Combining Multi-Grid and Domain Decomposition as Preconditioners for a Class of Multi-Level PDE-Constrained Optimization Problems

SAND2012-6744C

Paul T. Boggs

Sandia National Laboratories
Livermore, CA 94551, USA

ISMP 2012

August 19 – July 24, 2011

Sandia
National
Laboratories
A Department of Energy
National Laboratory

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

Outline

Collaborators

- David Gay (AMPL Corp.)
- R. Michael Lewis (College of William and Mary)
- Kevin Long (Texas Tech University)
- Stephen Nash (George Mason University)
- Julien Cortial (Sandia National Laboratories)

Outline

Domain Decomposition to Extend a Hierarchical (Multi-Grid) Optimization Algorithm

- In our problems, we have a natural hierarchy of levels with the physics described by a PDE at each level
- The optimize problem is formulated on the finest level
- We use the coarser levels in a multi-grid optimization (MG/Opt) algorithm (speeds convergence)
- To use MG/Opt, we have to iterate on finest level, but it will be too large, i.e., the problem is at the nano scale

We have developed a preliminary DD/Opt strategy that blends with the MG/Opt algorithm to extend the range of applicability

Lots left to do

Simple grid

Picture of domain and subdomains, including overlap

Problems and Domains

- Let Ω be the whole domain and Ω_i the i^{th} subdomain
- Assume that there are S subdomains
- Let f be objective function on Ω and f_i the objective on Ω_i
- Let x be all of the variables on Ω and x_i the variables on Ω_i
- Let x^0 be the intial guess

The Domain Decomposition Optimization (DD/Opt) Algorithm for Unconstrained Problem

Set $n = 0$

1. Extract x_i^n from x^n for each i
2. Perform k_1 iterations on each subproblem i to obtain $x_i^{k_1}$; compute the local search direction $d_i = x_i^{k_1} - x_i^n$
3. Assemble the search direction into gloabal search direction $d = (d_1, \ldots, d_S)^T$ (handling the overlap is an issue)
4. Perform k_2 iterations on

   $$\min_s \frac{1}{2} s^T B s + s^T d,$$

   where B is an approximation to $\nabla^2 f(x^n)$
5. Perform a linesearch on $f(x^n + \alpha s)$
6. Check convergence; n := n + 1; repeat as necessary

Comments on the Algorithm

- If $k_1 = 0$ then the direction d_i is taken to be $-\nabla f(x_i^n)$, i.e., steepest descent
- If $k_1 \neq 0$ then d_i can be considered to be a *preconditioned gradient*
- If $k_2 = 0$ then $s = d$
- If $k_2 \neq 0$ then we can solve with a preconditioned CG method using the subdomain solves as the preconditioner

Issues Being Investigated

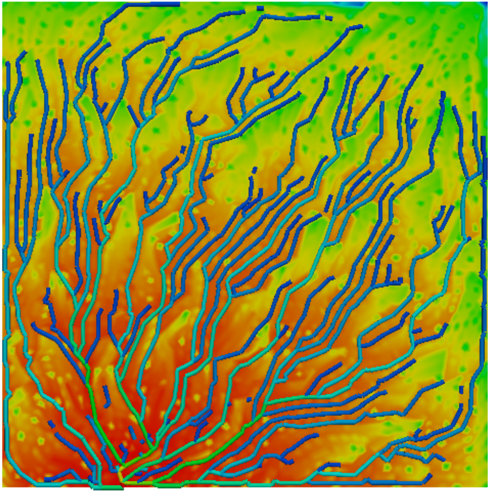
- How to set up and solve the subdomain problems
  e.g., set subdomain BCs, handle the overlap
- How many iterations (i.e., k_1 and k_2)
- How to iterate on the full problem
  - Need to evaluate the function and gradient, i.e., solve PDE for fixed value of design variables
  - Use DD along with MG/Opt to facilitate this evaluation
- How to deal with the constraints
  - Step 4 becomes a quadratic programming problem
  - Bounds have to be shifted
  - Other constraints have to be distributed

The Motivating Problem

(See next talk by Dave Gay for the details)

- We have a PDE-Constrained optimization problem
- Motivated by gas storage in a nanoporous material
- Want to put channels into material to facilitate charge/discharge of the gas

A Channel Network in a Nanoporous Material



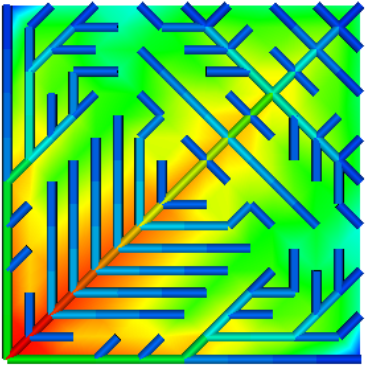
The Motivating Problem

- Trade-off between speed of charge and volume of material devoted to channels
- Our model allows flow along edges of finite element grid
- We want wide main channels and fine subchannels
- We create a hierarchy of subgrids to effect this

The Motivating Problem

- The will be many scales from macro to nano
- The physics will change as we refine the scale
- Can pose problem on finest scale, but it will be very large
- Question: Can we use the structure to create more efficient algorithm?

Example Showing Tree Structure



The Tree Constraint

- We use a network approximation to the physics to obtain a starting guess
- Can show that the optimal network will be a tree structure
- Thus, we want to maintain a tree constraint
  (This, of course, is an integer constraint)
- Our MG/Opt procedure starts with a tree and then "locks" into a tree after a small number of iterations

The Tree Constraint

- Hard to maintain tree structure in DD/Opt
  Can't just do it locally because of boundary problems, i.e., disconnects and occur across boundaries on the subdomains
- A subdomain may see no need for an edge to be maintained
- We developed a global "treeifying" procedure that settles on a tree by fast, local method, i.e., it can be parallelized

Some Results

(Will put in some results soon)

- Problem
- size
- solve by lmbfgs
- solve by DD
- solve in parallel

Outline

1 The Domain Decomposition Optimization (DD/Opt) Algorithm

2 Status and Future Work

Status

- Have paper describing χOpt accepted in SIOPT
- We have derived a basic convergence analysis of χOpt with constraints (Nash)
- We have a reasonable implementation of the software running
  - Uses Sundance to handle the PDEs
  - Have run a variety of tests
  - Have designed and implemented an automatic mesh refinement algorithm
  - Have coded the special update and downdate procedures making use of automatic refinement
  - Have added strategies to handle the "integer" nature of the problem, including the initial guess
  - Have developed DD/Opt that can be incorporated in this framework

Future Work

- Investigate strategies for handling constraints, especially inequality constraints, e.g., arising from manufacturability requirements
- Incorporate DD/Opt into χ*Opt*

Thanks!

Research supported by grant from Office of Advanced Scientific Computing Research, U.S. Department of Energy