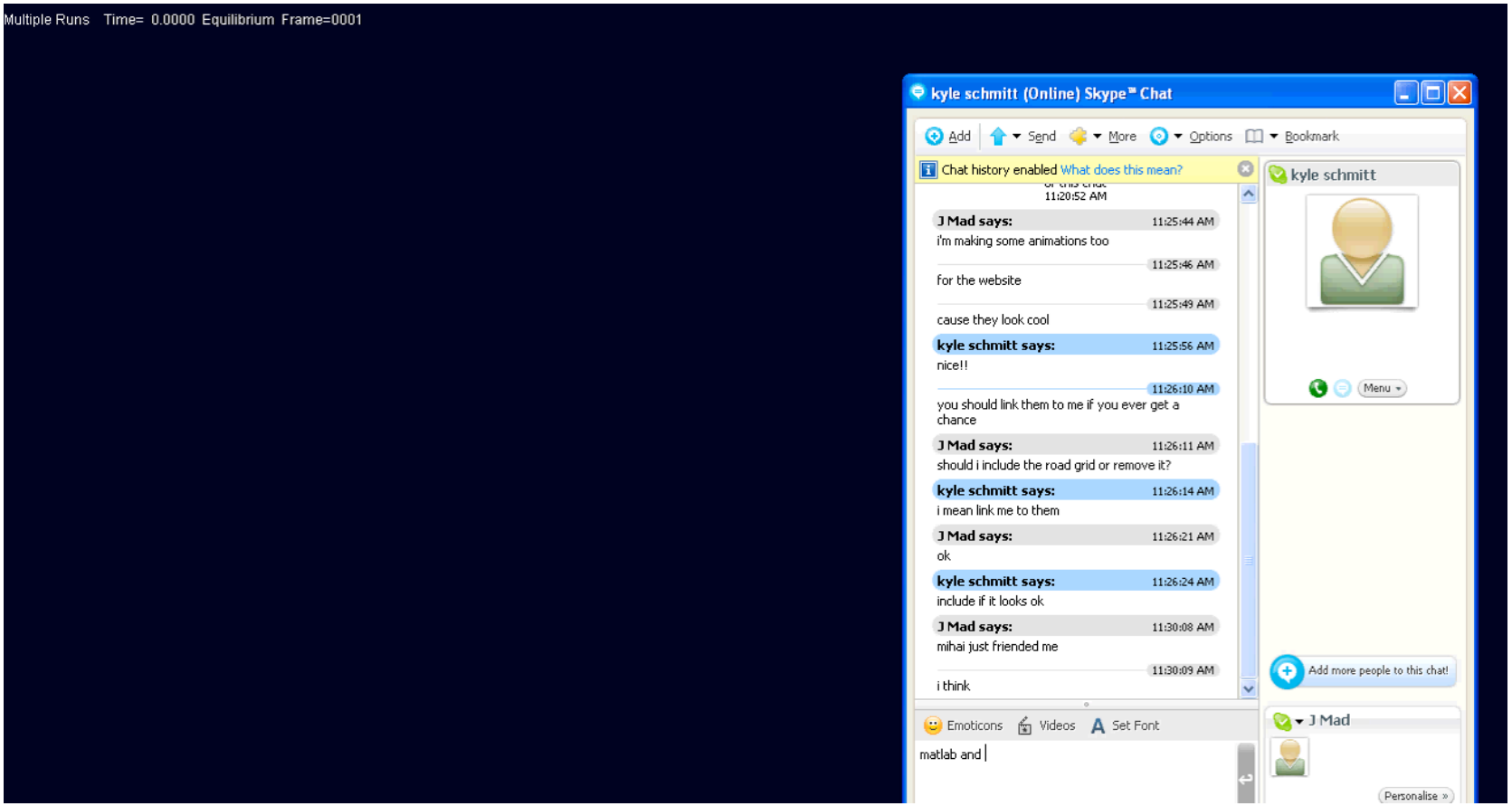


Finite Element Analysis

How is this useful? The input and the output



Vehicle behavior of a car on patches of ice: ADAMS model



Our method is faster and more efficient for storage