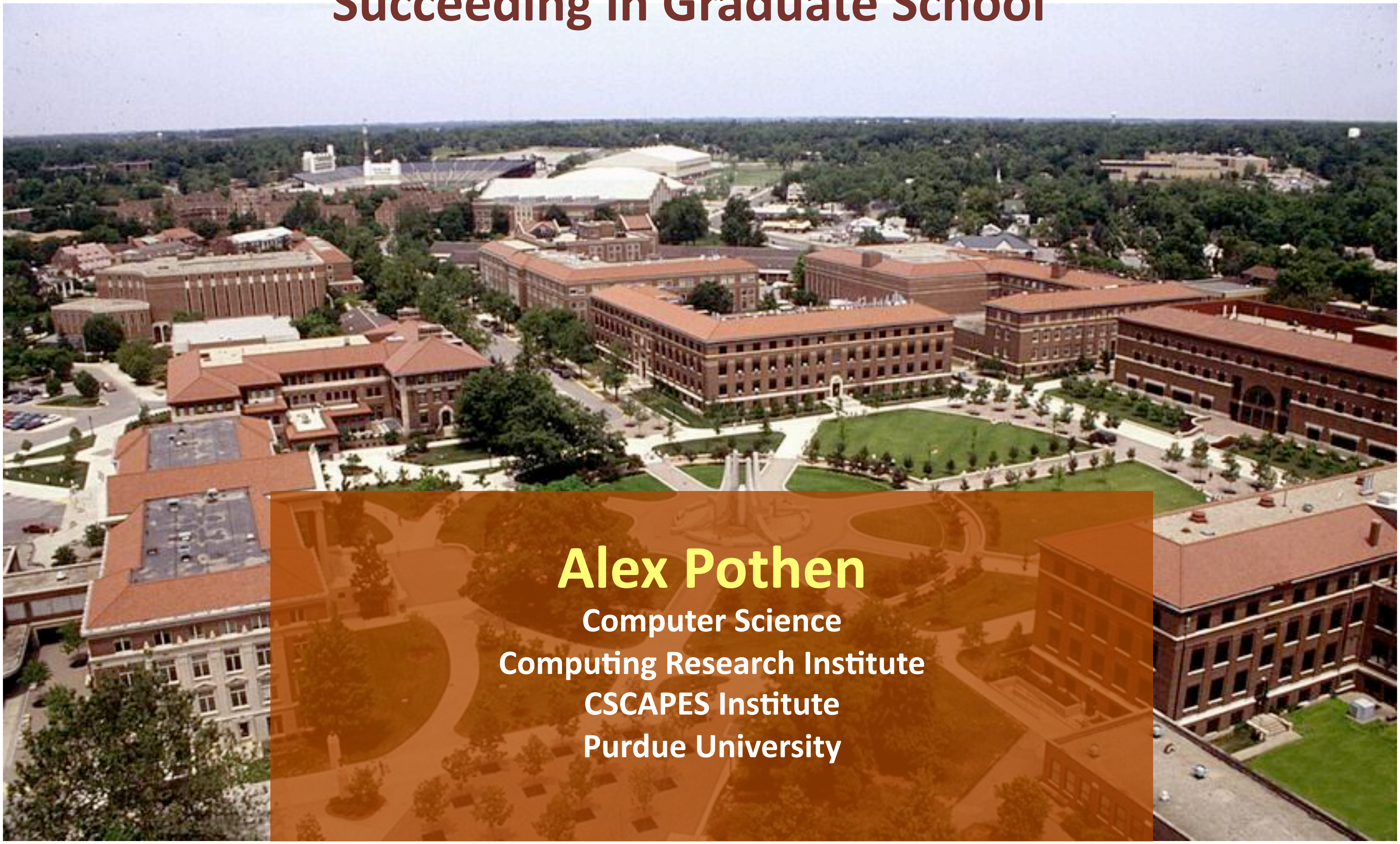
Research in CSE and Bioinformatics
Succeeding in Graduate School



Alex Pothen

Computer Science

Computing Research Institute

CSCAPES Institute

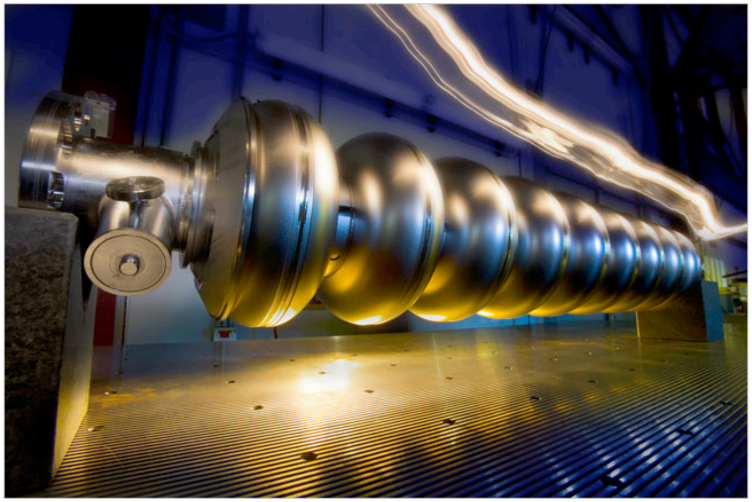
Purdue University

http://www.cs.purdue.edu/homes/apothen

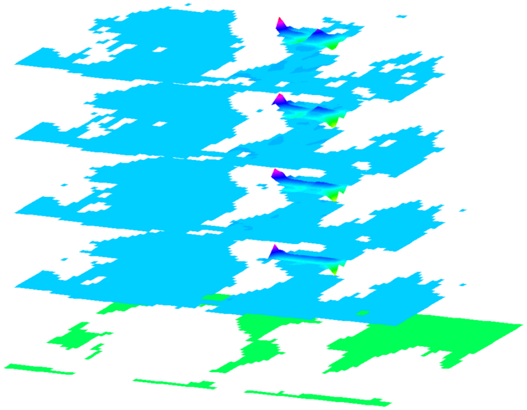
Overview

- Research in CSE and Bioinformatics
  - SciDAC program and the CSCAPES Institute
  - Automatic Differentiation
  - Others: Grama, Sameh, Skeel; Kihara, Qi, Si, Szpankowski, O. Vitek
- Succeeding in Graduate School
  - Focus
  - Writing
  - Giving a Talk

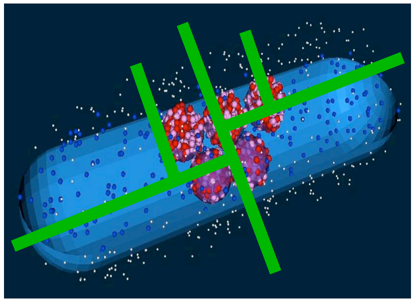
Energy sciences



…bring together scientists and engineers with computer scientists and applied mathematicians, recognizing that computation is not something you do at the end, but is built into the solution of the problem one is addressing…
Ray Orbach, Under-sec for Science, DOE



Climate



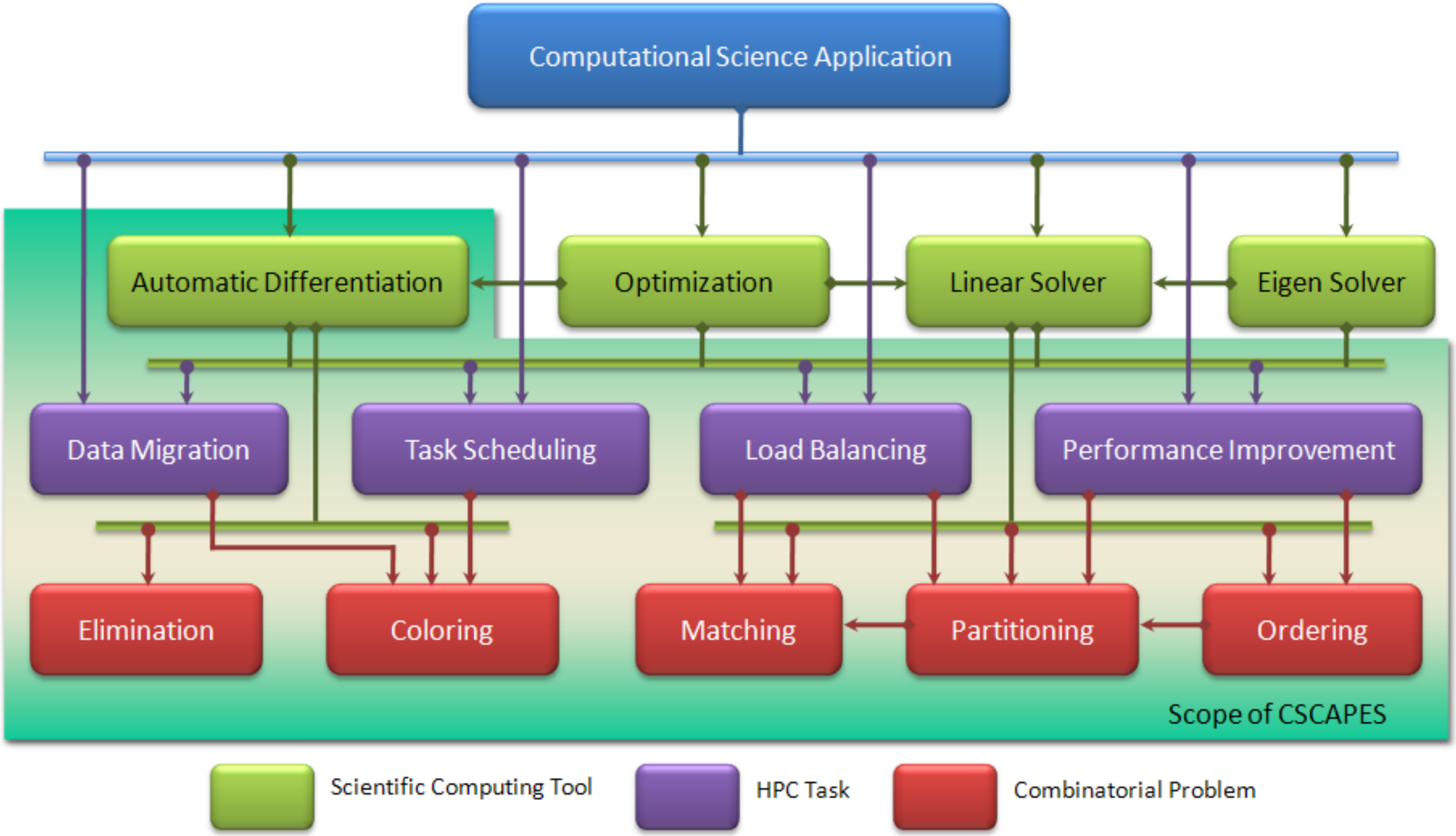
Biology and Medicine

App. Math, CS, Viz., Data

Combinatorial Scientific Computing and Petascale Simulations Institute

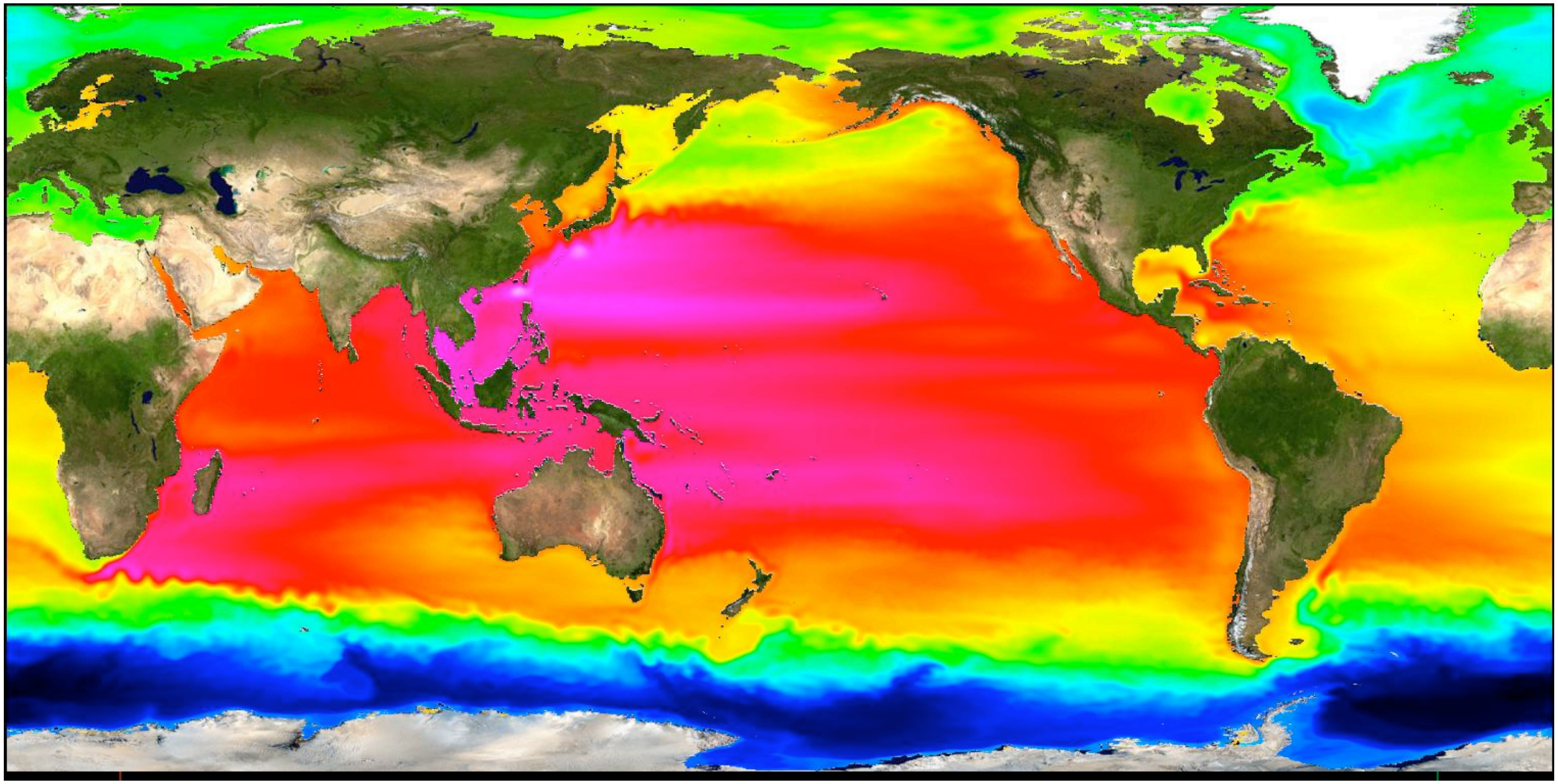
- www.cscapes.org
- One of four Institutes nationally (new in 2006-2012)
  - Excellence in research; Education; Collaborations with science projects and Outreach
- Focus areas of research:
  - Parallelization toolkits; Automatic Differentiation (AD); Parallel Graph Computations. These ‘‘irregular’’ problems need sophisticated algorithms to be scalable on Petascale machines.
- Purdue, Sandia Labs, Argonne Lab, Ohio State
- Popular article in SciDAC Review Fall 2007:
  - www.scidacreview.org/0703/html/cscapes.html

CSCAPES: Scope



Automatic Differentiation for Global Circulation Model (MIT)

- Model state of ocean (velocity, temp, pressure, salinity, surface var. …)
- Time for one simulation run (20 years at 4 degree resolution): 51.75 hrs
- Time for one gradient computation using AD: 204.2 hrs (8.5 days)
- Time to approximate one gradient using finite differences: 1.08 million yrs
- Goal: 10-100 gradient evaluations at 1/2 degree resolution



Computing Derivatives: let me count the ways

- Hand coding
  - Tedious and error-prone
  - Coding time grows with program size and complexity
  - No natural way to compute derivative matrix-vector products
- Divided (Finite) Differencing
  - Incurs truncation errors (is only an approximation)
  - Cost grows with number of independents
  - No natural way to compute transposed-Jacobian-vec products
- Symbolic Differentiation
  - Takes up lots of memory since it relies on first generating symbolic expressions explicitly
  - Does not exploit common sub-expressions directly

Automatic Differentiation overcomes all of these drawbacks.

AD: A One Slide Summary

- Technique for computing analytic derivatives of a function specified as a computer program (could be millions of loc)
- Key ingredients: analytic differentiation of elementary functions plus propagation by the chain rule of calculus
  - Every programming language provides a small set of elementary (intrinsic) mathematical functions
  - Every function computed by a program could be viewed as a composition of intrinsic functions
  - Derivatives of intrinsic functions are obtained by table-lookup, and combined using the chain rule
- Has two main modes (due to associativity of the chain rule):
  - Forward (Tangent) Mode
  - Reverse (Adjoint) Mode
- Can be implemented in one of two ways:
  - Operator overloading
  - Source transformation

Decomposition of function evaluation and its graph representation

Code list

$$v_j = \varphi_j(v_i)_{i \prec j}$$

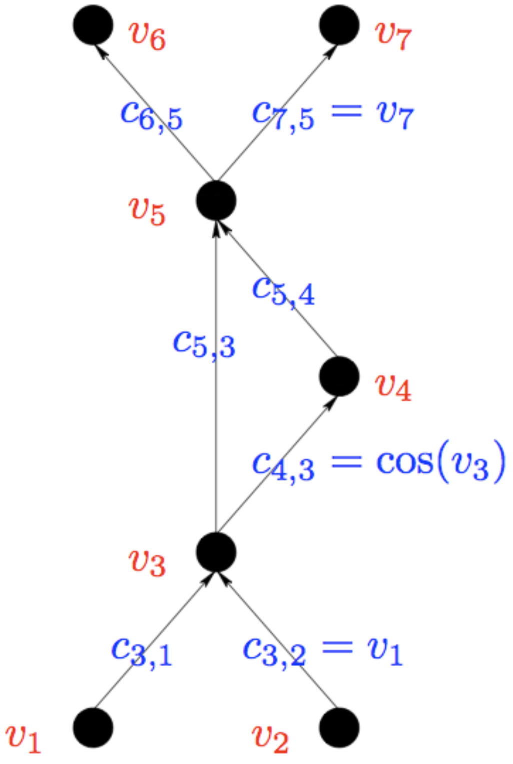
for $j = 1, \ldots, n + p + m$.

Local partial derivatives

$$c_{j,i} = \frac{\partial \varphi_j}{\partial v_i}$$

for $j = 1, \ldots, n + p + m$ and $i \prec j$.

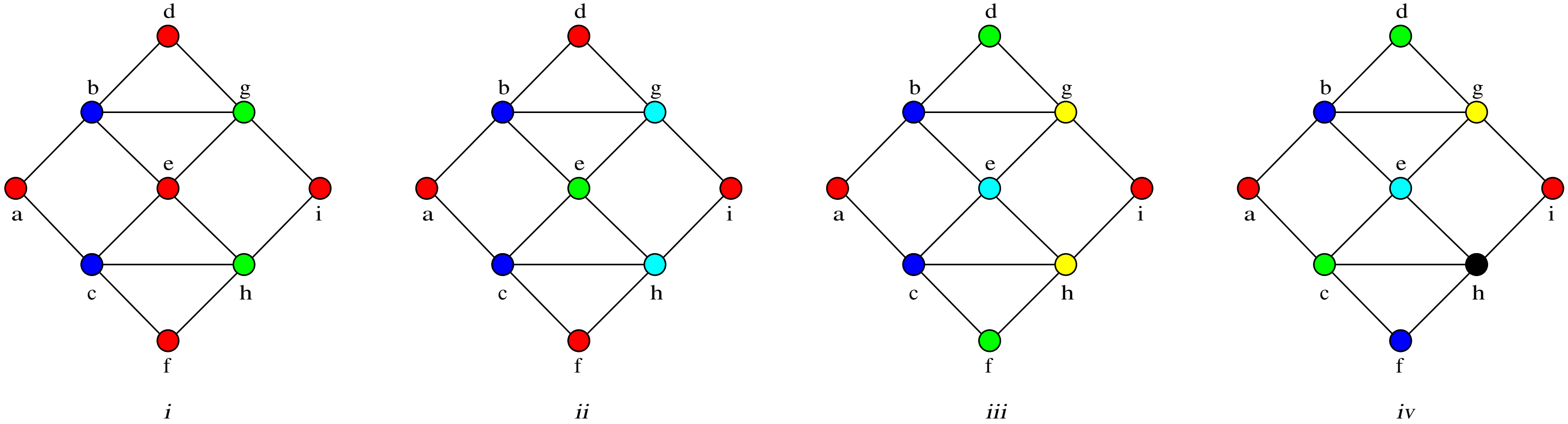
```
v3=v1*v2
v5=v3*sin(v3)  ...  v4=sin(v3);  v5=v3*v4
v6=cos(v5)
v7=exp(v5)
```



n: independents   m: dependents p: intermediates

y1 = cos(x1*x2 * sin(x1*x2))
y2 = exp(x1*x2 * sin(x1*x2))

Coloring problems in AD



- Variants of vertex-coloring problems
- Distance-1, acyclic, star, and distance-2
- Arise in efficient application of AD for computing derivative matrices

CSCAPES Institute: Research Needs

- Petascale Parallel Graph Algorithms
  - Approximation algorithm for graph matching on IBM Blue Gene/P
  - Graph coloring algorithms for use in AD
  - Generators for massive graphs
- Many-core computing
  - Extending ADOL-C to exploit parallelism (Walther)
  - A parallel version of OBLIO (sparse direct solver)
- Applications of AD
  - Pressure Swing Adsorption (purifying gases), process optimization in chemical engineering (Biegler)
- Computational Systems Biology
  - Knowledge discovery from proteomic networks

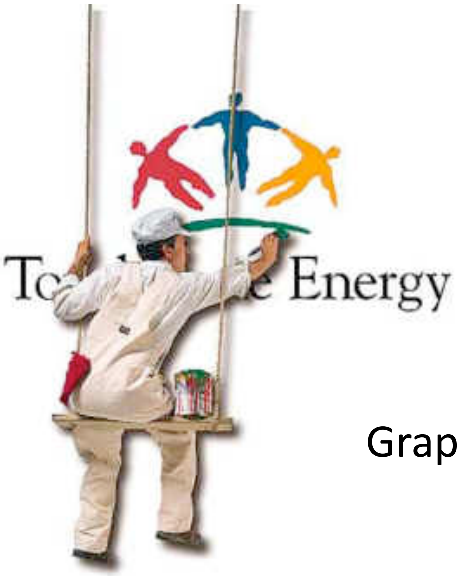
If it takes a village, this is the street on which I live…

cscapes

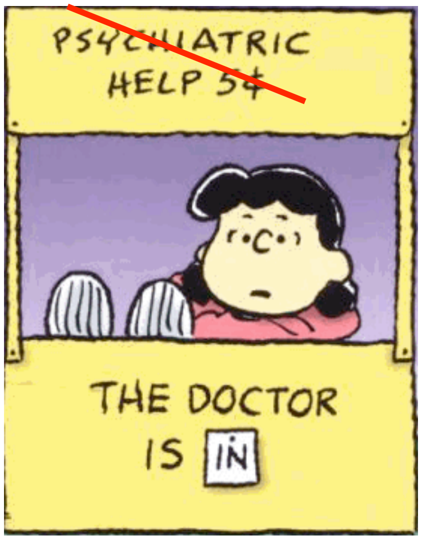
Parallelization,

Load Balancing

Graph Coloring



Combinatorial problems?



Automatic Differentiation

Graph Matching

Succeeding in Grad School: Focus

- Begin with the end in mind.
- Grad school is about…

Succeeding in Grad School: Focus

- Grad school is about…

  learning to do research.

Research in Graduate School

- Begin early! Attend seminars, research group meetings, actively read papers.
- Spend time in choosing a research area and mentor
  - Will you enjoy working in the area for many years?
  - Do the job opportunities in the area fit your desires?
- For more:
  - http://www.cs.umd.edu/users/oleary/gradstudy/gradstudy.html

Learning to Write

- Not something to do after completing your research.
- Learn to write; write to learn.
- Write a short report of your research every week, and have your mentor critique it.
- The CS conference paper culture is somewhat harmful to good writing skills, so be aware.
- Good writing skills will help you in your career, no matter what it is.

Learning to Write Clearly

- Writing clearly:
  - A clear sentence has actors as nouns, actions as verbs.
  - Put what you want to emphasize at the end of the sentence.
  - A paragraph is a collection of sentences, with a logical flow of thought from one to the next,   that explains one idea.
  - Write one section at a time.
  - Organization of a paper at the level of sections needs careful thought (and, often, iteration).
  - Write a section, put it away for a few days, then come back to it to see what a reader might find difficult to follow.

Learning to Write

- For more:
  - Style: Lessons in Clarity and Grace, Joseph M. Williams, 9th Ed., Longman, 2006.
  - Handbook of writing for the Mathematical Sciences, Nicholas J. Higham, Second ed., Society for Industrial and Applied Mathematics, 2008.

Giving a Talk

- Where do you begin?

Giving a Talk

- Where do you begin? Begin with the end in mind!
  - What results do you wish to talk about? What figures describe them?
  - What concepts and background material are needed to explain these results? Include these and omit everything else!
  - A common mistake: too much material, delivered too fast.
- Organize your talk to include the most important results early when the audience is fresh!
- For more: Higham’s book

Take-home Message

- Many-core machines will be on desktops…
  exploiting concurrency will become critical.
- Many problems we face as mankind (energy, climate and environment, personalized medicine, global security) require extreme-scale computational resources. These machines will be built out of commodity many-core chips.
- We are recruiting three faculty in high performance computing!
- We are looking for students who demonstrate aptitude for research, are motivated, and are excited about research in Petascale and many-core computing!