SAND2018-13226PE

Solving Real-World Problems with Mathematics and Computing

Michael Parks

Center for Computing Research

Sandia National Laboratories

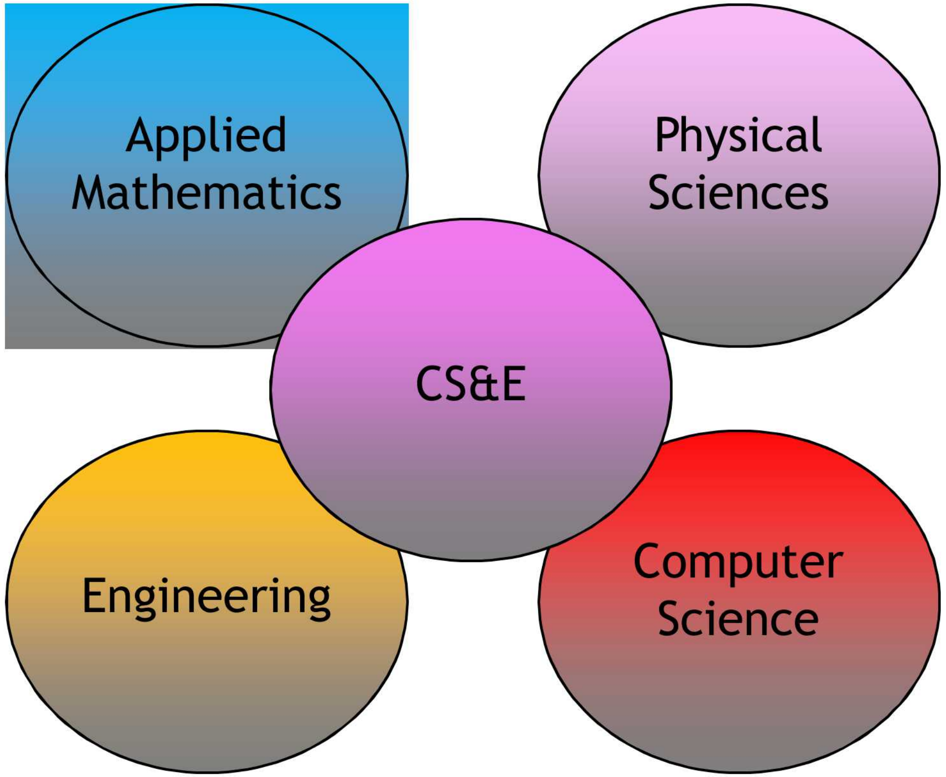
CCR
Center for Computing Research

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND-2018XXXX

What is Computational Science and Engineering?

- CS&E is a broad multidisciplinary area that encompasses applications in science, engineering, applied mathematics, numerical analysis, and computer science.*



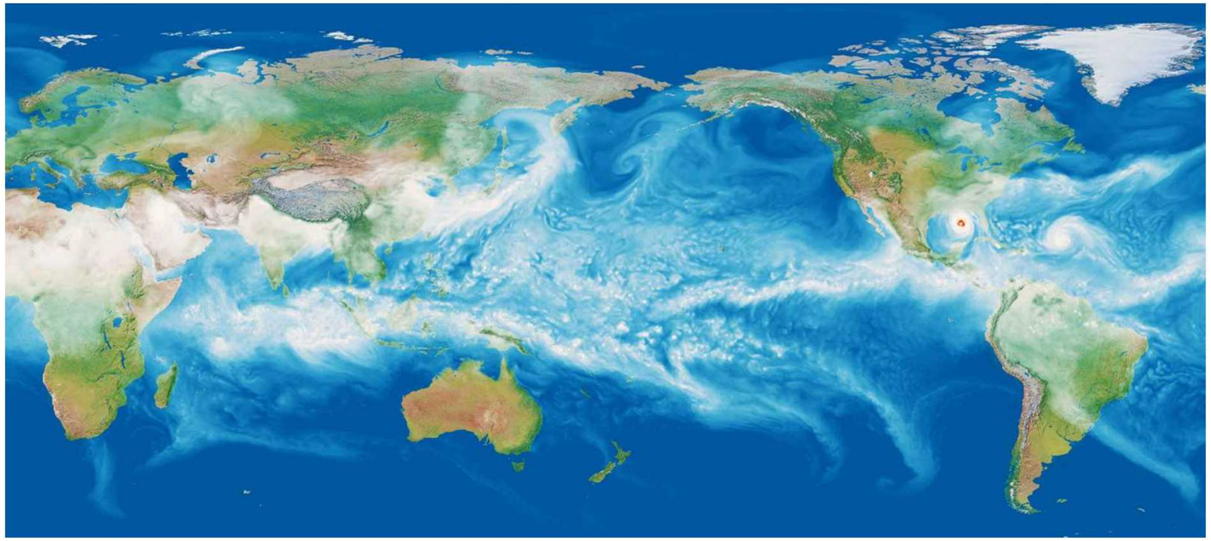
- CS&E is a third pillar of scientific investigation, along with theory and physical experiment
- In general, unified theories of complex multi-physics phenomena do not exist
- In many cases, physical experiments are too expensive or simply impossible

*** Society of Industrial and Applied Mathematics (SIAM)**

CS&E Application Space

Global Climate Modeling

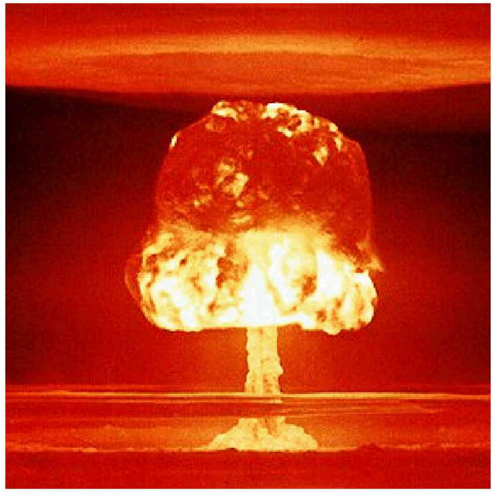
- The Energy Exascale Earth System Model (E^3SM) project is an ongoing, state-of-the-science Earth system modeling, simulation, and prediction project that optimizes the use of DOE laboratory resources to meet the science needs of the nation and the mission needs of DOE.



Cat 5 hurricane simulated by E3SM at 13km resolution. Credit: A. Scott and M. Taylor, SNL

Stockpile Stewardship

- Ensure safety and surety of aging US nuclear weapons stockpile without full scale nuclear weapons testing
- Perform virtual testing of stockpile weapons on supercomputers



Castle Romeo shot, Bikini Atoll. March 27, 1954

*** https://e3sm.org/**

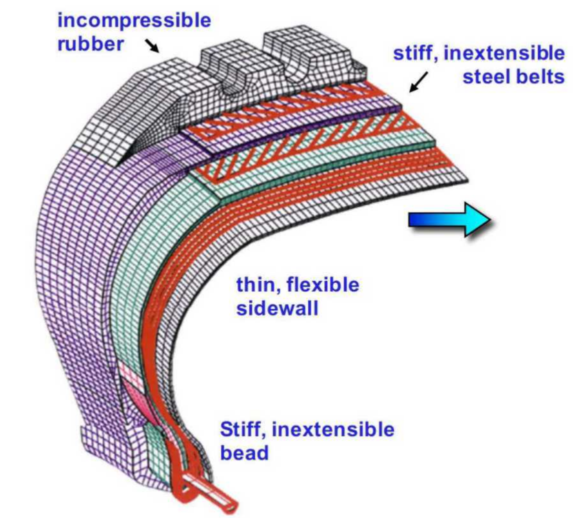
CS&E Application Space

Digital Design

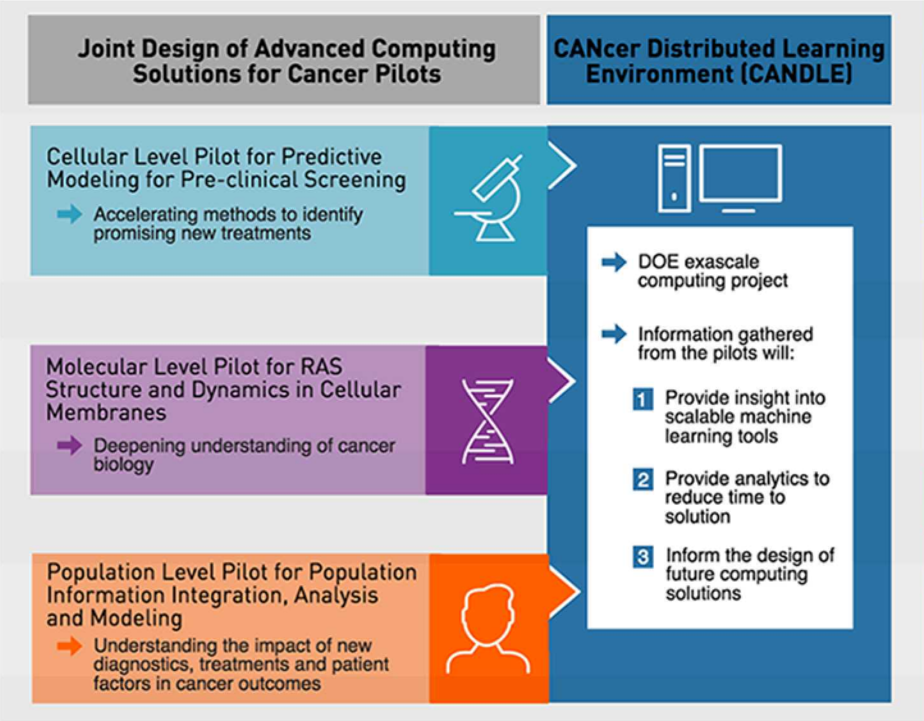
- Finite Element Analysis (FEA) allows numerical representation and virtual modeling of physical structures and processes
- Avoid expensive physical prototyping
- Example: 100% of tire designs at Goodyear are modeled before molds are ordered. Faster development times, better tires, reduced R&D costs
- Joint work with Sandia under CRADA relationship.

Cancer Research**

- Develop precision cancer treatments via supercomputing
- Joint Design of Advanced Computing Solutions for Cancer (JDACS4C) – A DOE and NCI partnership
- Supported by DOE's Exascale Computing Project (ECP) through CANcer Distributed Learning Environment (CANDLE)
- Synergistic with central goal of NCI Cancer Moonshot



Finite Element Model of Typical Tire*



National Cancer Institute and Dept. Of Energy Collaborations

* https://www.sandia.gov/news/publications/labnews/articles/2017/07-07/goodyear.html
** https://cbiit.cancer.gov/ncip/hpc/jdacs4c

CS&E Application Space

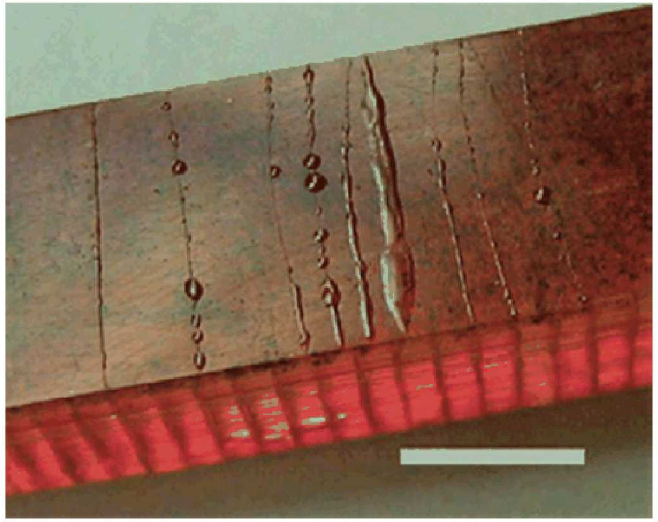
Power Grid[1]

- The power grid is "the world's largest machine"
- New intermittent renewable sources (wind and solar), electric vehicles, and smart loads will vastly change the behavior of the electric power grid
- Optimizing and hardening the grid requires optimization, dynamical systems, stochastic analysis, discrete mathematics, & scalable algorithms at a Grand Challenge scale

North American Power Grid

Materials Design[2]

- Directed synthesis of material with properties we dictate based on first-principles computational exploration
- Holy grail: Reverse design of materials
- Start with set of desired properties, then use mathematical models and computation to predict atom types and arrangements needed to obtain desired properties



Self-healing polymer that repairs cracks[1]

[1] https://www.exascaleproject.org/researchareas/power-grid/

[2] J. Hemminger, *et. al.*, Directing Matter and Energy: Five Challenges for Science and the

What's Involved in CS&E?

What do you need to know to do CS&E?

- Application expertise
- Physical laws
- Mathematical models
- Numerical models
- Solution algorithm
- Software implementation (High-Performance Computing)
- Program execution
- Analysis and visualization of results

This includes (but is not limited to)

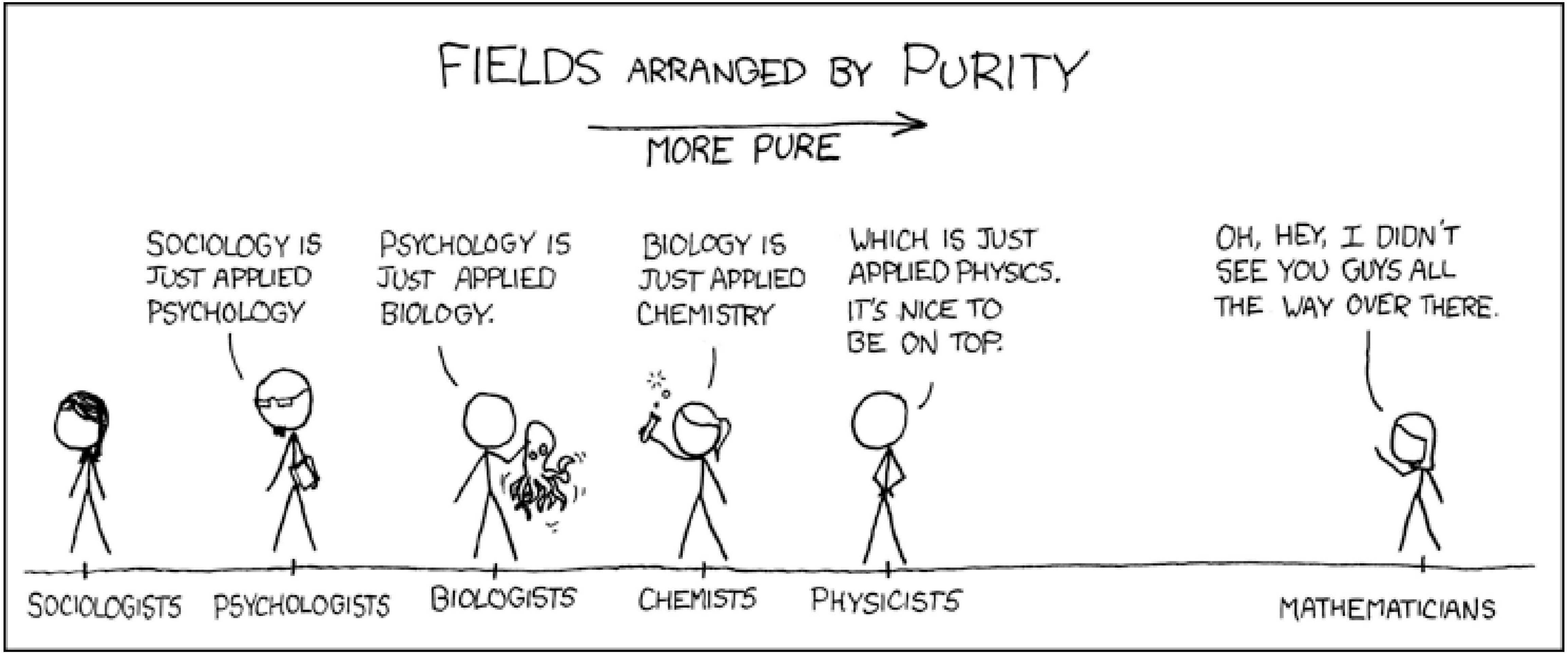
- Mathematical analysis
- Numerical analysis (accuracy, stability, convergence, etc.)
- Computational Science
- Verification & Validation
- Uncertainty quantification
- Machine learning
- Hardware-aware algorithms

Mathematics (of different kinds) involved in all of these

In general, one person doesn't know all of these things. We work in groups.

At this point, it's useful to explore the relationships between the fields that make up CS&E, and how they utilize mathematics.

Relating Mathematics & the Physical Sciences[1]

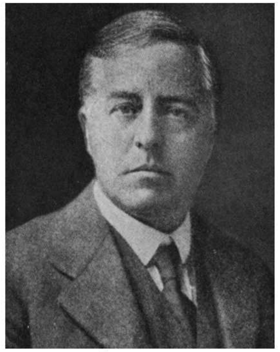


This suggests that the physical and social sciences are really just applications of mathematics…

[1] "Purity". xkcd. xkcd.com/435/.

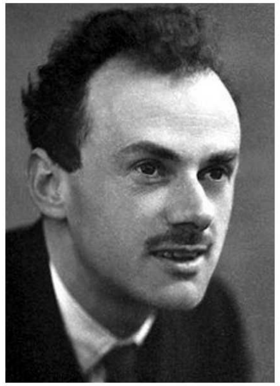
So Is Physical Science Really Just Mathematics?

Sir James Jeans wrote, "… from the intrinsic evidence of his creation, the Great Architect of the Universe now begins to appear as a pure mathematician."[1]



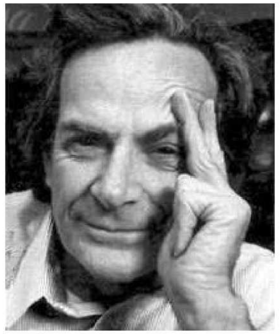
Jeans

Paul Dirac wrote, "It seems to be one of the fundamental features of nature that fundamental physical laws are described in terms of a mathematical theory of great beauty and power, needing quite a high standard of mathematics for one to understand it."[2]



Dirac

Richard Feynman wrote, "Physicists cannot make a conversion to any other language. If you want to learn about nature, to appreciate nature, it is necessary to understand the language she speaks in. She offers her information only in one form; we are not so unhumble as to demand that she change before we pay any attention."[3]

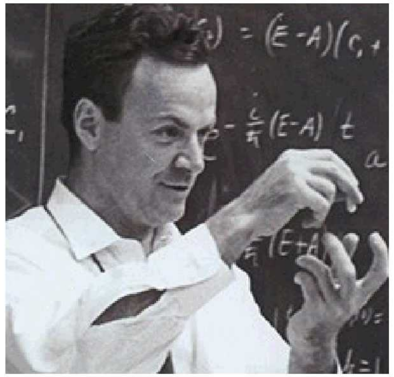


Feynman

[1] J. Jeans, The Mysterious Universe, Cambridge University Press, 1930.
[2] P.A.M. Dirac. "The Evolution of the Physicist's Picture of Nature", Scientific American, May 1963.
[3] R.P. Feynman, The Character of Physical Law, The MIT Press, 1965.

So Is Physical Science Really Just Mathematics?

Feynman continued, "It always bothers me that, according to the laws as we understand them today, it takes a computing machine an infinite number of logical operations to figure out what goes on in no matter how tiny a region of space, and in no matter how tiny a region of time. How can all that be going on in that tiny space? Why should it take an infinite amount of logic to figure out what one tiny piece of space/time is going to do? So I have often made the hypothesis that ultimately physics will not require a mathematical statement, that end the end the machinery will be revealed, and the laws will turn out to be simple…"[1]



Feynman

Mathematics is the language we must speak in order to do science …
… but the laws of nature have a reality of their own, independent of their mathematical description.

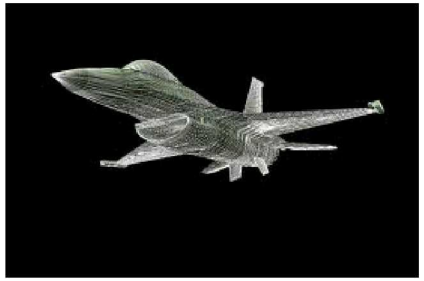
This important realization informs us that our mathematical equations merely *describe* nature. There may be multiple valid mathematical models, each of which may have different limits to their validity.

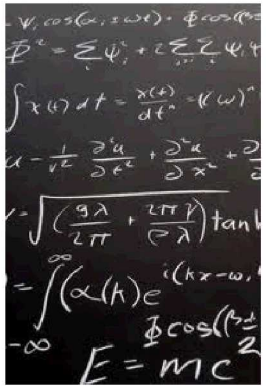
Now, let's explore the CS&E, from beginning to end…

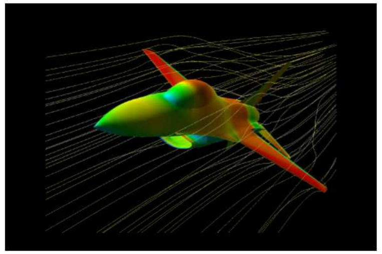
[1] R.P. Feynman, The Character of Physical Law, The MIT Press, 1965.

What is Computational Science and Engineering?

Application → Physical Law → Mathematical Model → Numerical Model → Solution Algorithm → Software Implementation → Program Execution → Analysis & Viz Of Results

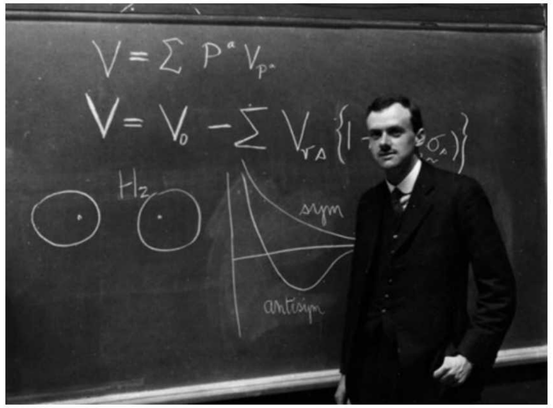






Application Drivers

Application



Applications provide drivers for research in CS&E

- Scientific understanding and investigation
- Programmatic/industrial need
- Provide guidance for policy makers facing critical decisions

Example: Understanding climate change

- Do we need to alter the way we generate & use energy?
- How?
- On what timetable?
- What are the consequences if we don't?

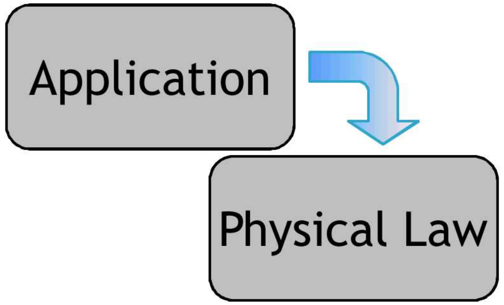
Scientific computation is being used to model climate change

- Do we believe our computational models?
- What are the error bars?
- Are we confident enough yet on our climate simulations to make public policy decisions using them?

It's important to be able to clearly communicate the importance of your research results and their real-world implications.

***Dirac, during interview*: My work has no practical significance.**
***Journalist*: But might it have?**
***Dirac*: That I do not know. I don't think so.**

Physical Law



What physics is relevant?

Mechanics

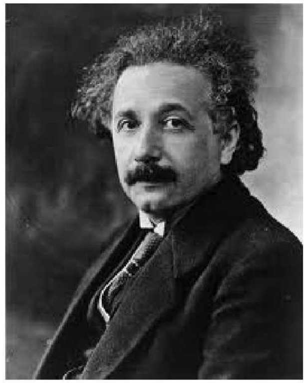
- Solids, Fluids, Gasses
- Thermal
- Radiation
- Relativity
- Quantum mechanics

What physics can we ignore?

- Approximate or ignore specific physics
- Approximate properties as constant
- Ignore fast/slow dynamics? (i.e., ignore some time scales)
- Ignore micro/macroscale (i.e., ignore some length scales)

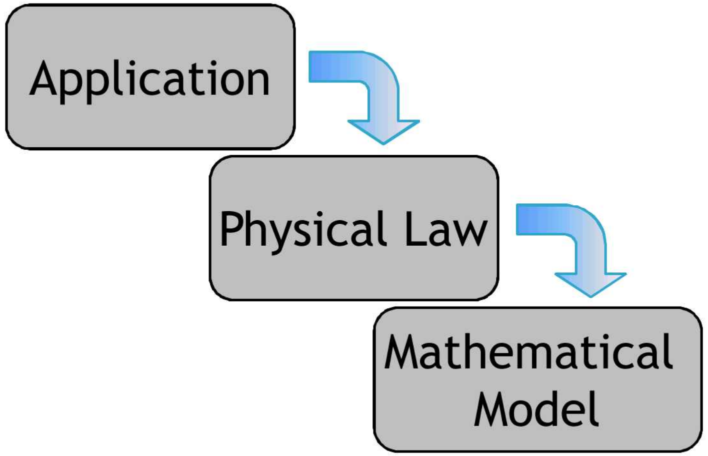
Why do we want to ignore some physics?

- We seek the simplest physical model describing the phenomena of the application of interest
- Unneeded physics adds complexity, complication, and computational modeling burden



"Everything should be made as simple as possible, but no simpler." – Albert Einstein

Mathematical Models



"In mathematics you don't understand things. You just get used to them."
- John von Neumann

What mathematical representation do we use?

Discrete vs. continuous

- Which best represents physics?

Governing Equations

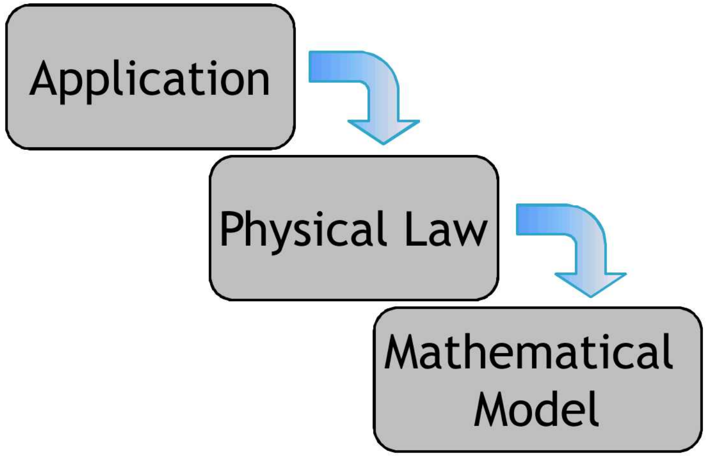
- Ordinary differential equations
- Partial differential equations
- Integral equations (Fredholm, Volterra, other)
- Stochastic PDEs
- Differential algebraic equations (DAEs)
- Delay differential equations (DDEs)

How to couple together governing equations representing different physics?

How to handle interacting bodies? (e.g., fluid-structure)

How to handle multi-physics systems with many governing equations?

Mathematical Models



In general, you must make several decisions when constructing a mathematical model describing a physical system.

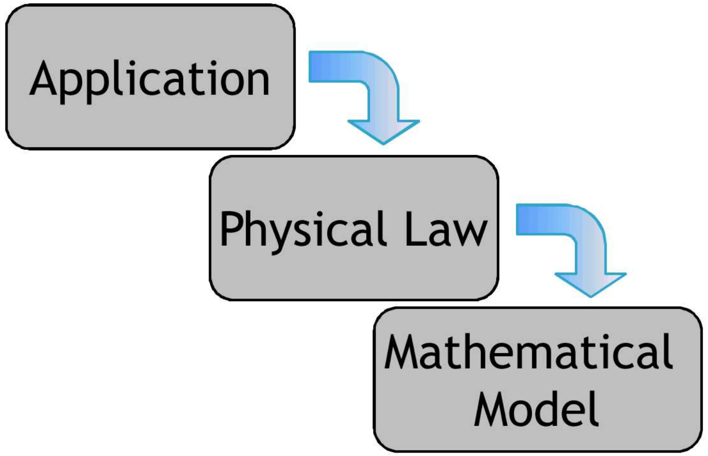
The choice of a specific mathematical model can make your life very hard, or very easy.

- Remember Feynman's conjecture

Don't believe anyone who tells you there's only one mathematical model for any given physical law or application, and that everything else is wrong.

"All models are wrong, but some are useful." – George Box

Mathematical Models



Linear or nonlinear equations?

We divide our models into “linear” and “nonlinear” models.

Why?

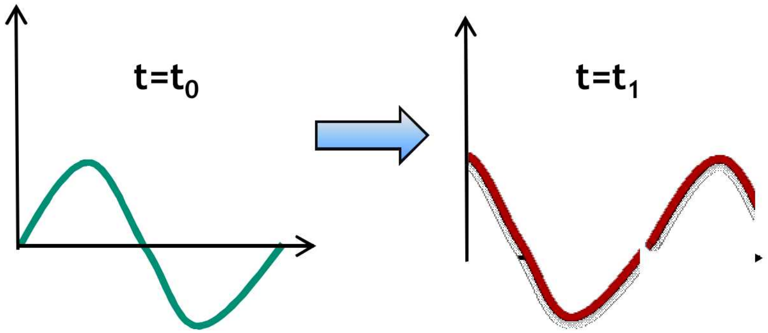
- **We’re good at solving linear equations.**
- **We’re not nearly as good at solving nonlinear equations.**

But virtually all of nature is described by nonlinear equations. Our linear equations are approximations.

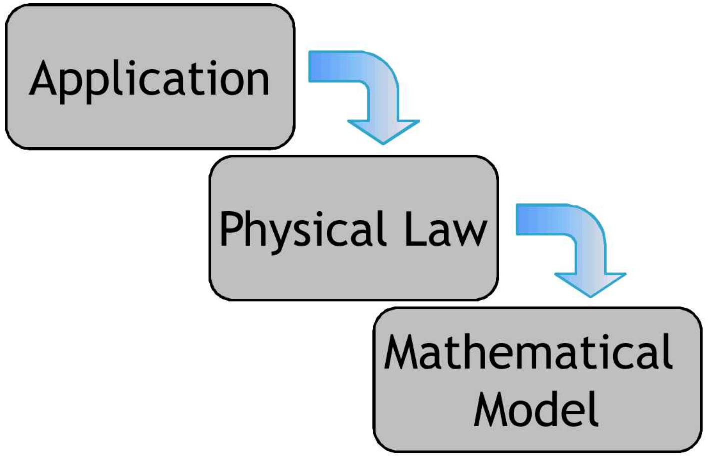
“Using a term like nonlinear science is like referring to the bulk of zoology as the study of non-elephant animals.” – Stanislaw Ulam

- **Example: Wave equation vs. Burgers’ Equation**

$$\frac{\partial u(x,t)}{\partial t} + c\frac{\partial u(x,t)}{\partial x} = 0$$



Mathematical Models



Linear or nonlinear equations?

We divide our models into "linear" and "nonlinear" models.

Why?

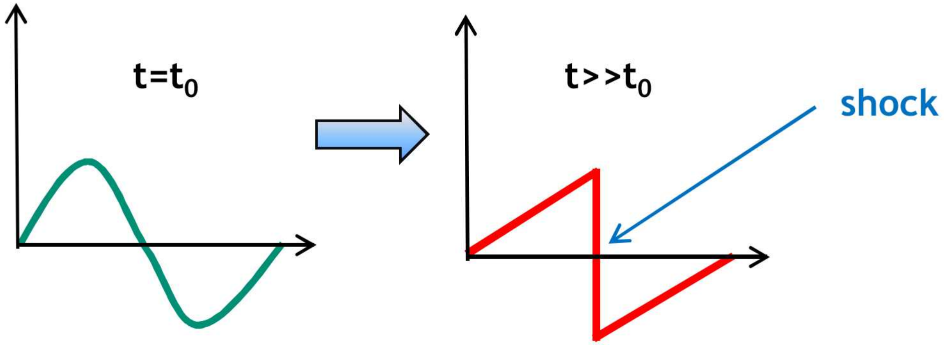
- **We're good at solving linear equations.**
- **We're not nearly as good at solving nonlinear equations.**

But virtually all of nature is described by nonlinear equations. Our linear equations are approximations.

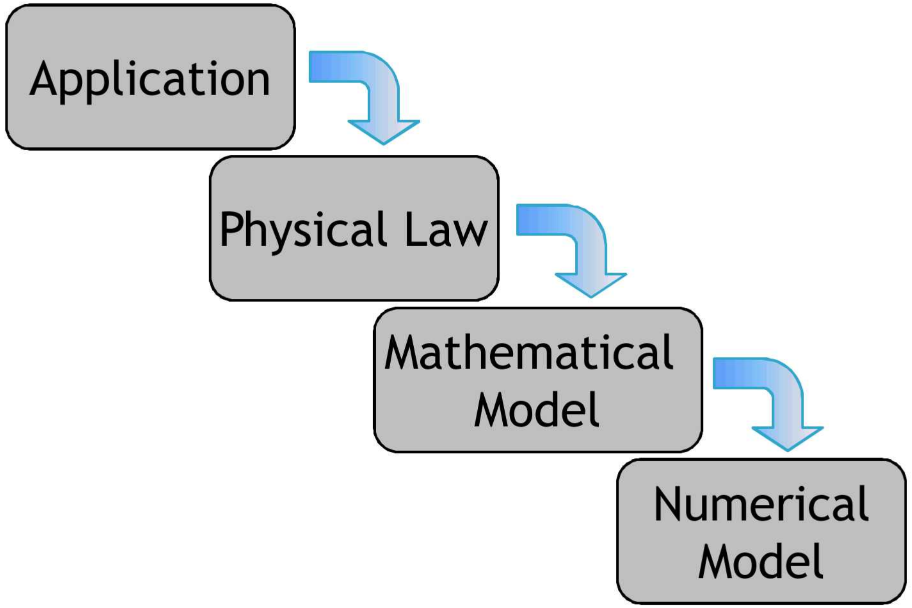
"Using a term like nonlinear science is like referring to the bulk of zoology as the study of non-elephant animals." – Stanislaw Ulam

- **Example: Wave equation vs. Burgers' Equation**

$$\frac{\partial u(x,t)}{\partial t} + u(x,t)\frac{\partial u(x,t)}{\partial x} = 0$$



Numerical Models



How do we put mathematical models on a computer?

We must discretize our continuum models.
We must find a suitable representation of our discrete models.

Spatial discretization

- Finite differences
- Finite elements
- Finite volumes

  …

Temporal discretization

- Explicit (CFL timestep limited)
- Implicit (must solve coupled system)
- Forward/backward Euler
- Runge-Kutta Methods
- IMEX

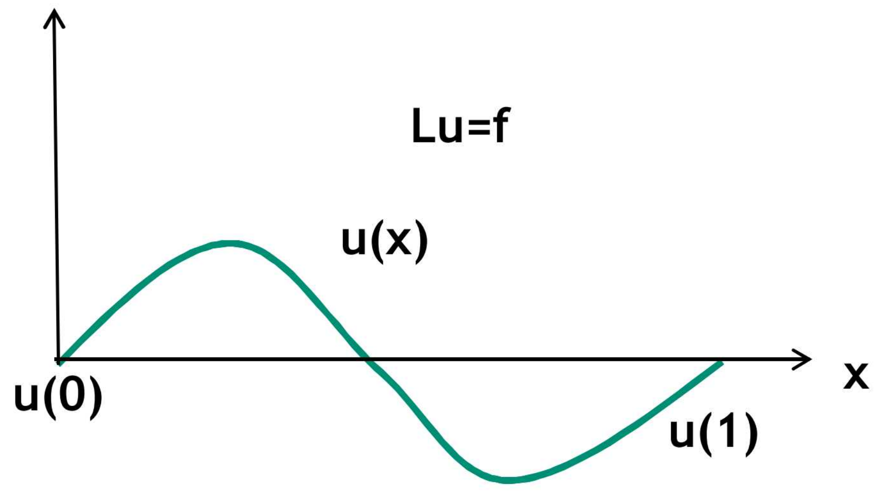
  …

"Think. Then discretize." – Vladimir Rokhlin

Finite Differences – A Simple Example

1D diffusion equation with variable diffusivity k(x):

$$-\frac{d}{dx}\left(k(x)\frac{du(x)}{dx}\right) = f(x) \quad \text{in } (0,1)$$
$$u(0) = u(1) = 0$$

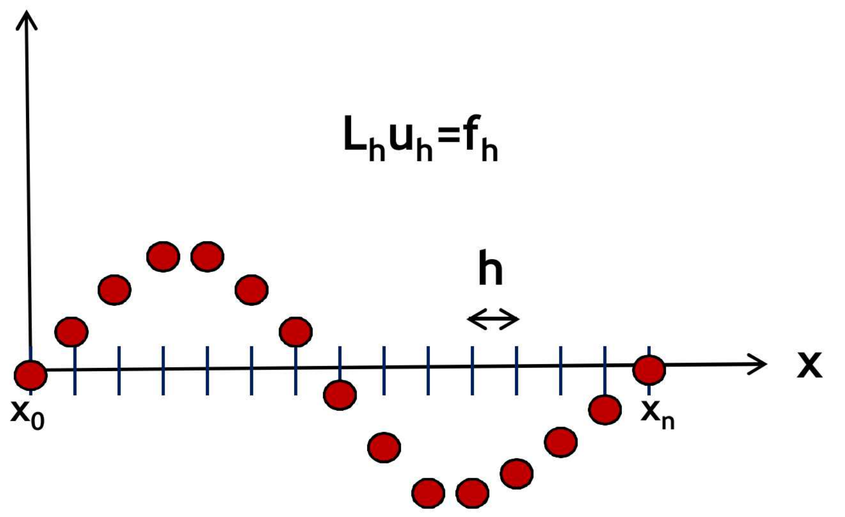


Apply chain rule:

$$-k(x)\frac{d^2u(x)}{dx} - \frac{dk(x)}{dx}\frac{du(x)}{dx} = f(x) \quad \text{in } (0,1)$$
$$u(0) = u(1) = 0$$

Replace x with a set of discrete points (mesh)
$x_0, x_1, \ldots, x_{n+1}$
Denote numerical solution at mesh points as $u_i := u(x_i)$

Approximate derivative operators:



$$\frac{du(x_i)}{dx} \approx \frac{u_{i+1} - u_{i-1}}{2h}$$

$$\frac{d^2u(x_i)}{dx^2} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

Finite Differences – A Simple Example

Set of linear equations, one for each mesh point

$$-k_i \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \left(\frac{dk}{dx}\right)_i \frac{u_{i+1} - u_{i-1}}{2h} = f_i \quad i=1,\ldots,n$$

$$u_0 = u_n = 0$$

Write as linear system **Au = f**, where

- **A** is a matrix
- **u**,**f** are vectors

Solve for **u**:

$$\frac{1}{h^2}\begin{bmatrix} 2k_1 & -k_1 + \frac{h}{2}\left(\frac{dk}{dx}\right)_1 & & \\ -k_2 - \frac{h}{2}\left(\frac{dk}{dx}\right)_2 & 2k_2 & -k_2 + \frac{h}{2}\left(\frac{dk}{dx}\right)_2 & \\ & \ddots & \ddots & \ddots \\ & & & \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix}$$

There's a problem -- A is nonsymmetric!

The original operator is Hermitian (self-adjoint), but A isn't.

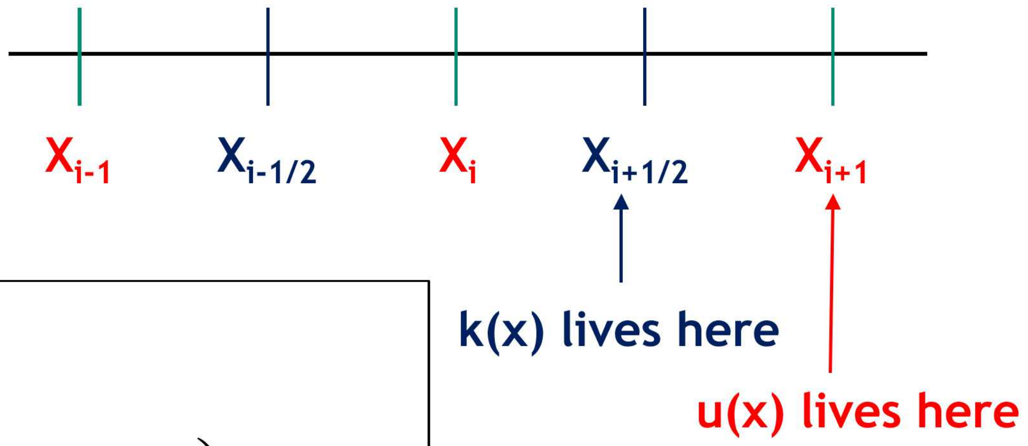
This means we haven't been very smart with our discretization. Let's try again….

Finite Differences – A Simple Example

Discretize original equation directly using offset grid (respect physics)

$$-\frac{d}{dx}\left(k(x)\frac{du(x)}{dx}\right) = f(x) \quad \text{in } (0,1)$$

$$u(0) = u(1) = 0$$



$$k(x_{i+1/2})\frac{du(x_{i+1/2})}{dx} \approx k_{i+1/2}\frac{u_{i+1}-u_i}{h}$$

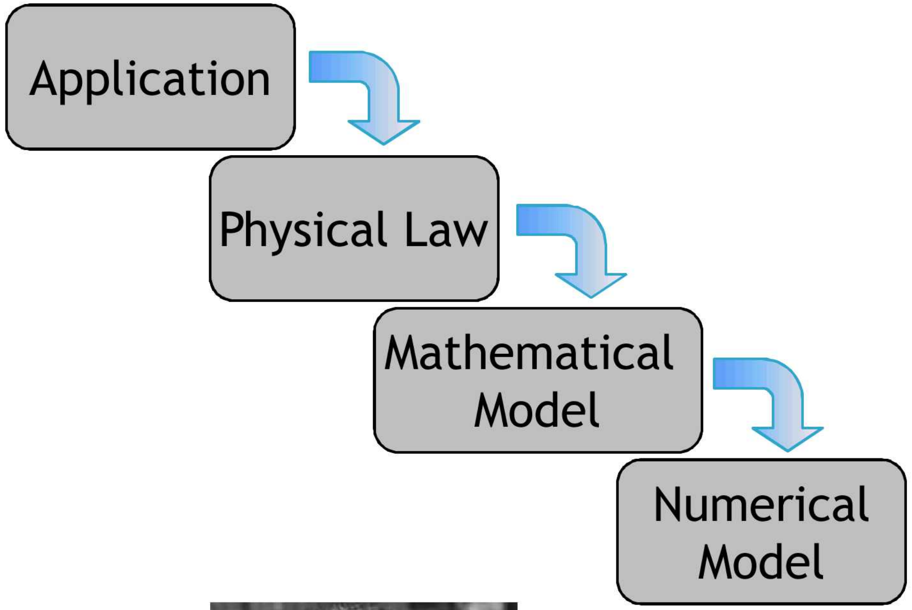
$$\frac{d}{dx}\left(k(x_{i+1/2})\frac{du(x_{i+1/2})}{dx}\right)_{x_i} \approx \frac{1}{h}\left(k_{i+1/2}\frac{u_{i+1}-u_i}{h} - k_{i-1/2}\frac{u_i-u_{i-1}}{h}\right)$$

$$= \frac{1}{h^2}\left(k_{i-1/2}u_{i-1} - \left(k_{i-1/2}+k_{i+1/2}\right)u_i + k_{i+1/2}u_{i+1}\right)$$

This gives a symmetric linear system!

$$\frac{1}{h^2}\begin{bmatrix} (k_{1/2}+k_{3/2}) & -k_{3/2} & & \\ -k_{3/2} & (k_{3/2}+k_{5/2}) & -k_{5/2} & \\ & \ddots & \ddots & \ddots \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix}$$

Numerical Models



"… the theory of difference equations is a rather sophisticated affair, more sophisticated than the corresponding theory of partial differential equations."
- Peter Lax

Let

- $E^h = U^h - \hat{U}^h$, where $\hat{U}$ is exact solution
- $\tau^h = A\hat{U} - F$
- $A^h E^h = -\tau^h$

Is our discretization:

Consistent?

- Does $\|\tau^h\| \to 0$ as $h \to 0$?

Stable?

- Is $\|(A^h)^{-1}\| < C \to$ for all $h < h_0$?

Convergent?

- Does $\|E^h\| \to 0$ as $h \to 0$?

Lax Equivalence Theorem:

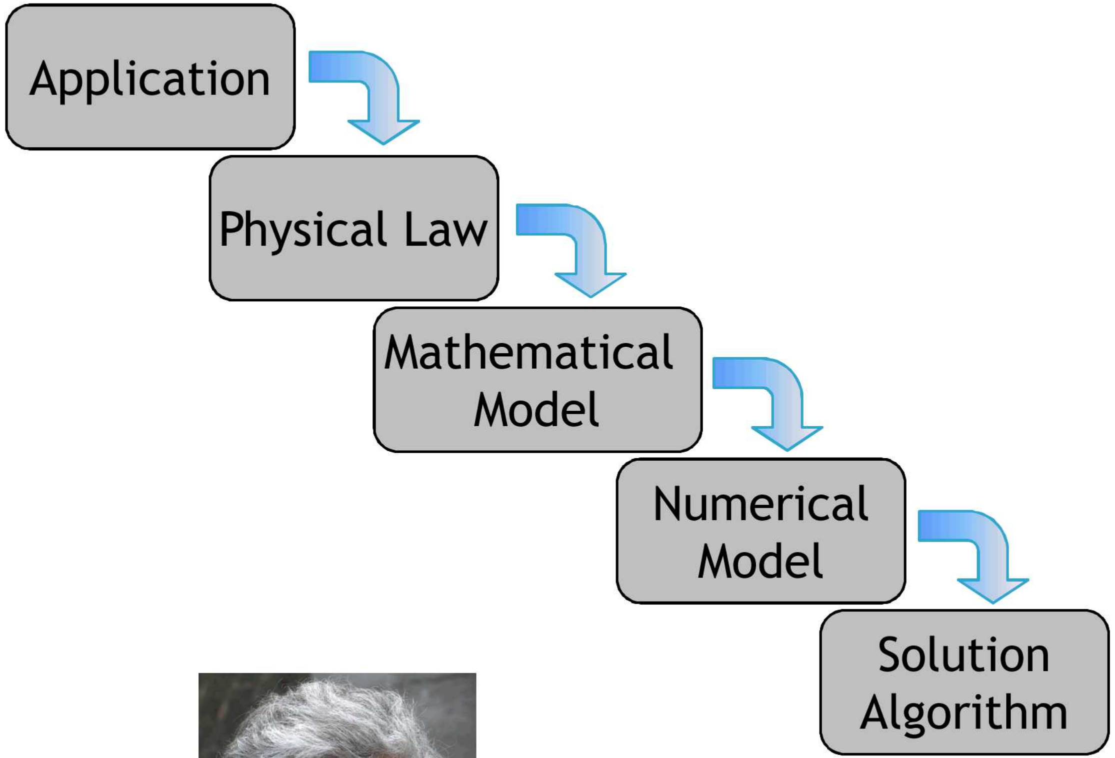
consistency + stability $\to$ convergence

In context of current problem,

- $\|E^h\| = \|(A^h)^{-1}\tau^h\|$
  $\leq \|(A^h)^{-1}\|\|\tau^h\|$
  $\leq C\,h^2$

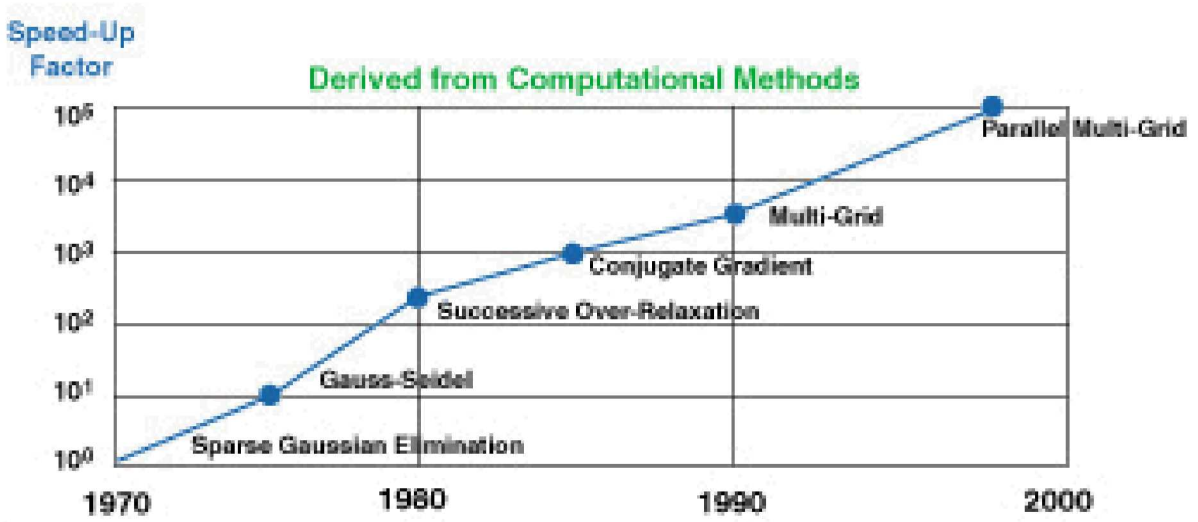
  for all $h < h_0$, $h \to 0$

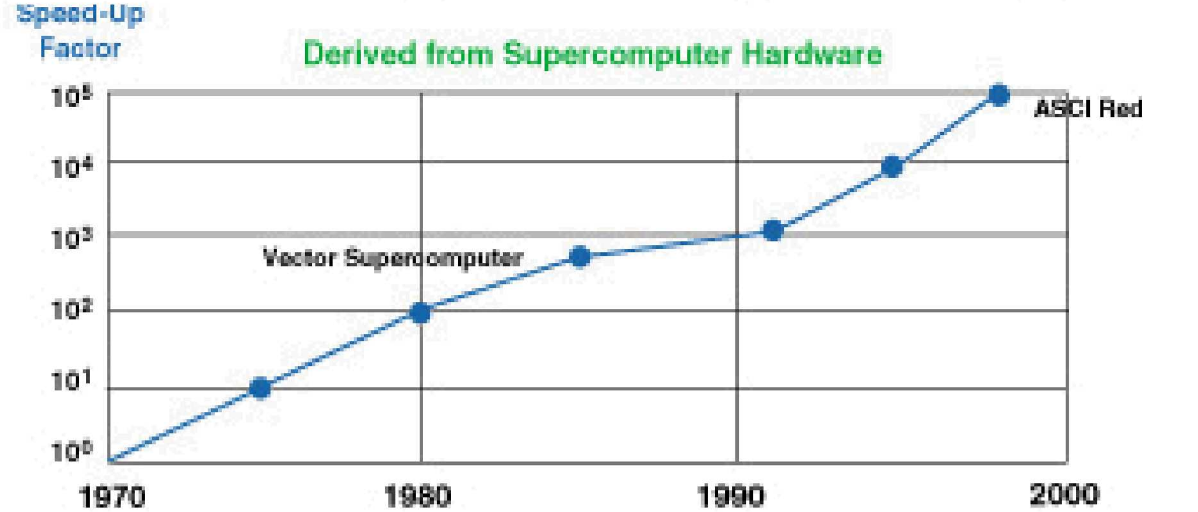
Solution Algorithms



How do we solve our numerical equations?
Example: Linear systems

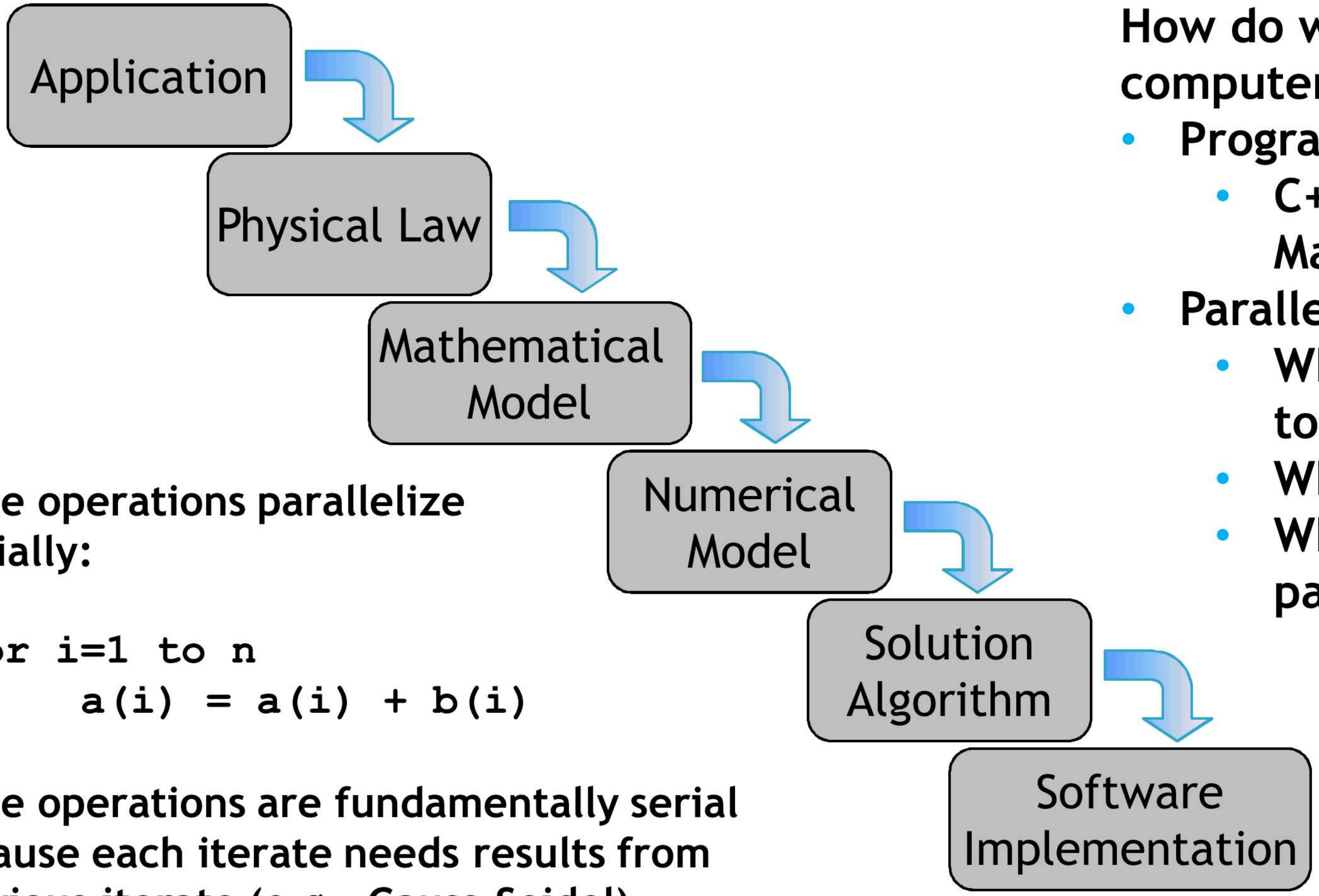
- **Best solver depends upon structure and sparsity of matrix and properties of originating physical problem**





"I would rather have today's algorithms on yesterday's computers than vice versa."
- Phillipe Toint

Software Implementation



How do we implement our algorithm on a real computer?

- **Programming Language**
  - **C++, Fortran 90/77, Python, Matlab, Mathematica**
- **Parallelism**
  - **What portions of our algorithm are amenable to parallelism?**
  - **What parts are fundamentally serial?**
  - **What are communication and computation patterns?**

- **Some operations parallelize trivially:**

```
for i=1 to n
     a(i) = a(i) + b(i)
```

- **Some operations are fundamentally serial because each iterate needs results from previous iterate (e.g., Gauss-Seidel)**

- **Some operations look serial but aren't:**

```
a(0) = 0
a(1) = 1
for i=2 to n
     a(i) = a(i-1) + a(i-2)
```

$$a(i) = \frac{\left(1+\sqrt{5}\right)^{i} - \left(1-\sqrt{5}\right)^{i}}{2^{i}\sqrt{5}}$$

Software Implementation

Application → Physical Law → Mathematical Model → Numerical Model → Solution Algorithm → Software Implementation

Hardware Issues

- Today's hardware is not "flat" -- manycore, GPUs
- Data movement and communication become dominant cost metrics
- Reliability (core failure, etc.) becomes a major issue

- We're being pressed to find and exploit more parallelism in our algorithms. Need million- and billion-way parallelism.
- This requires a lot of math. We must develop and analyze new algorithms.

Summit at ORNL is currently world's fastest computer
200 petaflops (200,000 trillion floating point ops per second)

Summit (6/8/18)

- 4608 Nodes
- 250 PB file system
- #1 on Top500
- #1 on HPCG
- #1 on Green 500 (level 3)

Each node has

- 2 IBM POWER9 processors
- 6 NVIDIA Tesla V100 GPUs
- 608 GB of fast memory
- 1.6 TB of non-volatile memory

Summit is a waypoint towards Exascale.

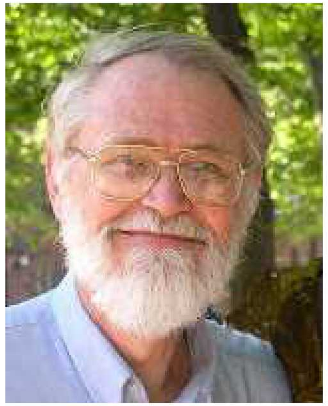
- Exaflop machines by 2021 & 2023

Program Execution

Application → Physical Law → Mathematical Model → Numerical Model → Solution Algorithm → Software Implementation → Program Execution

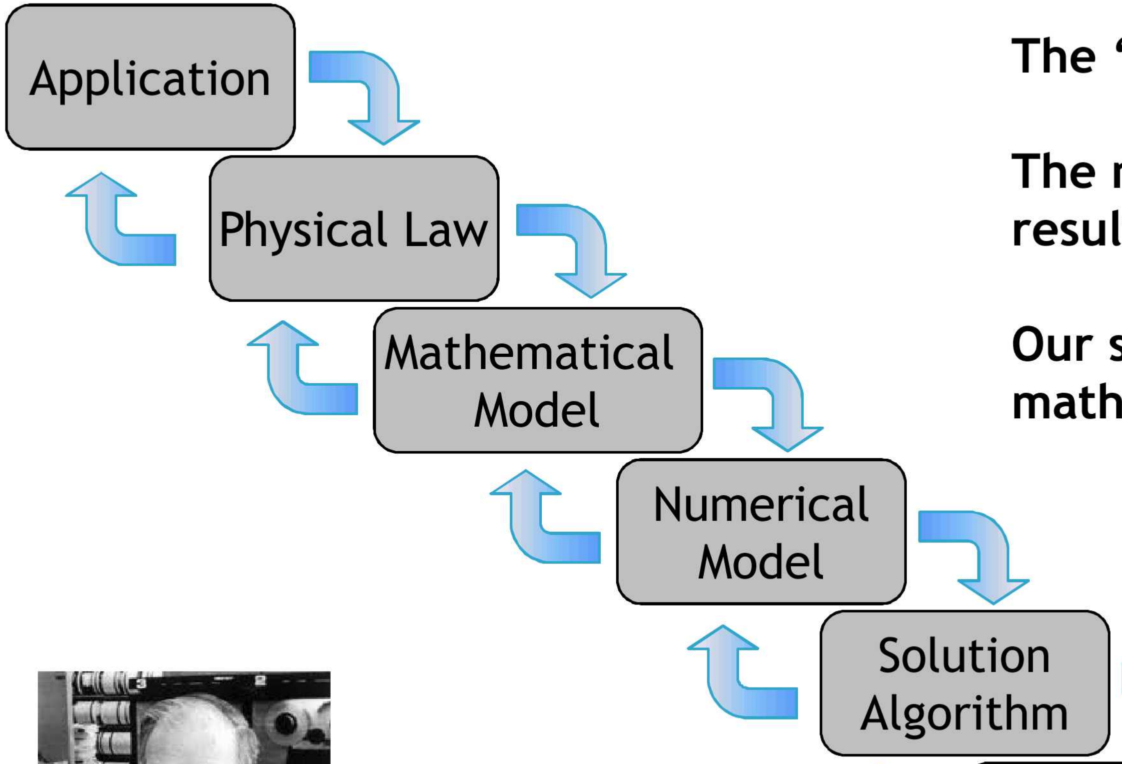
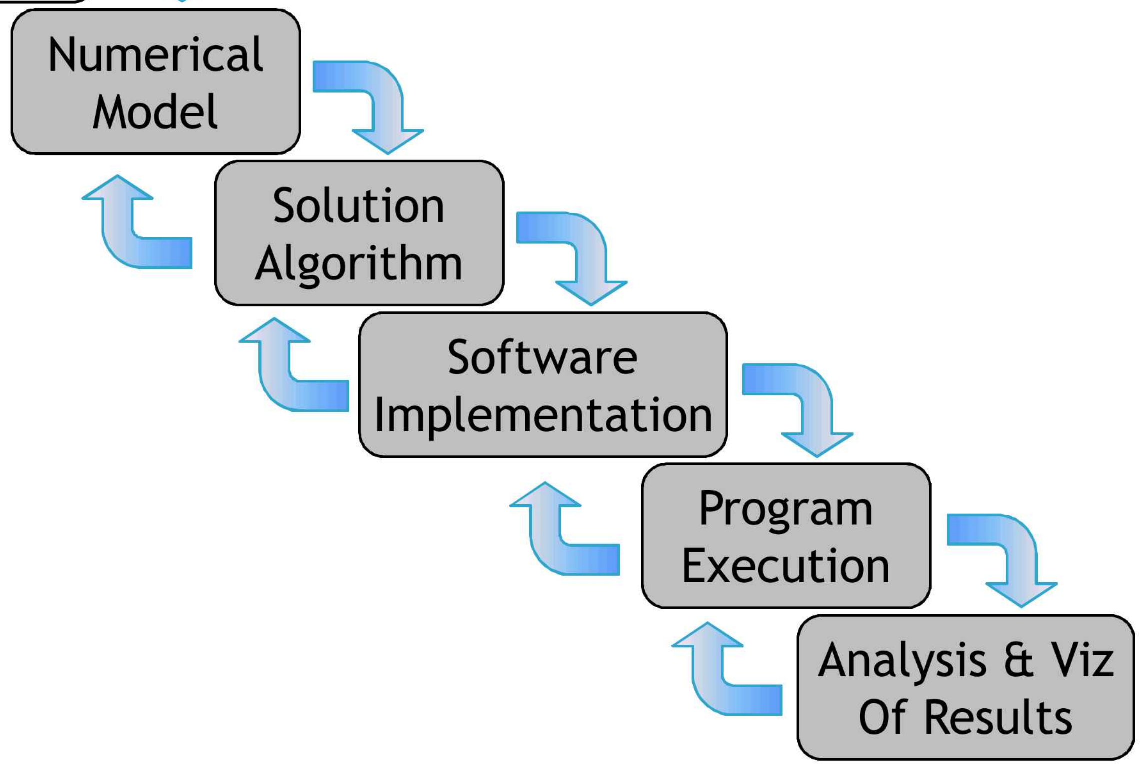
Execution is frequently the easiest part, as long as your job doesn't crash.

The worst possible thing is for your massively parallel code to fail inexplicably several hours/days into execution. Good luck debugging!



"Debugging is at least twice as hard as writing the program in the first place. So if your code is as clever as you can possibly make it, then by definition you're not smart enough to debug it."
- Brian Kernighan

Analysis and Visualization of Results

The “gee-whiz” moment

The most frequent response to analysis of simulation results is to go and run more simulations.

Our simulation results feed back into physical and mathematical models - it’s a closed loop!

“The purpose of computing is insight, not numbers.”
- Richard Hamming

Real-World Impact

I will present four real-world applications taken from work at Sandia National Laboratories.

- They demonstrate the practical importance of scientific computation.
- Applied mathematics is foundational to each of these examples.
- Many of these examples benefit from “big iron” –- massively parallel computers

Sandia is a Dept. of Energy (DOE) Laboratory (12,500+ employees)

- Sandia is a Federally Funded Research and Development Center (FFRDC) for the DOE’s National Nuclear Security Administration (NNSA).
- Sandia is a National Security Laboratory, and has addressed national security challenges for the country since its founding in 1949.
- **Sandia does work for the DOE Office of Science (SC):**
- The DOE Office of Science (SC) is the single largest supporter of basic research in the physical sciences in the United States, funding 40% of all physical sciences basic research
- Advanced Scientific Computing Research (ASCR) within SC has supported much of my research

Sandia’s Five Major Program Portfolios

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

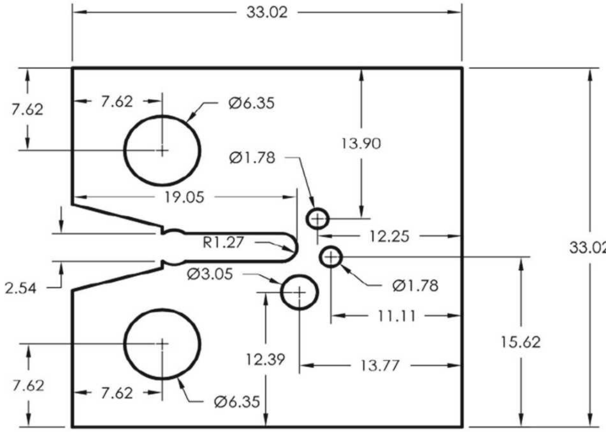
Sandia National Laboratories

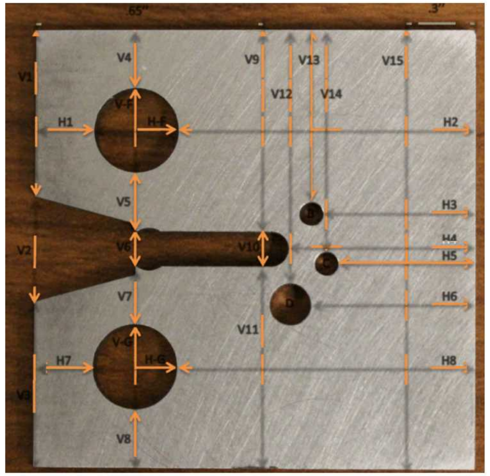
Office of Science U.S. Department of Energy

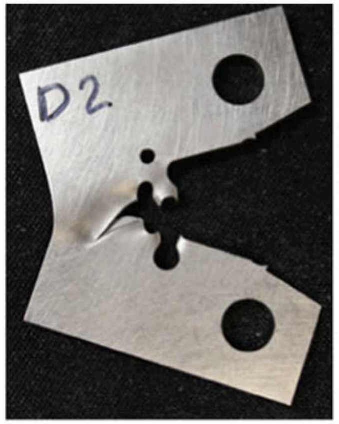
ASCR Advanced Scientific Computing Research

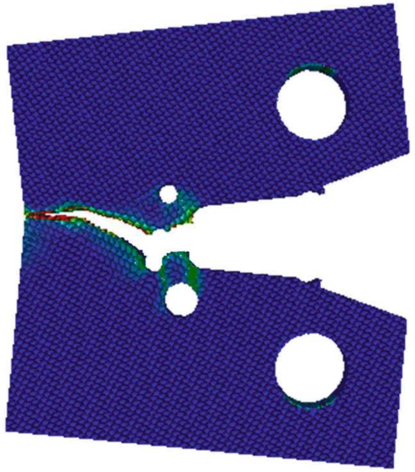
Application #1: Fracture Modeling

- Ductile failure of structural metals is a pervasive issue for applications such as automotive manufacturing, transportation infrastructures, munitions and armor, and energy generation.
- Our confidence in model predictions rests on unbiased assessments of the entire predictive capability, including the mathematical formulation, numerical implementation, calibration, and execution.
- The Sandia Fracture Challenge (SFC) is one such effort to evaluate blind predictive capabilities of ductile failure of an unfamiliar geometry, under practical engineering constraints including limited experimental evidence and computational time.









[*]B. Boyce, et al, The Sandia Fracture Challenge: Blind Round Robin Predictions of Ductile Tearing, Int J Fract. 186:5–68, 2014.

Application #1: Fracture Modeling

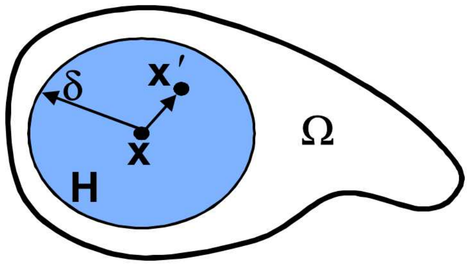
Peridynamics is one model of interest for representing fracture. The classical theory has some problems:

$$\rho\ddot{\mathbf{u}}(\mathbf{x},\mathbf{t}) = \nabla \cdot \sigma(\mathbf{x}) + \mathbf{b}(\mathbf{x},\mathbf{t})$$

- Classical theory predicts infinite stress (1/sqrt(r) singularity) at crack tip.
- Classical theory (based on PDEs) not defined on crack surfaces
- Common numerical approaches (XFEM, etc.) enrich solution space with (for example) heaviside functions to allow admission of discontinuous solutions

Peridynamics was developed by Stewart Silling in 2000 to address shortcomings with the classical theory:

$$\rho\ddot{\mathbf{u}}(\mathbf{x},\mathbf{t}) = \int_{\mathbf{H_x}} \mathbf{f}(\mathbf{u}(\mathbf{x}') - \mathbf{u}(\mathbf{x}), \mathbf{x}'\text{-}\mathbf{x})d\mathbf{V}' + \mathbf{b}(\mathbf{x},\mathbf{t})$$



Point x interacts directly with all points x' within H

- Replace derivatives with integrals
- Peridynamics is nonlocal; classical theory is local
- Peridynamic model admits larger solution space than classical theory
- Utilize same equation everywhere; nothing special about cracks
- Classical theory can be shown to be a special case of peridynamics

Peridynamics opened a path for new mathematics and new computational mechanics capabilities.

Application #1: Fracture Modeling

Peridynamics Open-Source Codes

PDLAMMPS (Peridynamics-in-LAMMPS) (Open source, C++)

- Developers: Parks, Seleson, Plimpton, Silling, Lehoucq
- Particular discretization of PD has computational structure of molecular dynamics (MD)
- LAMMPS: Sandia's open-source massively parallel MD code (lammps.sandia.gov)
- More info & user guide: www.sandia.gov/~mlparks

Peridigm (Open Source, C++)

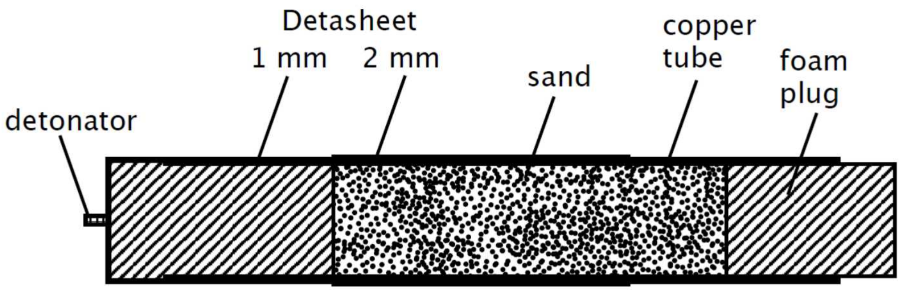
- peridigm.sandia.gov; github.com/peridigm/peridigm
- Developers: Parks, Littlewood, Mitchell, Silling
- Intended as Sandia's primary open-source PD code
- Built upon Sandia's Trilinos Project (trilinos.sandia.gov)
- Massively parallel
- Explicit, implicit time integration
- State-based linear elastic, elastic-plasticity, viscoelastic models
- DAKOTA interface for UQ/optimization/calibration, etc. (dakota.sandia.gov)

Application #1: Fracture Modeling

Explosively Compressed Cylinder*

- Motived by experiments of Vogler & Lappo*
- Commonly used for consolidation of powders
- Copper cylinders filled with granular material and wrapped with Detasheet explosive
- Polyurethane foam plugs used to keep granular sample in tube.

- Geometry and Material Properties
- Copper tubes 305 mm long, ID 50.8 mm, wall thickness of 1.52 mm
- PETN based Detasheet with thicknesses of 1, 2, 4, or 6 mm were used, and a
- Detonation traveled down length of tube, compressing both tube and sand fill



Cylinder schematic

Cylinder after compression

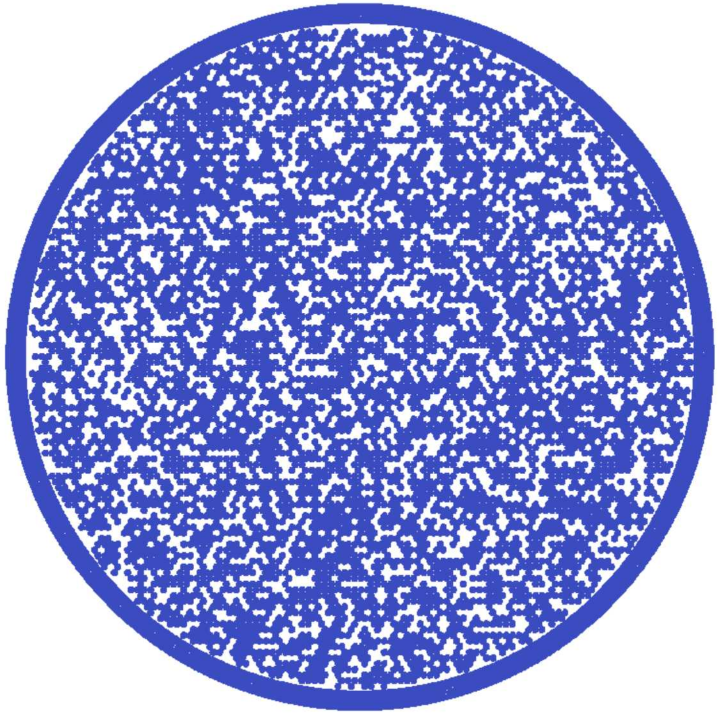
* T.J. Vogler and K.M. Lappo, Cylindrical Compaction of Granular Ceramics: Experiments and Simulations, The 12th Hypervelocity Impact Symposium. 2012.

Application #1: Fracture Modeling

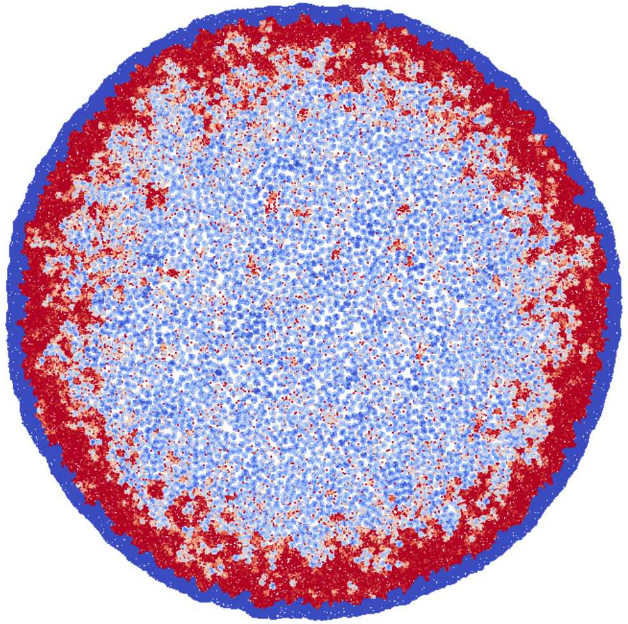
Explosively Compressed Cylinder*

- Peridigm computational results (with C. Hoffarth (ASU), D. Littlewood (SNL))
- Color indicates damage (blue = undamaged, red = damaged)

Simulation performed with Peridigm



Before



After

Application #1: Fracture Modeling

The development of peridynamics has led to a new nonlocal calculus*

Simplest peridynamics operator looks like a "continuum difference"

- $L(u) \coloneqq \int_{-\delta}^{\delta} \frac{u(x+\varepsilon)-u(x)}{|\varepsilon|^{a}} d\varepsilon = \int_{0}^{\delta} \frac{u(x+\varepsilon)-2u(x)+u(x-\varepsilon)}{|\varepsilon|^{a}} d\varepsilon$ for $\mathbf{x} \in \mathbb{R}$

Nonlocal point divergence operator

- Let $\boldsymbol{\nu}(\mathbf{x},\mathbf{y}): \mathbb{R}^{\boldsymbol{n}} \times \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}^{\boldsymbol{n}}$, $\boldsymbol{\alpha}(\mathbf{x},\mathbf{y}): \mathbb{R}^{\boldsymbol{n}} \times \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}^{\boldsymbol{n}}$, $\boldsymbol{\alpha}$ antisymmetric. Then,
  - $\mathcal{D}(\boldsymbol{\nu})(\mathbf{x}) \coloneqq \int_{\mathbb{R}^n} (\boldsymbol{\nu} + \boldsymbol{\nu}') \cdot \boldsymbol{\alpha}\, d\mathbf{y}$ for $\mathbf{x} \in \mathbb{R}^{\boldsymbol{n}}$

  where $\mathcal{D}(\boldsymbol{\nu})(\mathbf{x}): \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}$

Nonlocal point gradient operator

- Let $\eta(\mathbf{x},\mathbf{y}): \mathbb{R}^{\boldsymbol{n}} \times \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}$, $\boldsymbol{\beta}(\mathbf{x},\mathbf{y}): \mathbb{R}^{\boldsymbol{n}} \times \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}$, $\boldsymbol{\beta}$ antisymmetric. Then,
  - $\mathcal{G}(\eta)(\mathbf{x}) \coloneqq \int_{\mathbb{R}^n} (\eta + \eta')\boldsymbol{\beta} d\mathbf{y}$ for $\mathbf{x} \in \mathbb{R}^{\boldsymbol{n}}$

  where $\mathcal{G}(\eta)(\mathbf{x}): \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}$

Nonlocal point curl operator

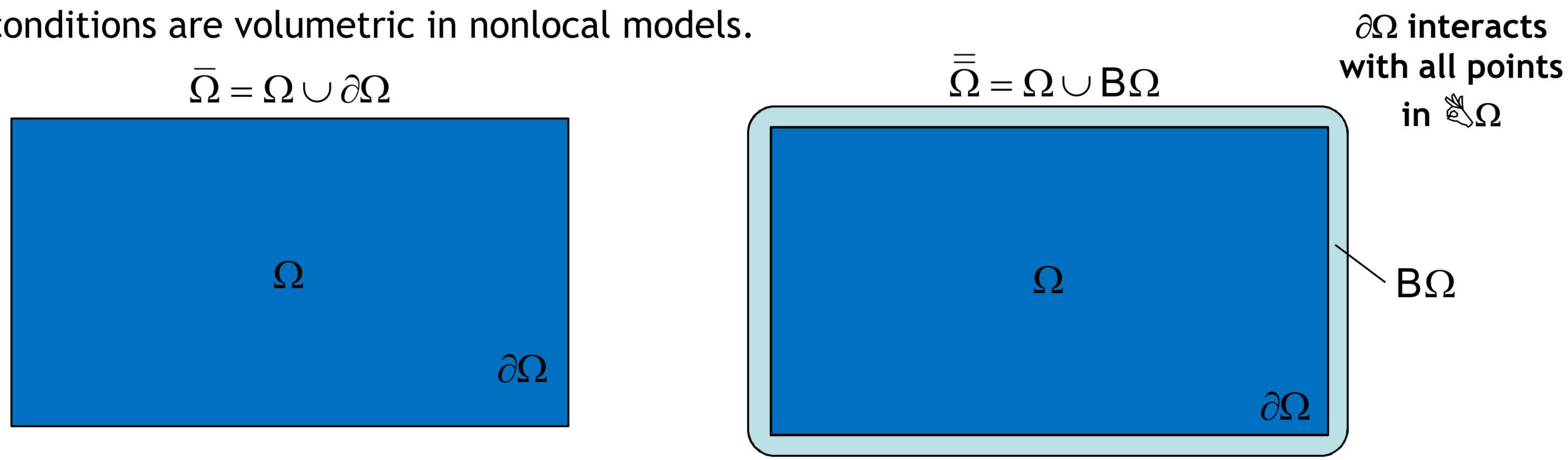
- Let $\boldsymbol{\mu}(\mathbf{x},\mathbf{y}): \mathbb{R}^{\boldsymbol{n}} \times \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}^{\mathbf{3}}$, $\boldsymbol{\gamma}(\mathbf{x},\mathbf{y}): \mathbb{R}^{\boldsymbol{n}} \times \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}^{\mathbf{3}}$, $\boldsymbol{\gamma}$ antisymmetric. Then,
  - $\mathcal{C}(\boldsymbol{\mu})(\mathbf{x}) \coloneqq \int_{\mathbb{R}^n} \boldsymbol{\gamma} \times (\boldsymbol{\mu} + \boldsymbol{\mu}') d\mathbf{y}$ for $\mathbf{x} \in \mathbb{R}^{\boldsymbol{n}}$

  where $\mathcal{C}(\eta)(\mathbf{x}): \mathbb{R}^{\boldsymbol{n}} \to \mathbb{R}^{\mathbf{3}}$

* Q. Du, M. Gunzburger, R. Lehoucq, K. Zhou, A nonlocal vector calculus, nonlocal volume-constrained problems, and nonlocal balance laws, Math. Models Methods Appl. Sci. 23 (2013) 493–540

Application #1: Fracture Modeling

"Boundary" conditions are volumetric in nonlocal models.



Local **Nonlocal**

One can define associated nonlocal interaction operators:*

- $\mathcal{N}(\boldsymbol{\nu})(\mathbf{x}) \coloneqq -\int_{\Omega \cup B\Omega} (\boldsymbol{\nu} + \boldsymbol{\nu}') \cdot \boldsymbol{\alpha}\, d\mathbf{y}$ for $\mathbf{x} \in B\Omega$
- $\mathcal{S}(\eta)(\mathbf{x}) \coloneqq -\int_{\Omega \cup B\Omega} (\eta + \eta')\boldsymbol{\beta} d\mathbf{y}$ for $\mathbf{x} \in B\Omega$
- $\mathcal{T}(\boldsymbol{\mu})(\mathbf{x}) \coloneqq -\int_{\Omega \cup B\Omega} \boldsymbol{\gamma} \times (\boldsymbol{\mu} + \boldsymbol{\mu}') d\mathbf{y}$ for $\mathbf{x} \in B\Omega$

* Q. Du, M. Gunzburger, R. Lehoucq, K. Zhou, A nonlocal vector calculus, nonlocal volume-constrained problems, and nonlocal balance laws, Math. Models Methods Appl. Sci. 23 (2013) 493–540

Application #1: Fracture Modeling

This provides the tools to express familiar identities and relationships from local calculus in nonlocal calculus:

Nonlocal integral theorems*

- $\int_{\Omega} \mathcal{D}(\boldsymbol{v})(\mathbf{x})\, d\mathbf{x} = \int_{\mathrm{B}\Omega} \mathcal{N}(\boldsymbol{v})(\mathbf{x})\, d\mathbf{x}$
- $\int_{\Omega} \mathcal{G}(\boldsymbol{v})(\mathbf{x})\, d\mathbf{x} = \int_{\mathrm{B}\Omega} \mathcal{S}(\boldsymbol{v})(\mathbf{x})\, d\mathbf{x}$
- $\int_{\Omega} \mathcal{C}(\boldsymbol{v})(\mathbf{x})\, d\mathbf{x} = \int_{\mathrm{B}\Omega} \mathcal{T}(\boldsymbol{v})(\mathbf{x})\, d\mathbf{x}$

Classical integral theorems

$\int_{\Omega} \nabla \cdot \mathbf{v}\, d\mathbf{x} = \int_{\partial\Omega} \mathbf{v} \cdot \mathbf{n}\, d\mathbf{x}$

$\int_{\Omega} \nabla v\, d\mathbf{x} = \int_{\partial\Omega} v\mathbf{n}\, d\mathbf{x}$

$\int_{\Omega} \nabla \times \mathbf{v}\, d\mathbf{x} = \int_{\partial\Omega} \mathbf{n} \times \mathbf{v} d\mathbf{x}$

There are physical interpretations to these equations:

- For example, the first states that the integral of the nonlocal divergence of $\boldsymbol{v}$ over Ω is equal to the total flux out of Ω into $\mathrm{B}\Omega$.
- We recognize this as the Gauss divergence theorem.

The development of the nonlocal calculus has some parallels with development of the local calculus:

- We can't agree on notation. The engineers use one notation (a nonlocal analog of classical solid mechanics notation); the mathematicians use another (a nonlocal analog of classical calculus notation).

In nonlocal mathematics, hard things become easy, but some easy things become hard:

- How to apply Neumann-like boundary conditions is an active area of research!

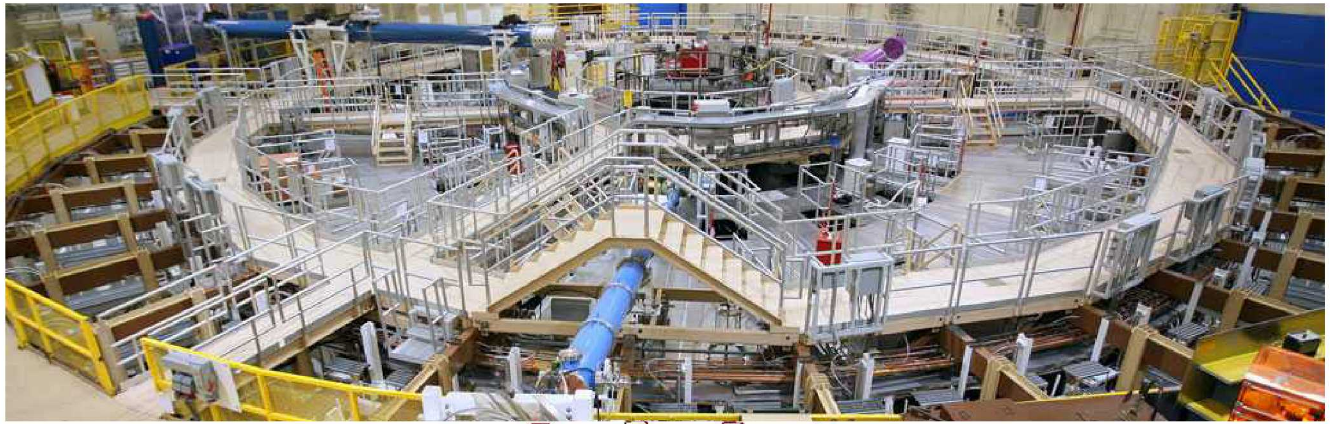
There is an enormous amount of research on this topic at UNL! (Bobaru, Foss, Radu, students+postdocs)

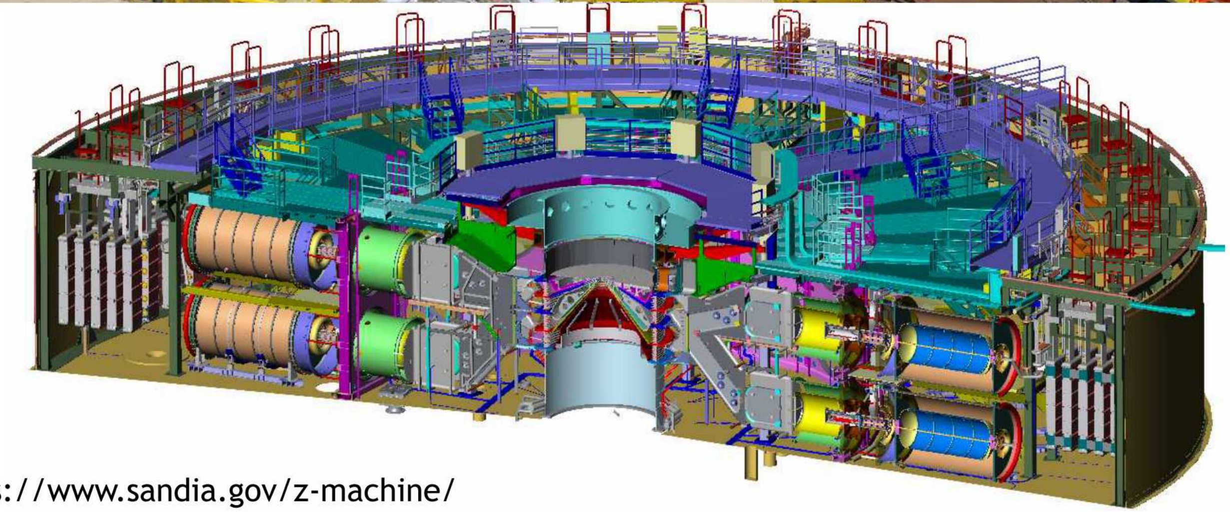
* Q. Du, M. Gunzburger, R. Lehoucq, K. Zhou, A nonlocal vector calculus, nonlocal volume-constrained problems, and nonlocal balance laws, Math. Models Methods Appl. Sci. 23 (2013) 493–540

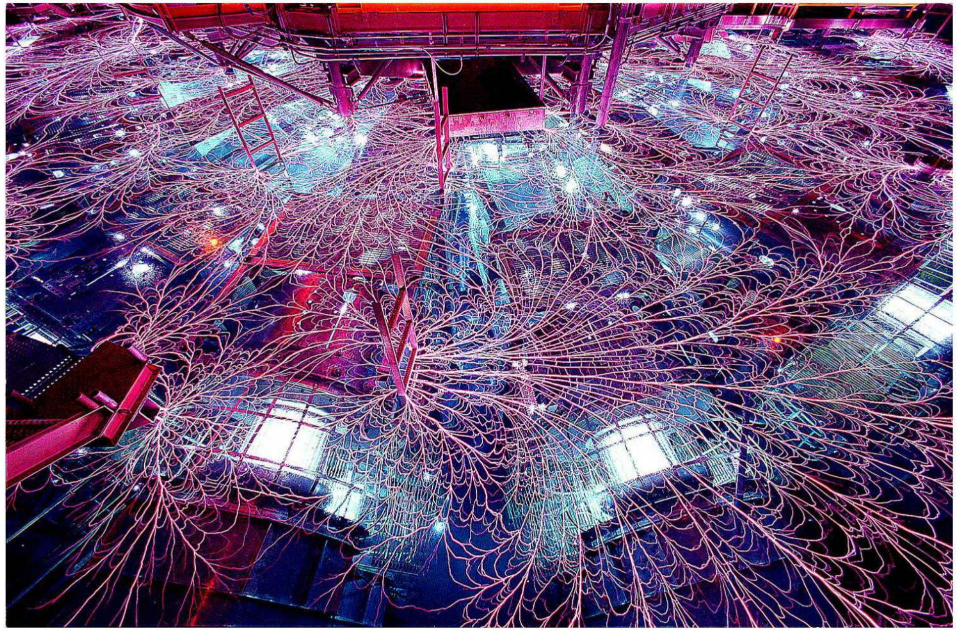
Application #2: Compatible Discretizations

The Z machine uses electricity to create radiation and high magnetic pressure, which are both applied to a variety of scientific purposes ranging from weapons research to the study of fusion energy.

The Z-machine at Sandia Lab is the world’s most powerful x-ray source. It’s peak electrical power is about 80 trillion watts (~6x the world's electrical power output) – but only for ~100 nanoseconds![1]







Electrical discharges illuminating surface of Z machine

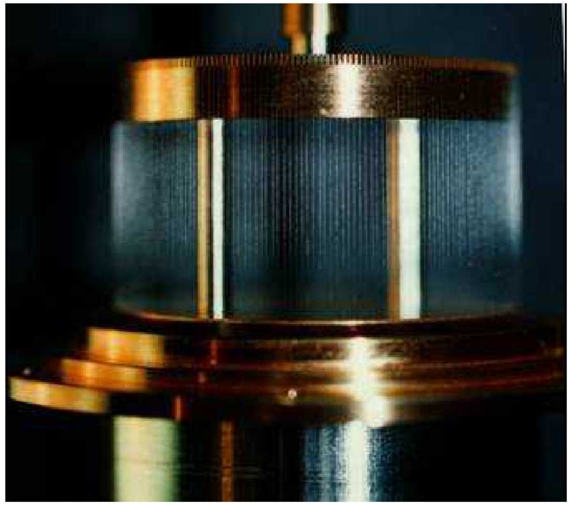
[1] https://www.sandia.gov/z-machine/

Application #2: Compatible Discretizations

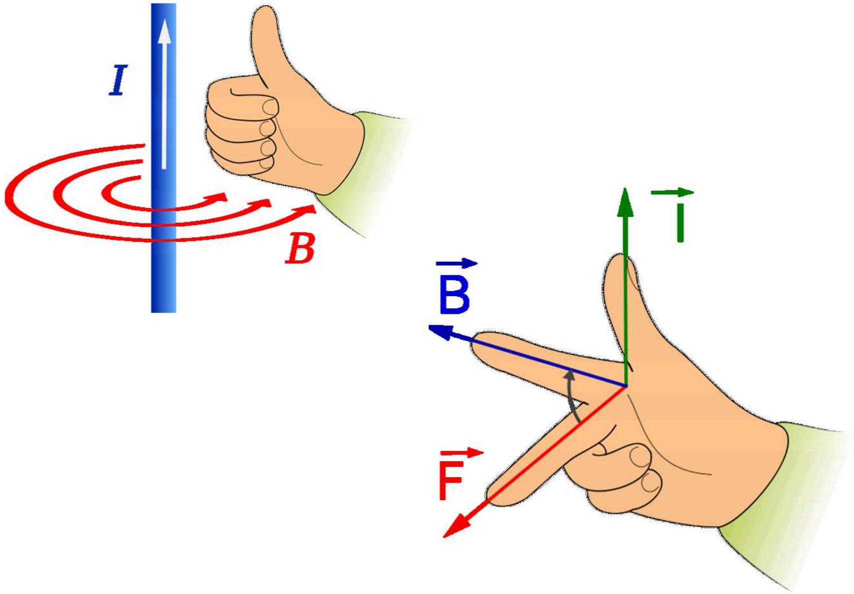
A powerful electrical discharge is fired into a target consisting of hundreds of tungsten wires, each thinner than a human hair, enclosed in a small metal container known as a hohlraum.

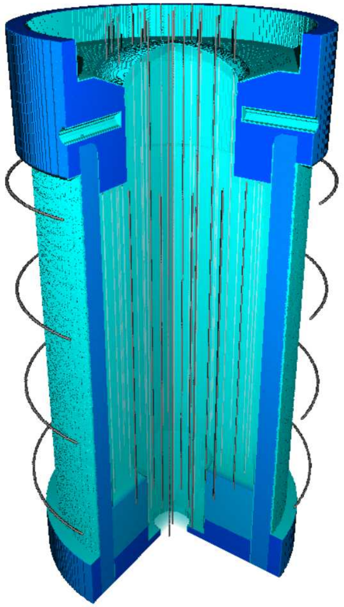
The high electrical current (about 26 million amps) vaporizes the wires, which are transformed into a cylindrical plasma curtain, and creates strong magnetic forces that that radially compress the plasma along the z-axis (hence the name, Z-Machine).

The imploding plasma produces intense radiation (2 million joules of X-ray energy) and can heat hohlraum walls to over 2 billion degrees Celsius.



Wire array





Application #2: Compatible Discretizations[1]

Understanding complex, multi-physics processes involved in Z requires modeling magnetic diffusion in highly heterogeneous conductors.

This proved difficult with standard approaches.

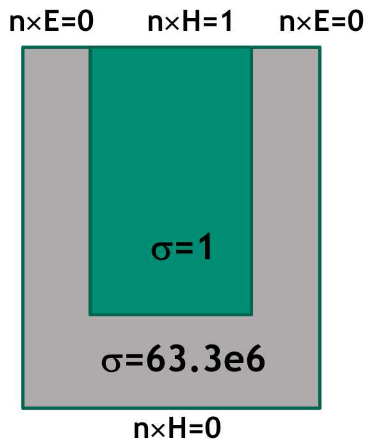
$$\nabla \times \mathbf{H} = \mathbf{J}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial \mathbf{t}} \qquad \mathbf{B} = \mu \mathbf{H}$$

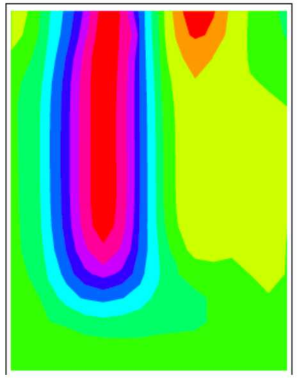
$$\nabla \cdot \mathbf{B} = 0 \qquad \mathbf{J} = \sigma \mathbf{E}$$

$$\nabla \cdot \mathbf{J} = 0$$

Eddy current equations (Maxwell equations neglecting displacement current)



2D Model Problem



Steady state B field (nodal discretization; solution contaminated)

Small discretization errors, usually benign, can be disastrous if amplified by other system components.

For example, discrete curl operator with nodal discretization is nonsingular; Continuous counterpart has infinite-dimensional nullspace. Nodal finite element space does not contain exact gradients.

[1] P.B. Bochev and A.C. Robinson, Matching algorithms with physics: exact sequences of finite element spaces (with A. Robinson). In: Collected lectures on preservation of stability under discretization, edited by D. Estep and S. Tavener, SIAM, Philadelphia, 2001.

Application #2: Compatible Discretizations[1]

Solution: Align the discretization with the physics!

- Domains of grad, div, and curl:
  - $H_0(\Omega, \text{grad}) = \{ \varphi \in H(\Omega, \text{grad}) \mid \varphi = 0 \text{ on } \Gamma \}$
  - $H_0(\Omega, \text{curl}) = \{ \varphi \in H(\Omega, \text{curl}) \mid u \times n = 0 \text{ on } \Gamma \}$
  - $H_0(\Omega, \text{div}) = \{ \varphi \in H(\Omega, \text{div}) \mid u \cdot n = 0 \text{ on } \Gamma \}$

- $H(\Omega, \text{grad})$, $H(\Omega, \text{curl})$, $H(\Omega, \text{div})$ are spaces of square integrable functions with gradients, curls, or divergences that are also square integrable.
- $L^2(\Omega)$ denotes space of square integrable functions

- The four spaces above and the operators ∇, $\nabla\times$, $\nabla\cdot$ form a De Rham complex relative to Γ.
- Each differential operator maps its domain onto the kernel of the next operator.
- This is called an *exact sequence*.

$$H_0(\Omega,\text{grad}) \xrightarrow{\nabla} H_0(\Omega,\text{curl}) \xrightarrow{\nabla\times} H_0(\Omega,\text{div}) \xrightarrow{\nabla\cdot} L_0^2(\Omega)$$

- Why is this important? Because the terms in Maxwell's equations can be built on this complex.

[1] P.B. Bochev and A.C. Robinson, Matching algorithms with physics: exact sequences of finite element spaces (with A. Robinson). In: Collected lectures on preservation of stability under discretization, edited by D. Estep and S. Tavener, SIAM, Philadelphia, 2001.

Application #2: Compatible Discretizations[1]

$H_0(\Omega, \text{grad})$	ψ		**0**	$\mathbf{L}_0^2(\Omega)$
∇	$\downarrow$		$\Uparrow$	$\nabla\cdot$
$H_0(\Omega, \text{curl})$	**H**	$\Rightarrow \mu H = B \Rightarrow$	**B**	$\mathbf{H}_0^*(\Omega, \text{div})$
$\nabla\times$	$\Downarrow$		$\Uparrow$	$\nabla\times$
$H_0(\Omega, \text{div})$	**J**	$\Leftarrow J = \sigma E \Leftarrow$	**E**	$\mathbf{H}_0^*(\Omega, \text{curl})$
$\nabla\cdot$	$\Downarrow$		$\uparrow$	∇
$\mathbf{L}_0^2(\Omega)$	**0**		φ	$\mathbf{H}_0^*(\Omega, \text{curl})$

- This table reveals the structure of Maxwell's equations.
- To capture this behavior, our discretization must approximate both these spaces and the relationships between them.
- If we build 4 finite elements spaces W^1, W^2, W^3, W^4 that are *subspaces* of these 4 spaces *and* which form an exact sequence then we can address the underlying issue.

$$\mathbf{W}^0 \xrightarrow{\nabla} \mathbf{W}^1 \xrightarrow{\nabla\times} \mathbf{W}^2 \xrightarrow{\nabla\cdot} \mathbf{W}^3$$

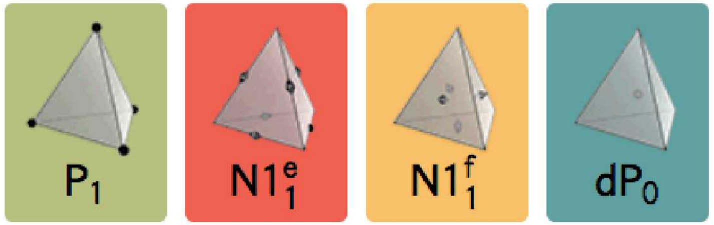
- This is known as a *compatible discretization.*

[1] P.B. Bochev and A.C. Robinson, Matching algorithms with physics: exact sequences of finite element spaces (with A. Robinson). In: Collected lectures on preservation of stability under discretization, edited by D. Estep and S. Tavener, SIAM, Philadelphia, 2001.

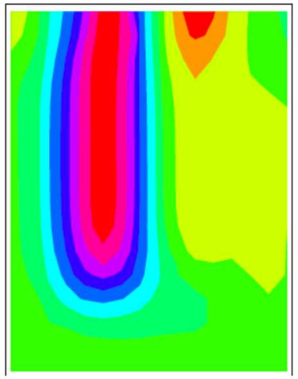
Application #2: Compatible Discretizations[1]

As a consequence, we have that

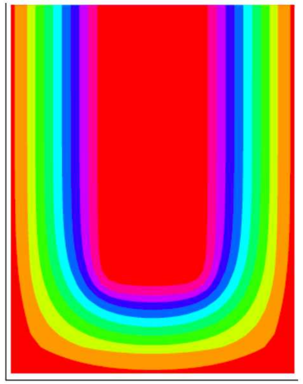
- Electric fields live on edges: H(Ω, curl)
- Magnetic fields live on faces: H(Ω, div)
- Densities/potentials live at nodes: H(Ω, grad)



Compatible finite element methods (i.e., that respect Maxwell's equations at the element level) are the key to getting the right answer!



Steady state B field (nodal discretization; solution contaminated)



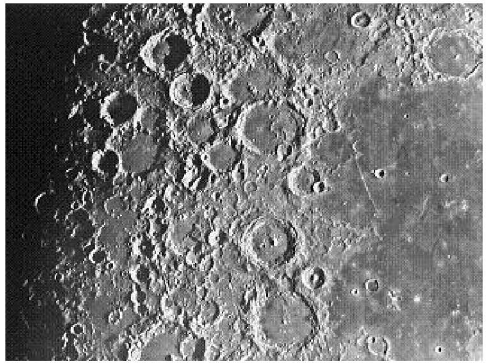
Steady state B field (compatible discretization)

[1] P.B. Bochev and A.C. Robinson, Matching algorithms with physics: exact sequences of finite element spaces (with A. Robinson). In: Collected lectures on preservation of stability under discretization, edited by D. Estep and S. Tavener, SIAM, Philadelphia, 2001.

Application #3: Asteroid Impact Modeling

Earth and other bodies frequently suffer impact events

More visible on moon due to much less erosion



Moon

Objects with diameters of approx. 5-10 meters:

- Impact Earth approximately once per year
- Energy release is ~ 15 kilotons of TNT

Treefall at Tunguska

Objects with diameters of approx. 50 m:

- Impact Earth approximately once every thousand years.
- Energy release is ~ 15 megatons of TNT
- Tunguska in 1908 (only 3-5 MT?)
- Barringer Crater (50,000 years ago)

Barringer Crater, AZ.

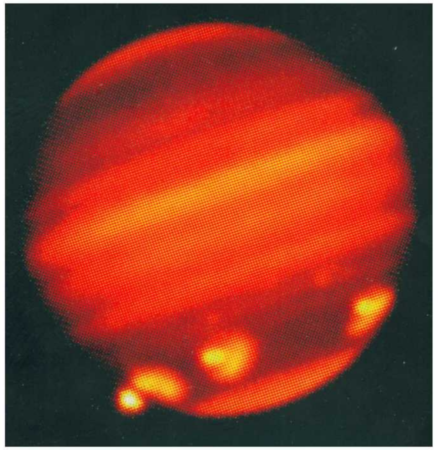
Application #3: Asteroid Impact Modeling

Objects with diameters of 5 km
- Impact Earth approximately every ten million years
- Example: Chicxulub impact, ~ 65 million years ago
- Energy release ~ 100 teratons of TNT
- Believed to be cause of K-T mass extinction event (including dinosaurs)

Chicxulub Impact (Artist rendering)

Similar size impacts have been observed in recent history:
- Comet Shoemaker-Levy 9 event (1994)
- Series of comet fragments ~ 2 km impacted Jupiter
- Energy release ~ 6,000,000 MT of TNT
- Created dark spot 12,000 km across

What can we do to prevent this from happening today?



Jupiter, 7/20/1994

Application #3: Asteroid Impact Modeling

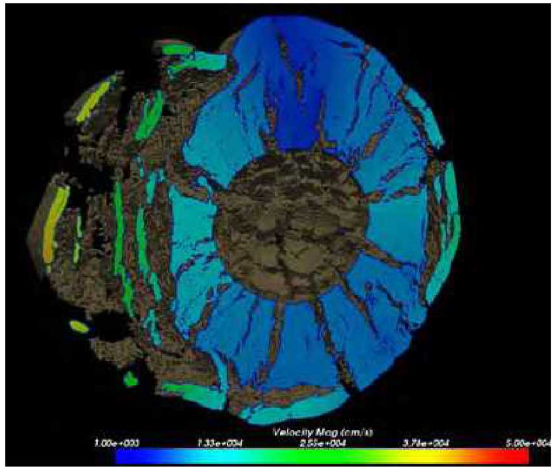
Using modern computational science tools, explore impact effects and mitigation strategies

- Explore effects of impact, plan for evacuations
- Explore impact avoidance strategies (deflection, disruption)
- Ultimate low-risk, high-reward venture!

Boslough

Recently retired Sandia physicist Mark Boslough is an expert on planetary impacts and has performed detailed computational simulations on asteroid impacts and mitigation strategies.

Boslough used Sandia supercomputers to examine both deflection and disruption strategies to prevent large asteroid impact with Earth. Deflection keeps the asteroid in one piece and changes its trajectory to miss Earth. Disruption fragments and disperses the asteroid so that all the large pieces miss Earth.



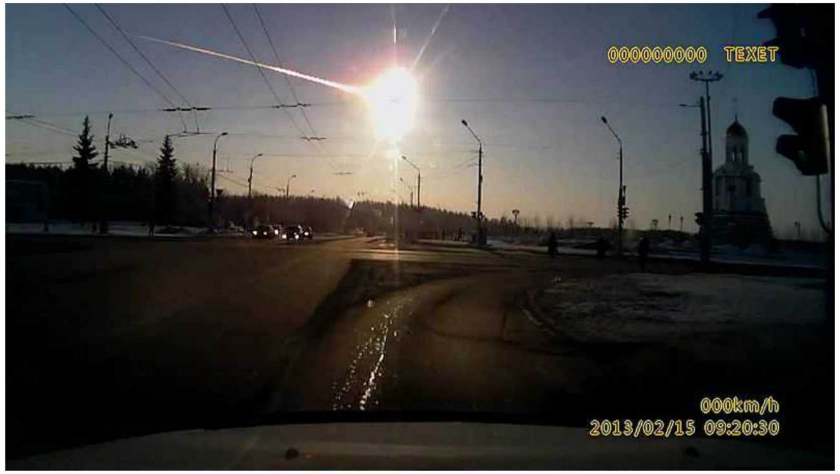
Numerical simulation of destruction of 500 meter asteroid Golveka by 10 megaton explosion using a Sandia shock-physics code in study of defensive strategies for Earth-crossing asteroids.[1]

[1] M. Boslough, Red Storm evaluates strategies to protect the Earth from a cosmic impact. SAND 2006-6071.

Application #3: Asteroid Impact Modeling

On February 15, 2013, an asteroid exploded over Chelyabinsk, Russia.

- The rock was an ~20m ordinary chondrite
- It has a speed of about 19 km/s (> Mach 60)
- It exploded in an airburst at a height of ~18.5 miles
- It had an energy yield equivalent to 500 kilotons of TNT
- It shone 30 times brighter than the Sun
- The shock waves resulting from the main blast blew people off their feet and shattered thousands of windows in Chelyabinsk



Meteor over Chelyabinsk on February 15, 2013[1]

Vapor cloud trail left by the Chelyabinsk asteroid as seen by M. Ahmetvaleev on 15 February 2013.[2]

[1] https://en.wikipedia.org/wiki/Chelyabinsk_meteor
[2] http://www.esa.int/spaceinimages/Images/2017/06/Chelyabinsk_asteroid

Application #3: Asteroid Impact Modeling[1]

- Mark Boslough found out about the event via Facebook, hours before the sound wave from the explosion reached North America.

- He traveled to Chelyabinsk (with a NOVA TV crew) to collect information and samples from the meteorite.

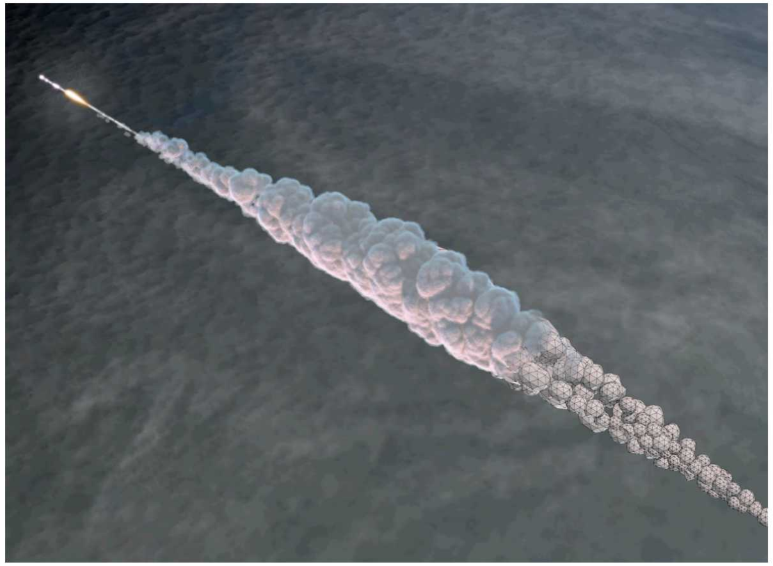
- He performed stellar calculations of the asteroid's trajectory by visiting — at night when the stars shone — the exact spots where the footage was recorded. If the stars show up on the digital camera, one can get those angles and then calibrate that image to know exactly the angles to the trajectory of the fireball. This provinces a very precise asteroid trajectory through the atmosphere, so one can backtrack that to get the pre-impact orbit.

- Mark also collected light curves to estimate the energy released.[2] (~500 kT of TNT)

[1] https://share-ng.sandia.gov/news/resources/news_releases/boslough_asteriod/
[2] Brown, P. G. *et al. A 500-kiloton airburst over Chelyabinsk and an enhanced hazard from small impactors. Nature* http://dx.doi.org/10.1038/nature12741

Application #3: Asteroid Impact Modeling[1]

- Returning to Sandia, he ran simulations of the asteroid's fragmentation. (CS&E!)
- His research was featured in Nature[1] and Physics Today[2], as well as on the NOVA Episode, "Meteor Strike"[3]
- Simulations also explained the splitting of the wake into bilateral, contrarotating vortices: The center of the cylinder rose faster than the edges, which caused both sides of the rising parcel to rotate outward in a way analogous to the toroidal vortex surrounding the buoyant fireball in a nuclear explosion.[2]



Chelyabinsk Asteroid Simulation
(viz by Brad Carvey)[4]

[1] Brown, P. G. *et al. A 500-kiloton airburst over Chelyabinsk and an enhanced hazard from small impactors. Nature* http://dx.doi.org/10.1038/nature12741
[2] A. Kring, David & Boslough, Mark. (2014). Chelyabinsk: Portrait of an asteroid airburst. Physics Today. 67. 32-37. 10.1063/PT.3.2515.
[3] https://www.pbs.org/wgbh/nova/video/meteor-strike
[4] https://share-ng.sandia.gov/news/resources/news_releases/boslough_asteriod/

Application #4: Operation Burnt Frost

On February 20, 2008, a malfunctioning satellite was destroyed in "Operation Burnt Frost."[1,2]

This is an approximate timeline of events:

December 14, 2006:

- Satellite malfunctioned shortly after deployment. Satellite was in deteriorating orbit and expected to reenter Earth's atmosphere.
- Satellite contained 1000 pounds of toxic fuel, which was expected to survive re-entry.

January 4, 2008:

- President ordered threat of satellite mitigated.

February 14, 2008:

- General James Cartwright, Vice Chairman of the Joint Chiefs of Staff, announces that U.S. intended to shoot down malfunctioning satellite. Desire was to intercept it before it entered Earth's atmosphere, but to wait until the satellite was close to re-entry in order to limit the amount of debris created. This gave an eight day window.

February 20, 2008:

- At 1:00 p.m. EST, Secretary of Defense Gates, with consultation of the White House, approves mission.
- At 10:26 p.m. EST, SM-3 missile launched from USS Lake Erie. Satellite destroyed minutes later.

[1] https://www.af.mil/News/Article-Display/Article/124266/joint-effort-made-satellite-success-possible/
[2] http://www.norad.mil/Newsroom/Article/578258/navy-missile-hits-decaying-satellite-over-pacific-ocean/

Application #4: Operation Burnt Frost

Missile Defense Agency (MDA) asked Sandia Labs to tell them where to strike the satellite.

Technical issues:

- Satellite moving at 17,000 miles per hour
- Satellite 153 miles above the Earth
- Kill vehicle moving at thousands of miles per hour at impact

Operation Burnt Frost
SAND2009-0665P

Technical objectives:

- Must hit satellite.
- Must fragment fuel tank
- Must hit such that resulting debris contains no large fragments that might threaten Earth

Many hypervelocity impact simulations were run on Red Storm (largest computer at Sandia at that time) and used to program intercept vehicle to achieve objectives.

(CS&E) This required

- Validated computational models describing the physics (physical laws, mathematical models)
- Massively parallel simulation code (scalable numerical algorithms, software implementation)

Goal was not enhanced scientific understanding, but to advise policy makers.

Summary

What is CS&E?

- CS&E Application Space
- Relationship between math & physical sciences
- What is CS&E? – A guided tour.

Applications

- Fracture mechanics, nonlocal modeling
- Compatible discretizations
- Asteroid impact modeling
- Errant satellite shootdown

Today's Opportunities:

- Heterogeneous computing; Exascale
- Data Science
- Scientific Machine Learning

I've left out a *lot* about CS&E.

Thank you!