SAND2018-5228PE

Mid-Residency Review

Ethan Epperly

College of Creative Studies, University of California, Santa Barbara

epperly@umail.ucsb.edu

April 24, 2018

Sandia National Laboratories

- A significant part of this presentation will include work I performed with Sandia National Laboratories.
- 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 Energys National Nuclear Security Administration under contract DE-NA0003525.

Overview

My Journey

Physics, Chemistry, and Materials – My First Passion

- My first research experience: molecular dynamics simulations of Metal-Organic Frameworks (MOFs):

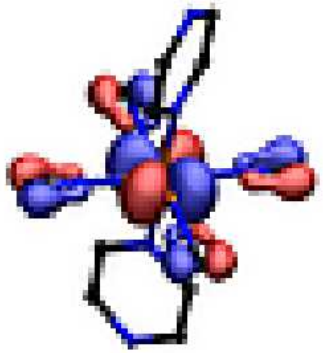

Figure: Hybrid Monte Carlo-molecular dynamics simulations of guest molecule dependent magnetic properties of Fe-based MOFs

Physics, Chemistry, and Materials – My First Passion

- My first research experience: molecular dynamics simulations of Metal-Organic Frameworks (MOFs):

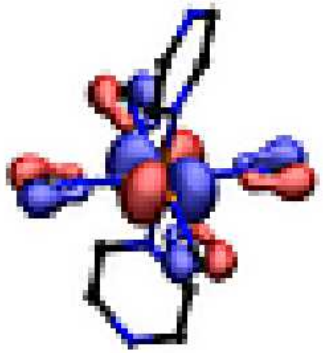

Figure: Hybrid Monte Carlo-molecular dynamics simulations of guest molecule dependent magnetic properties of Fe-based MOFs

- Up until midway through my senior year of high school, was planning on studying Materials Science or Electrical Engineering with a focus on semiconductor device physics and quantum electronics.

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.
- Decided to study what I truly loved with mathematics.

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.
- Decided to study what I truly loved with mathematics.
- Computation as a tool of solving mathematical problems to impractical or impossible to solve by hand.

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.
- Decided to study what I truly loved with mathematics.
- Computation as a tool of solving mathematical problems to impractical or impossible to solve by hand.
- Numerical and scientific computation—an intricate dance between mathematics and computer science:

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.
- Decided to study what I truly loved with mathematics.
- Computation as a tool of solving mathematical problems to impractical or impossible to solve by hand.
- Numerical and scientific computation—an intricate dance between mathematics and computer science:
  - Floating point calculations and round-off error

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.
- Decided to study what I truly loved with mathematics.
- Computation as a tool of solving mathematical problems to impractical or impossible to solve by hand.
- Numerical and scientific computation—an intricate dance between mathematics and computer science:
  - Floating point calculations and round-off error
  - Architecture considerations (Cache-optimal Algorithms, Parallel Architectures, Exoscale Computing)

My Turn Towards Computation

- At some point, I realized that in my studies of physics and engineering, I was really just looking for challenging math problems to solve.
- Decided to study what I truly loved with mathematics.
- Computation as a tool of solving mathematical problems to impractical or impossible to solve by hand.
- Numerical and scientific computation—an intricate dance between mathematics and computer science:
  - Floating point calculations and round-off error
  - Architecture considerations (Cache-optimal Algorithms, Parallel Architectures, Exoscale Computing)
  - Software Design

My Plan (Roughly)

My goal is to become a researcher who develops and implements high-performance algorithms for problems coming from science and engineering.

My Journey:

Past	$\rightarrow$	Present	$\rightarrow$	Future
High School	$\rightarrow$	UCSB	$\rightarrow$	UCSB and PhD Program
Running Codes	$\rightarrow$	Writing Codes	$\rightarrow$	Writing Libraries

Studying at UCSB

First Year Computer Science Curriculum

- Learned C/C++

First Year Computer Science Curriculum

- Learned `C`/`C++`
- Gained a basic understanding of how computers work—architecture, compilation, and operating systems.

First Year Computer Science Curriculum

- Learned `C`/`C++`
- Gained a basic understanding of how computers work—architecture, compilation, and operating systems.
- Strengthened my understanding of core data structures and algorithms—sorting, searching, balanced binary trees, heaps, hash tables, and graphs.

Software Design

- Perhaps the most valuable lesson I learned in computing was the value of good software design principles. In particular,

Software Design

- Perhaps the most valuable lesson I learned in computing was the value of good software design principles. In particular,
  - Object-orient design,

Software Design

- Perhaps the most valuable lesson I learned in computing was the value of good software design principles. In particular,
  - Object-orient design,
  - Data encapsulation,

Software Design

- Perhaps the most valuable lesson I learned in computing was the value of good software design principles. In particular,
  - Object-orient design,
  - Data encapsulation,
  - Descriptive variable names,

Software Design

- Perhaps the most valuable lesson I learned in computing was the value of good software design principles. In particular,
  - Object-orient design,
  - Data encapsulation,
  - Descriptive variable names,
  - Test-driven development.

Software Design

- Perhaps the most valuable lesson I learned in computing was the value of good software design principles. In particular,
  - Object-orient design,
  - Data encapsulation,
  - Descriptive variable names,
  - Test-driven development.
- Many codes in scientific computation are not written with these principles in mind, leading to codes that are hard to maintain and difficult for end users.

Scientific Computation

- When solving continuous mathematical problems on finite-memory computers, all computations involve many forms of error (e.g model error, truncation error, roundoff error, iteration error)

Scientific Computation

- When solving continuous mathematical problems on finite-memory computers, all computations involve many forms of error (e.g model error, truncation error, roundoff error, iteration error)
- Tradeoff between complexity (time and space) and accuracy.

Scientific Computation

- When solving continuous mathematical problems on finite-memory computers, all computations involve many forms of error (e.g model error, truncation error, roundoff error, iteration error)
- Tradeoff between complexity (time and space) and accuracy.
- Mathematical analysis of numerical algorithms requires an understanding of the underlying sensitivity of the problem to perturbations, which will be incurred during computation by various forms of error

Scientific Computation

- When solving continuous mathematical problems on finite-memory computers, all computations involve many forms of error (e.g model error, truncation error, roundoff error, iteration error)
- Tradeoff between complexity (time and space) and accuracy.
- Mathematical analysis of numerical algorithms requires an understanding of the underlying sensitivity of the problem to perturbations, which will be incurred during computation by various forms of error
- Special structure of a problem can be exploited for faster or more accurate algorithms (e.g. positive definite linear systems of equations can be solved with fewer operations and more accuracy than general linear systems of equations).

Scientific Computation

- Took many courses on scientific computation:
  - CMPSC 111 (Introduction to Scientific Computing)
  - CMPSC 211A (Matrix Analysis and Computation)
  - CMPSC 211C (Numerical Solution of Partial Differential Equations by Finite Differences)
  - ECE 594V (Fast Matrix Algorithms)
  - CMPSC 219 (Sparse Matrix Algorithms)
  - ECE 271A (Convex Optimization)
  - Math 132A (Linear Programming)

Scientific Computation

- Took many courses on scientific computation:
  - CMPSC 111 (Introduction to Scientific Computing)
  - CMPSC 211A (Matrix Analysis and Computation)
  - CMPSC 211C (Numerical Solution of Partial Differential Equations by Finite Differences)
  - ECE 594V (Fast Matrix Algorithms)
  - CMPSC 219 (Sparse Matrix Algorithms)
  - ECE 271A (Convex Optimization)
  - Math 132A (Linear Programming)
- And plan to take more:
  - CMPSC 211B (Numerical Solution of Ordinary Differential Equations)
  - CMPSC 211D (Numerical Solution of Partial Differential Equations by Finite Elements)
  - CMPSC 240A (Applied Parallel Computing)

The Strength of Materials

The Strength of Materials

- Since Ancient times, there has been considerable interest in processes to make strong metallic compounds.

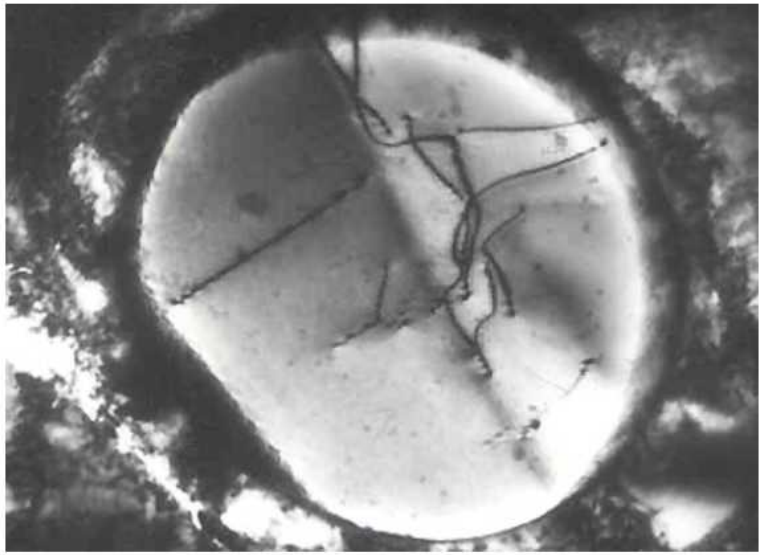

Figure: Transmission electron micrograph of dislocations in Steel [1]

The Strength of Materials

- Since Ancient times, there has been considerable interest in processes to make strong metallic compounds.
- Despite millenia of interest on the subject, the fundamental physics of material strength is still a very active research topic.

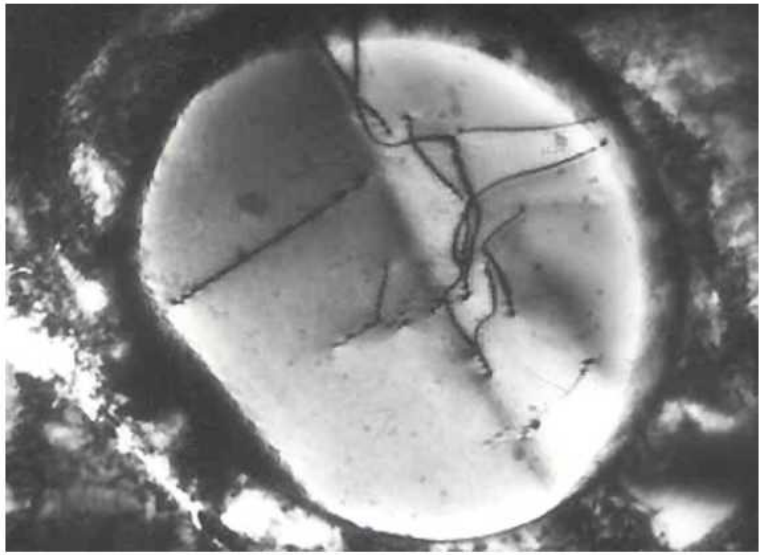

Figure: Transmission electron micrograph of dislocations in Steel [1]

The Strength of Materials

- Since Ancient times, there has been considerable interest in processes to make strong metallic compounds.
- Despite millenia of interest on the subject, the fundamental physics of material strength is still a very active research topic.
- At the microscopic level, material strength is governed by the complicated interplay of one-, two- and three-dimensional defects.

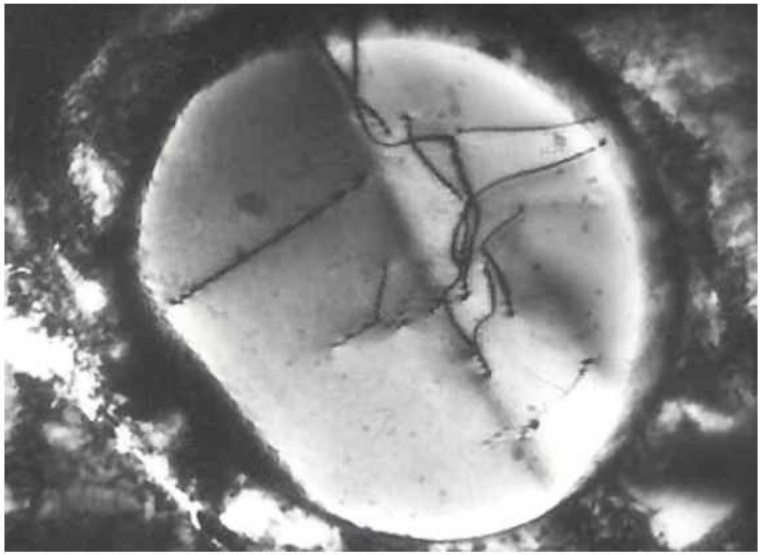

Figure: Transmission electron micrograph of dislocations in Steel [1]

Plastic Deformation of Metals

- For many (but not all) applications, material scientists are interested in materials which do not undergo plastic (permanent) deformation under ordinary use.

Figure: Plastic deformation refers to deformation of a material that remains even after the loads (forces) on the material are removed.

Dislocations Govern Plastic Deformation of Metals

- At a microscopic level, the plastic deformation of metals is governed by the motion of one dimensional defects called **dislocations**.

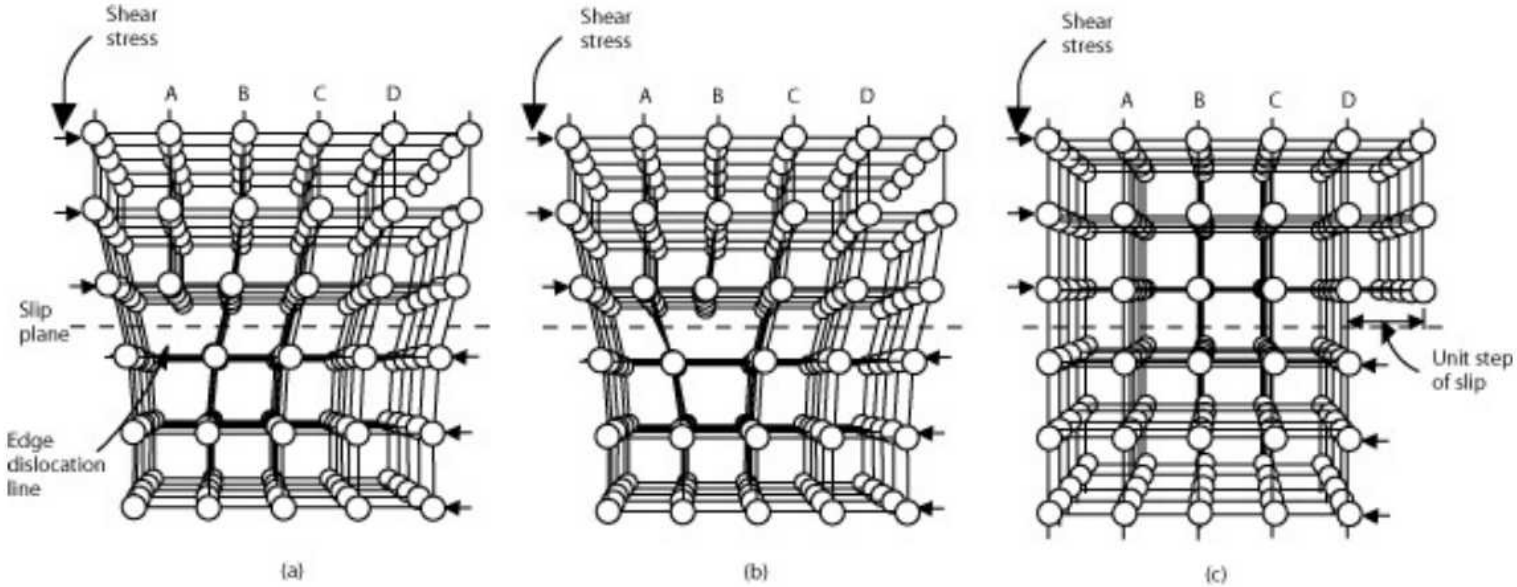

Figure: Plastic deformation refers to deformation of a material that remains even after the loads (forces) on the material are removed.

Defect Micromechanics and Material Strength

- To make a material strong, we introduce intentional defects in a material by alloying it with another material to attempt to impede the motion of dislocations.

Defect Micromechanics and Material Strength

- To make a material strong, we introduce intentional defects in a material by alloying it with another material to attempt to impede the motion of dislocations.
- Many types of defects—solutes, precipitates, cracks, grain boundaries, vacancies—arranged in complex structures make understanding metal strength a challenging scientific problem.

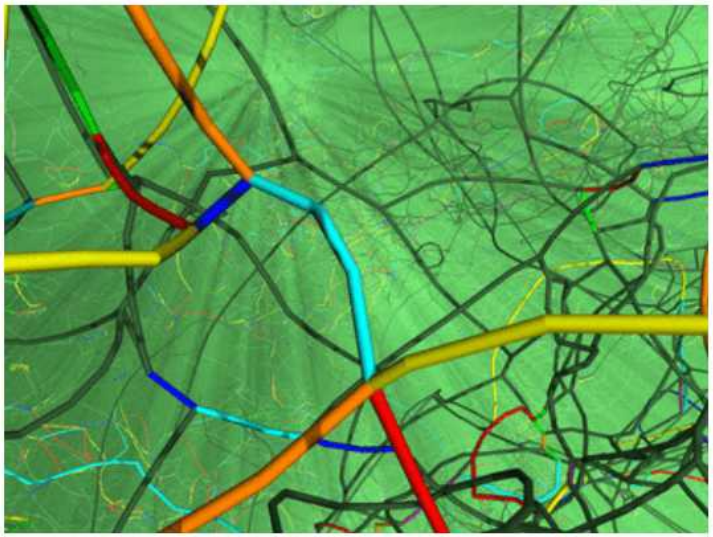

Figure: Computer simulation of dislocation microstructure, [3]

Dislocation Cutting of Precipitates in AlCu – Summer 2015

- My first project at Sandia National Labs, I ran Molecular Dynamics (MD) simulations of dislocations in Aluminum Copper alloys using the LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) code.

Dislocation Cutting of Precipitates in AlCu – Summer 2015

- My first project at Sandia National Labs, I ran Molecular Dynamics (MD) simulations of dislocations in Aluminum Copper alloys using the LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) code.
- Simulations were of dislocations interacting with copper precipitates in Aluminum Copper alloys under different loading conditions.

Dislocation Cutting of Precipitates in AlCu – Summer 2015

- My first project at Sandia National Labs, I ran Molecular Dynamics (MD) simulations of dislocations in Aluminum Copper alloys using the LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) code.
- Simulations were of dislocations interacting with copper precipitates in Aluminum Copper alloys under different loading conditions.
- If enough force was applied to the material, dislocations could "cut" through the copper precipitates leading to significant dislocation motion and therefore plastic deformation.

Dislocation Cutting of Precipitates in AlCu – Summer 2015

- My first project at Sandia National Labs, I ran Molecular Dynamics (MD) simulations of dislocations in Aluminum Copper alloys using the LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) code.
- Simulations were of dislocations interacting with copper precipitates in Aluminum Copper alloys under different loading conditions.
- If enough force was applied to the material, dislocations could "cut" through the copper precipitates leading to significant dislocation motion and therefore plastic deformation.
- Results of my work could be fed into a higher length-scale model to predict bulk material behavior.

Dislocation Cutting of Precipitates in AlCu – Summer 2015

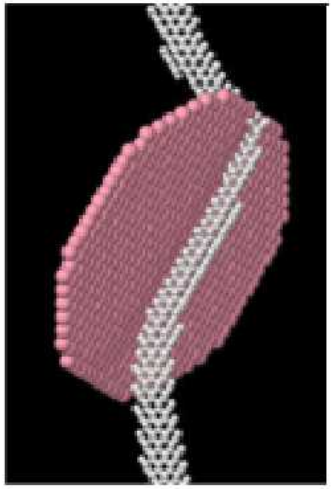

Figure: A visualization of a dislocation pinned at a copper precipitate. If a higher force was applied, dislocation could cut through copper precipitate and move unimpeded.

- Work published in a technical report, but never a peer-reviewed academic publication.

Dislocation Solute Interactions – Summer 2016

- My next summer, I began work with Ryan Sills on understanding the interactions between dislocations and solute atoms

Dislocation Solute Interactions – Summer 2016

- My next summer, I began work with Ryan Sills on understanding the interactions between dislocations and solute atoms
- **Solute**: Single atom of material Y in a lattice of material X.

Dislocation Solute Interactions – Summer 2016

- My next summer, I began work with Ryan Sills on understanding the interactions between dislocations and solute atoms
- **Solute**: Single atom of material Y in a lattice of material X.
- Dislocations create local distortions in the atomic lattice leading to solutes congregating around dislocations forming **solute atmospheres**.

Dislocation Solute Interactions – Summer 2016

- My next summer, I began work with Ryan Sills on understanding the interactions between dislocations and solute atoms
- **Solute**: Single atom of material Y in a lattice of material X.
- Dislocations create local distortions in the atomic lattice leading to solutes congregating around dislocations forming **solute atmospheres**.
- Solute atmospheres "pull" on dislocations, slowing their movement.

Dislocation Solute Interactions – Summer 2016

- My next summer, I began work with Ryan Sills on understanding the interactions between dislocations and solute atoms
- **Solute**: Single atom of material Y in a lattice of material X.
- Dislocations create local distortions in the atomic lattice leading to solutes congregating around dislocations forming **solute atmospheres**.
- Solute atmospheres "pull" on dislocations, slowing their movement.
- This phenomena of **solute drag** in part explains why alloys of two metals can be stronger than either metal individually. Solute atoms from the alloying metal impede the motion of dislocations!

Dislocation Solute Interactions – Summer 2016

- My next summer, I began work with Ryan Sills on understanding the interactions between dislocations and solute atoms
- **Solute**: Single atom of material Y in a lattice of material X.
- Dislocations create local distortions in the atomic lattice leading to solutes congregating around dislocations forming **solute atmospheres**.
- Solute atmospheres "pull" on dislocations, slowing their movement.
- This phenomena of **solute drag** in part explains why alloys of two metals can be stronger than either metal individually. Solute atoms from the alloying metal impede the motion of dislocations!
- Despite its critical importance in understanding strength of alloys, much remains to be known about solute drag.

Dislocation Solute Interactions – Summer 2016

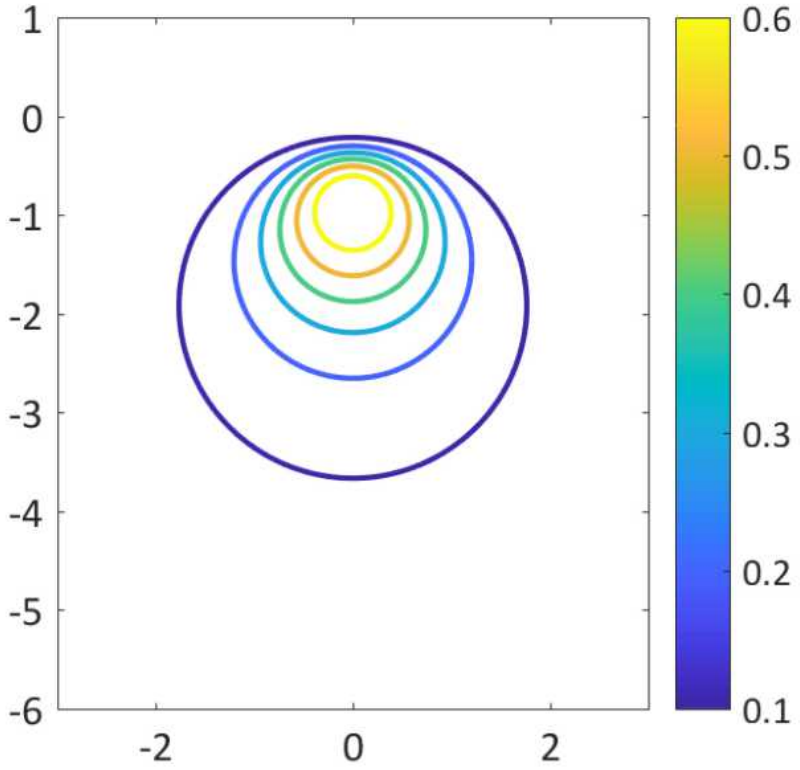

Figure: The concentration of solutes around a stationary edge dislocation located at $(0, 0)$. In the region of peek concentration, solutes occupy around 60% of the theoretical maximum occupancy.

Dislocation Solute Interactions – Summer 2016

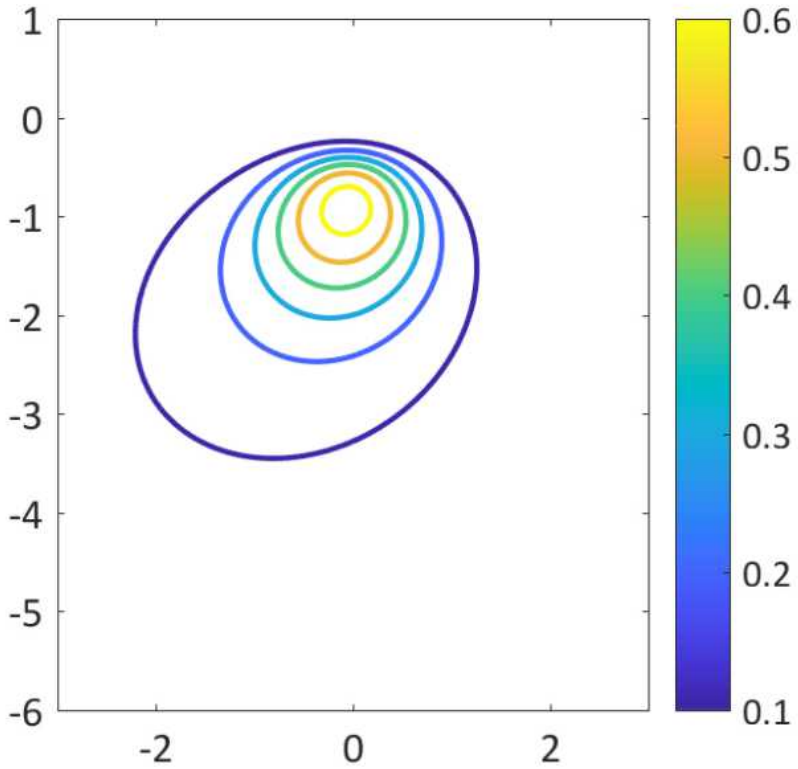

Figure: The concentration of solutes around a right-moving edge dislocation located at $(0, 0)$. In the region of peek concentration, solutes still occupy around 60% of the theoretical maximum occupancy, but the region of peek occupancy is much smaller than for the stationary dislocation.

Dislocation Solute Interactions – Summer 2016

- Two big questions:

Dislocation Solute Interactions – Summer 2016

- Two big questions:
  - How quickly do the solute atmospheres on dislocations condense?

Dislocation Solute Interactions – Summer 2016

- Two big questions:
  - How quickly do the solute atmospheres on dislocations condense?
  - How strong is the solute drag on a dislocation as it moves through the material?

Dislocation Solute Interactions – Summer 2016

- Two big questions:
  - How quickly do the solute atmospheres on dislocations condense?
  - How strong is the solute drag on a dislocation as it moves through the material?
- My job was to write and interpret codes to solve the partial differential equations governing the time evolution of solute-dislocation systems.

$$\frac{\partial \chi}{\partial t} = \nabla \cdot \left(\frac{D\chi}{k_B T} \nabla \mu_{\text{chem}} + \chi v \right)$$

Dislocation Solute Interactions – Winter 2016-present

- Our original code was in Matlab. To run more simulations, we wanted to rewrite our code in C++ to be faster and to run on the supercomputers.

Dislocation Solute Interactions – Winter 2016-present

- Our original code was in Matlab. To run more simulations, we wanted to rewrite our code in C++ to be faster and to run on the supercomputers.
- Porting the code to C++ allowed me to apply the software design principles I had learned to a scientific computation problem.

Dislocation Solute Interactions – Winter 2016-present

- Our original code was in Matlab. To run more simulations, we wanted to rewrite our code in `C++` to be faster and to run on the supercomputers.
- Porting the code to `C++` allowed me to apply the software design principles I had learned to a scientific computation problem.
- Unfortunately, the `C++` code was only marginally faster than the Matlab code as both my `C++` code and the Matlab code both called the same library, `UMFPACK`, to do sparse matrix computations, which constituted the lion's share of the runtime.

Dislocation Solute Interactions – Winter 2016-present

- Our original code was in Matlab. To run more simulations, we wanted to rewrite our code in `C++` to be faster and to run on the supercomputers.
- Porting the code to `C++` allowed me to apply the software design principles I had learned to a scientific computation problem.
- Unfortunately, the `C++` code was only marginally faster than the Matlab code as both my `C++` code and the Matlab code both called the same library, `UMFPACK`, to do sparse matrix computations, which constituted the lion's share of the runtime.
    - Lesson learned: Profile before trying to optimize.

Dislocation Solute Interactions – Winter 2016-present

- Our original code was in Matlab. To run more simulations, we wanted to rewrite our code in C++ to be faster and to run on the supercomputers.
- Porting the code to C++ allowed me to apply the software design principles I had learned to a scientific computation problem.
- Unfortunately, the C++ code was only marginally faster than the Matlab code as both my C++ code and the Matlab code both called the same library, UMFPACK, to do sparse matrix computations, which constituted the lion's share of the runtime.
  - Lesson learned: Profile before trying to optimize.
- Writing the C++ code was still very helpful as it allowed us to run hundreds of simulations on the supercomputer.

Dislocation Solute Interactions – Winter 2016-present

- Towards a fundamental equation of solute atmosphere condensation:

Dislocation Solute Interactions – Winter 2016-present

- Towards a fundamental equation of solute atmosphere condensation:
  - Running hundreds of different atmosphere condensation simulations allows us to discover a general equation for the dynamics of atmosphere condensation for any arbitrary material system.

Dislocation Solute Interactions – Winter 2016-present

- Towards a fundamental equation of solute atmosphere condensation:
  - Running hundreds of different atmosphere condensation simulations allows us to discover a general equation for the dynamics of atmosphere condensation for any arbitrary material system.
  - Such an equation will necessarily be approximate and a deeply simplified model of reality, but can provide useful insights on the design and analysis of real-world devices.

Dislocation Solute Interactions – Winter 2016-present

- Towards a fundamental equation of solute atmosphere condensation:
  - Running hundreds of different atmosphere condensation simulations allows us to discover a general equation for the dynamics of atmosphere condensation for any arbitrary material system.
  - Such an equation will necessarily be approximate and a deeply simplified model of reality, but can provide useful insights on the design and analysis of real-world devices.
- Final results of this work are being tidied up just now with the intention of submitting a paper to a journal soon.

Dislocation Solute Interactions – Winter 2016-present

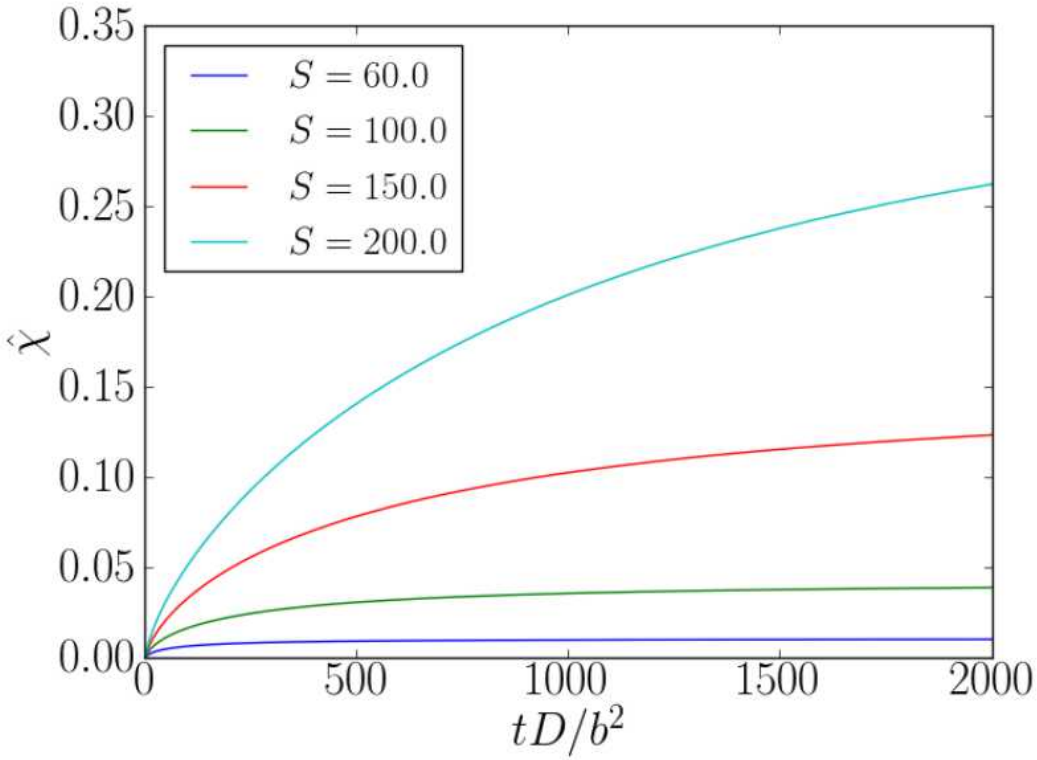

Figure: Concentration versus time for a variety of different material systems (whose properties are governed by the quantity S) in terms of a nondimensionalized time Dt/b^2.

The Future

Computational Materials Science – Summer 2018

- Will continue my work with Ryan this summer using computers to deepen our understanding of the fundamental physics of materials.

Computational Materials Science – Summer 2018

- Will continue my work with Ryan this summer using computers to deepen our understanding of the fundamental physics of materials.
- Potential projects might include:

Computational Materials Science – Summer 2018

- Will continue my work with Ryan this summer using computers to deepen our understanding of the fundamental physics of materials.
- Potential projects might include:
  - Extending solute-dislocation work to get a better understanding of the interplay between atmosphere formation and dislocation motion.

Computational Materials Science – Summer 2018

- Will continue my work with Ryan this summer using computers to deepen our understanding of the fundamental physics of materials.
- Potential projects might include:
  - Extending solute-dislocation work to get a better understanding of the interplay between atmosphere formation and dislocation motion.
  - Dislocation dynamics: Simulating the trajectories of large ensembles of dislocations. Bulk material behavior is governed by a complicated mess of dislocation networks.

Algebraic Multigrid – Summer 2018

- Will begin work with another scientist, Jonathan Hu, who works on developing algebraic multigrid software to run on massive parallel supercomputers.

Algebraic Multigrid – Summer 2018

- Will begin work with another scientist, Jonathan Hu, who works on developing algebraic multigrid software to run on massive parallel supercomputers.
- What is algebraic multigrid?

Algebraic Multigrid – Summer 2018

- Will begin work with another scientist, Jonathan Hu, who works on developing algebraic multigrid software to run on massive parallel supercomputers.
- What is algebraic multigrid?
  - The goal: solve systems of linear systems of equations $Ax = b$ in which the matrix A is "sparse" in the sense that most of the entries in A are zero. (Typically, if A is $n \times n$, then there are only $O(n)$ nonzero entries in A)

Algebraic Multigrid – Summer 2018

- Will begin work with another scientist, Jonathan Hu, who works on developing algebraic multigrid software to run on massive parallel supercomputers.
- What is algebraic multigrid?
  - The goal: solve systems of linear systems of equations $Ax = b$ in which the matrix A is "sparse" in the sense that most of the entries in A are zero. (Typically, if A is $n \times n$, then there are only $O(n)$ nonzero entries in A)
  - Gaussian Elimination from Math 4A: $O(n^3)$ algorithm. Can we do better?

Algebraic Multigrid – Summer 2018

- Will begin work with another scientist, Jonathan Hu, who works on developing algebraic multigrid software to run on massive parallel supercomputers.
- What is algebraic multigrid?
  - The goal: solve systems of linear systems of equations $Ax = b$ in which the matrix A is "sparse" in the sense that most of the entries in A are zero. (Typically, if A is $n \times n$, then there are only $O(n)$ nonzero entries in A)
  - Gaussian Elimination from Math 4A: $O(n^3)$ algorithm. Can we do better?
  - Iterative Methods: Produce a sequence of guesses x_k to the solution of $Ax = b$. Hope that after some number N of iterations, x_N is "close" to x.

Algebraic Multigrid – Summer 2018

- Will begin work with another scientist, Jonathan Hu, who works on developing algebraic multigrid software to run on massive parallel supercomputers.
- What is algebraic multigrid?
  - The goal: solve systems of linear systems of equations $Ax = b$ in which the matrix A is "sparse" in the sense that most of the entries in A are zero. (Typically, if A is $n \times n$, then there are only $O(n)$ nonzero entries in A)
  - Gaussian Elimination from Math 4A: $O(n^3)$ algorithm. Can we do better?
  - Iterative Methods: Produce a sequence of guesses x_k to the solution of $Ax = b$. Hope that after some number N of iterations, x_N is "close" to x.
  - Convergence of iterative methods is slow. Different "flavors" of error are eliminated at different rates, with "fine grain" error eliminated much more quickly then "coarse grain" error.

Algebraic Multigrid – Summer 2018

- The hope of algebraic multigrid

- The hope of algebraic multigrid
  - Our original problem $Ax = b$ can be "coarsened" into a smaller system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$.

- The hope of algebraic multigrid
  - Our original problem $Ax = b$ can be "coarsened" into a smaller system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$.
  - By using the iterative method on the coarser system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$, the error will be eliminated more quickly than in the original system of equations $Ax = b$.

- The hope of algebraic multigrid
  - Our original problem $Ax = b$ can be "coarsened" into a smaller system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$.
  - By using the iterative method on the coarser system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$, the error will be eliminated more quickly than in the original system of equations $Ax = b$.
  - Using a multilevel scheme where we transition between coarser and less coarse equations, can we solve $Ax = b$ quickly?

- The hope of algebraic multigrid
  - Our original problem $Ax = b$ can be "coarsened" into a smaller system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$.
  - By using the iterative method on the coarser system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$, the error will be eliminated more quickly than in the original system of equations $Ax = b$.
  - Using a multilevel scheme where we transition between coarser and less coarse equations, can we solve $Ax = b$ quickly?
  - Often, yes! Multigrid methods are some of the fastest known methods of solving sparse linear systems of equations, often able to solve such systems in optimal $O(n)$ time.

Algebraic Multigrid – Summer 2018

- The hope of algebraic multigrid
  - Our original problem $Ax = b$ can be "coarsened" into a smaller system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$.
  - By using the iterative method on the coarser system of linear equations $\tilde{A}\tilde{x} = \tilde{b}$, the error will be eliminated more quickly than in the original system of equations $Ax = b$.
  - Using a multilevel scheme where we transition between coarser and less coarse equations, can we solve $Ax = b$ quickly?
  - Often, yes! Multigrid methods are some of the fastest known methods of solving sparse linear systems of equations, often able to solve such systems in optimal $O(n)$ time.
- Will be working on some problem related to algebraic multigrid, but I'm not sure exactly what yet.

Goals for the Next Two Years

- **Expand my ability to write parallel software.** Almost all of my work so far has been in serial computation. Given the prevalence of massive parallel computation in scientific computing, this is a pretty glaring oversight.

Goals for the Next Two Years

- **Expand my ability to write parallel software.** Almost all of my work so far has been in serial computation. Given the prevalence of massive parallel computation in scientific computing, this is a pretty glaring oversight.
- **Continue to improve as a software developer.** Looking forward to taking CMPSC 160, 170, and 240A, which I hope will provide significant coding challenges.

Goals for the Next Two Years

- **Expand my ability to write parallel software.** Almost all of my work so far has been in serial computation. Given the prevalence of massive parallel computation in scientific computing, this is a pretty glaring oversight.
- **Continue to improve as a software developer.** Looking forward to taking CMPSC 160, 170, and 240A, which I hope will provide significant coding challenges.
- **Maybe learn some more machine learning.** Machine learning is a giant application area for numerical computation, with many machine learning algorithms requiring intense use of linear algebraic computations.

Beyond UCSB

- Intending to pursue a PhD in Applied Mathematics or Computer Science with a focus on matrix computation, optimization, or numerical partial differential equations.

Beyond UCSB

- Intending to pursue a PhD in Applied Mathematics or Computer Science with a focus on matrix computation, optimization, or numerical partial differential equations.
- Potential careers:

Beyond UCSB

- Intending to pursue a PhD in Applied Mathematics or Computer Science with a focus on matrix computation, optimization, or numerical partial differential equations.
- Potential careers:
  - Professor,

Beyond UCSB

- Intending to pursue a PhD in Applied Mathematics or Computer Science with a focus on matrix computation, optimization, or numerical partial differential equations.
- Potential careers:
  - Professor,
  - Industry Research,

Beyond UCSB

- Intending to pursue a PhD in Applied Mathematics or Computer Science with a focus on matrix computation, optimization, or numerical partial differential equations.
- Potential careers:
  - Professor,
  - Industry Research,
  - Government Research.

References

Wikityke

Wikityke at the English language Wikipedia [CC BY-SA 2.5 (https://creativecommons.org/licenses/by-sa/2.5), GFDL (http://www.gnu.org/copyleft/fdl.html) or CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/)], via Wikimedia Commons

NDT Resource Center

Dislocation multi-junctions and strain hardening, Nature, 440, 1174 (2006). Graphics by Meijie Tang, Rich Cook, Sean Ahern (LLNL).