The M&M of Optimization: Modeling and Methods

Sven Leyffer leyffer@mcs.anl.gov

Mathematics & Computer Science Division,
Argonne National Laboratory

Optimization Applications

Basic Ingredient: Newton's Method

Optimization Modeling Languages

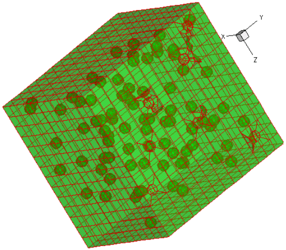
Other Flavours of Optimization

Conclusions

OPTIMIZATION APPLICATIONS

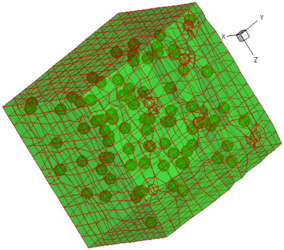
Optimal Mesh Smoothing [Munson]

- optimize quality of meshes for solving PDEs
- reposition vertices of mesh (no new elements)
- regular vs. optimal mesh: reduce solve time by 30% $\equiv$ 10 hours!
- optimization solved in minutes



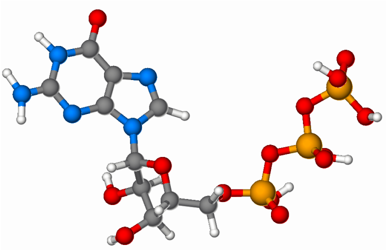
Optimal Mesh Smoothing [Munson]

- optimize quality of meshes for solving PDEs
- reposition vertices of mesh (no new elements)
- regular vs. optimal mesh: reduce solve time by 30% $\equiv$ 10 hours!
- optimization solved in minutes



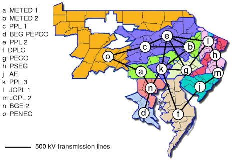
Molecular Geometry Optimization [Benson]

- molecule's functionality determined by geometry
  $\Rightarrow$ fundamental problem in computational chemistry
- stable geometries $\equiv$ least energy state
- TAO: parallel optimization in NWChem
  $\Rightarrow$ $2\times$ faster & more robust



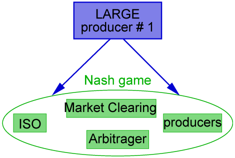
Electric Power and NO_x Markets [L & Munson]

- Interaction of power & NO_x allowance markets in Eastern US
- California power crisis: NO_x over-consumption $\Rightarrow$ price increases



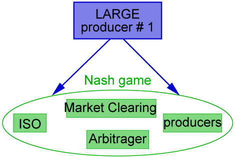
- ○ NO_x market: cap-and-trade program
- ○ Controls ground-level ozone in summer
- ○ Allowances distributed at start of cycle
- ○ Generators redeem allowances to cover emissions
- ○ Secondary market for NO_x allowance

Electric Power and NO_x Markets [L & Munson]



- multi-level optimization ⇒ tough
- Market shares: 6–19% of capacity
- Leader is largest company
- Cournot followers: influence price
- Price taking followers, ISO & arbitrager

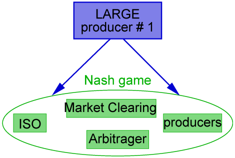
Electric Power and NO_x Markets [L & Munson]



- multi-level optimization ⇒ tough
- Market shares: 6–19% of capacity
- Leader is largest company
- Cournot followers: influence price
- Price taking followers, ISO & arbitrager

- Engineering data: heat rates, emission rates, fuel costs
- 14 nodes, 18 arcs, and 5 periods; 6 larger companies
  ⇒ 20,000 vars and 10,000 cons

Electric Power and NO_x Markets [L & Munson]



- ◦ multi-level optimization ⇒ tough
- ◦ Market shares: 6–19% of capacity
- ◦ Leader is largest company
- ◦ Cournot followers: influence price
- ◦ Price taking followers, ISO & arbitrager

- Engineering data: heat rates, emission rates, fuel costs
- 14 nodes, 18 arcs, and 5 periods; 6 larger companies
  ⇒ 20,000 vars and 10,000 cons

leader exploits market power to drive up NO_x prices

Other Cool Applications

- Data assimilation in weather forecasting
- Image reconstruction from acoustic wave data
- Crew scheduling, vehicle routing
- Nuclear reactor core reloading
- Radio therapy treatment planning
- Oil field infrastructure design

$\Rightarrow$ wide range of applications; permeates scientific computing

Other Cool Applications

- Data assimilation in weather forecasting
- Image reconstruction from acoustic wave data
- Crew scheduling, vehicle routing
- Nuclear reactor core reloading
- Radio therapy treatment planning
- Oil field infrastructure design

$\Rightarrow$ wide range of applications; permeates scientific computing

... and even fun: optimization solves sudoku

Other Cool Applications

- Data assimilation in weather forecasting
- Image reconstruction from acoustic wave data
- Crew scheduling, vehicle routing
- Nuclear reactor core reloading
- Radio therapy treatment planning
- Oil field infrastructure design

$\Rightarrow$ wide range of applications; permeates scientific computing

... and even fun: optimization solves sudoku

ANL-MCS optimization: Anitescu, Moré, Munson & L

Nonlinear Optimization Problem

Nonlinear programming (NLP) problem

$$\begin{cases} \underset{x}{\text{minimize}} & f(x) & \text{objective} \\ \text{subject to} & c(x) = 0 & \text{constraints} \\ & x \geq 0 & \text{variables} \end{cases}$$

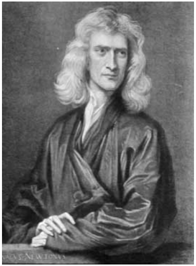
Assumptions:

1. $x, c(x)$ finite dimensional
2. gradients (& Hessians) available
3. functions are smooth

NEWTON'S METHOD

Newton's Method

... everybody's favourite method for nonlinear equations ...



Solve $F(x) = 0$:
Get approx. x_{k+1} of solution of $F(x) = 0$ by solving linear model about x_k:

$$F(x_k) + \nabla F(x_k)^T (x - x_k) = 0$$

for $k = 0, 1, \ldots$

... converges quadratically near a solution
... most nonlinear solvers (IPM/SQP) based on this idea

Sequential Quadratic Programming (SQP)

Nonlinear optimization problem

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } c(x) = 0 \quad \& \quad x \geq 0$$

repeat

1. Solve quadratic approx^n for (s, y_{k+1}, z_{k+1})

$$\begin{cases} \underset{s}{\text{minimize}} & \nabla f_k^T s + \frac{1}{2} s^T H_k s \\ \text{subject to} & c_k + \nabla c_k^T s = 0 \qquad (\perp \; y_{k+1}) \\ & x_k + s \geq 0 \qquad (\perp \; z_{k+1} \geq 0) \end{cases}$$

2. Set $x_{k+1} = x_k + s$, & $k = k + 1$

until convergence

Fast quadratic convergence near x^*

Modern Interior Point Methods (IPM)

General NLP

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } c(x) = 0 \quad \& \quad x \geq 0$$

Perturbed $\mu > 0$ optimality conditions $(x, z \geq 0)$

$$F_\mu(x, y, z) = \left\{ \begin{array}{r} \nabla f(x) - \nabla c(x)^T y - z \\ c(x) \\ Xz - \mu e \end{array} \right\} = 0$$

- Primal-dual formulation
- Central path $\{x(\mu), y(\mu), z(\mu) \; : \; \mu > 0\}$
- Apply Newton's method for sequence $\mu \searrow 0$

Modern Interior Point Methods (IPM)

repeat

1. Choose $\mu_k \searrow 0$ tol for $F_{\mu_k}(x, y, z) = 0$
2. Apply Newton to primal-dual system ...

$$\begin{bmatrix} \nabla^2 \mathcal{L}_k & -A_k & -I \\ A_k^T & 0 & 0 \\ Z_k & 0 & X_k \end{bmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = -F_\mu(x_k, y_k, z_k)$$

   where $A_k = \nabla c(x_k)^T$, X_k diagonal matrix of x_k.

3. Set $x_{k+1} = x_k + \Delta x$, ... & $k = k + 1$

until convergence

Path-following (homotopy) method
Polynomial run-time guarantee for convex problems

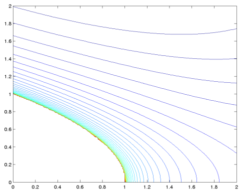
Interior Point Methods (IPM)

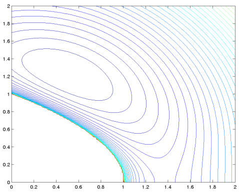
$$\underset{x}{\text{minimize}} \ f(x) \quad \text{subject to } c(x) = 0 \quad \& \quad x \geq 0$$

Related to classical barrier methods [Fiacco & McCormick]

$$\begin{cases} \underset{x}{\text{minimize}} & f(x) - \mu \sum \log(x_i) \\ \text{subject to} & c(x) = 0 \end{cases}$$

$\mu = 10$ $\mu = 1$





$$\text{minimize } x_1^2 + x_2^2 \quad \text{subject to} \quad x_1 + x_2^2 \geq 1$$

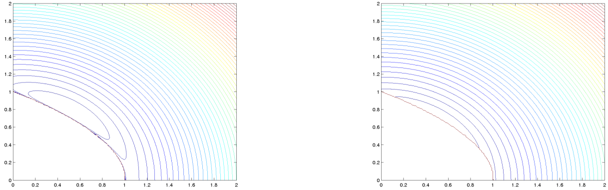
Interior Point Methods (IPM)

$$\underset{x}{\text{minimize}}\ f(x) \quad \text{subject to } c(x) = 0 \quad \& \quad x \geq 0$$

Related to classical barrier methods [Fiacco & McCormick]

$$\begin{cases} \underset{x}{\text{minimize}} & f(x) - \mu \sum \log(x_i) \\ \text{subject to} & c(x) = 0 \end{cases}$$

$\mu = 0.1$ $\mu = 0.001$



$$\text{minimize}\ x_1^2 + x_2^2 \quad \text{subject to} \quad x_1 + x_2^2 \geq 1$$

Global Convergence of SQP/IPM

SQP & IPM converge quadratically "near" solution
... but diverge far from solution

convergence from remote starting points:

1. penalty function: $\pi > 0$ penalty parameter

$$\underset{x}{\text{minimize}} \ \Phi(x, \pi) = f(x) + \pi \|c(x)\|$$

   - equivalence of optimality $\Rightarrow$ exact for $\pi > \|y^*\|_D$
   - nonsmooth $\Rightarrow$ $S\ell_1QP$ method

2. enforce descent in penalty function by ...
   - 2.1 line-search ... backtrack on SQP/IPM step:
     $x_k + \alpha \Delta x \qquad \text{for} \ \ \alpha = 1, \frac{1}{2}, \ldots$
   - 2.2 restrict step with trust-region: $\|\Delta x\| \le \rho_k$

Filter Methods for NLP

Penalty function can be inefficient

- Penalty parameter not known a priori: $\pi > \|y^*\|_D$
- Large penalty parameter $\Rightarrow$ slow convergence

Two competing aims in optimization:

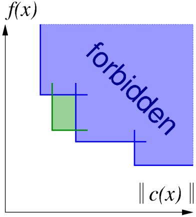
1. Minimize $f(x)$
2. Minimize $h(x) := \|c(x)\|$ ... more important

$\Rightarrow$ concept from multi-objective optimization:
(h_{k+1}, f_{k+1}) dominates (h_l, f_l) iff $h_{k+1} \leq h_l$ & $f_{k+1} \leq f_l$

Filter Methods for NLP

Filter $\mathcal{F}$: list of non-dominated pairs (h_l, f_l)

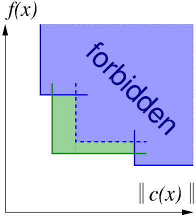
- new x_{k+1} acceptable to filter $\mathcal{F}$, iff
  1. $h_{k+1} \leq h_l \; \forall l \in \mathcal{F}$, or
  2. $f_{k+1} \leq f_l \; \forall l \in \mathcal{F}$



Filter Methods for NLP

Filter $\mathcal{F}$: list of non-dominated pairs (h_l, f_l)

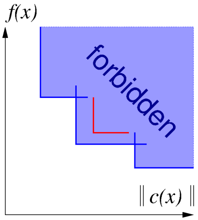
- new x_{k+1} acceptable to filter $\mathcal{F}$, iff
  1. $h_{k+1} \leq h_l \; \forall l \in \mathcal{F}$, or
  2. $f_{k+1} \leq f_l \; \forall l \in \mathcal{F}$
- remove redundant entries



Filter Methods for NLP

Filter $\mathcal{F}$: list of non-dominated pairs (h_l, f_l)

- new x_{k+1} acceptable to filter $\mathcal{F}$, iff
  1. $h_{k+1} \leq h_l \; \forall l \in \mathcal{F}$, or
  2. $f_{k+1} \leq f_l \; \forall l \in \mathcal{F}$
- remove redundant entries
- reject new x_{k+1},
  if $h_{k+1} > h_l$ & $f_{k+1} > f_l$
  ... reduce trust region radius $\Delta = \Delta/2$



$\Rightarrow$ often accept new x_{k+1}, even if penalty function increases

Nonlinear Optimization Software

- Augmented Lagrangian
  - `LANCELOT`: bound constrained; trust-region
  - `MINOS`: linearly constrained; line-search
  - `PENNON`: line-search or trust-region
- Sequential quadratic programming
  - `FILTER`: trust-region; no penalty function
  - `KNITRO`: trust-region; SLP-EQP ... `options "alg=3";`
  - `SNOPT`: line-search; ℓ_1 exact penalty function
- Interior point methods
  - `KNITRO`: trust-region; SQP on barrier problem
  - `LOQO`: line-search; diagonal perturbation
  - `IPOPT`: line-search; no penalty function

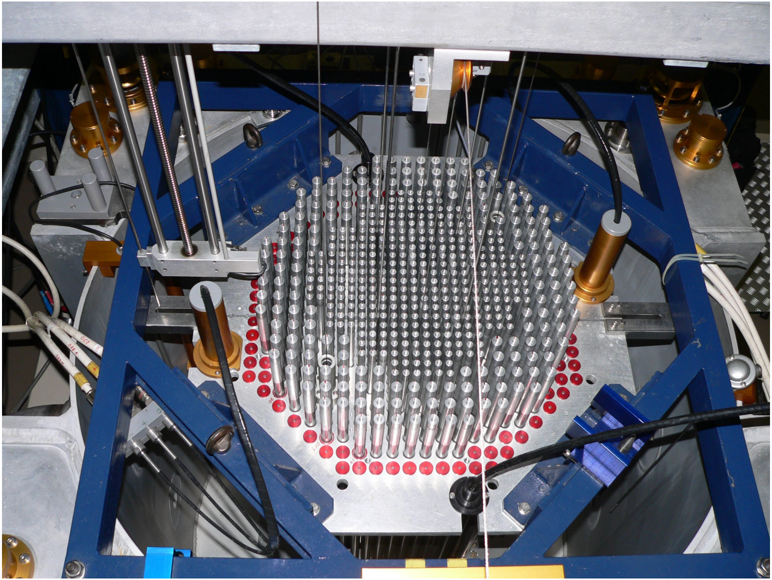
MODELING LANGUAGES

Optimization Modeling Languages

AMPL & GAMS

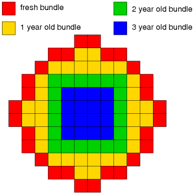
- high level languages for nonlinear optimization
- interpret problem description
- interface to solvers & returns results
- details of solver, derivatives & pre-solve are hidden from user
- modeling language (`var, minimize, subject to, ...`)
- programming language (`while, if, ...`)

AMPL Example: Core Reload Operation



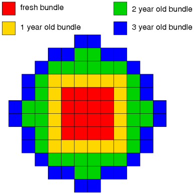
AMPL Example: Core Reload Operation

- simplified physics (neutron transport)
- maximize reactor efficiency after reload
- subject to diffusion process & safety
  $\Rightarrow$ integer & nonlinear model
- avoid reactor becoming sub-critical



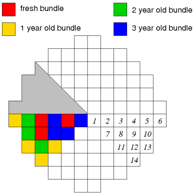
AMPL Example: Core Reload Operation

- simplified physics (neutron transport)
- maximize reactor efficiency after reload
- subject to diffusion process & safety
  $\Rightarrow$ integer & nonlinear model
- avoid reactor becoming sub-critical
- avoid reactor becoming super-heated



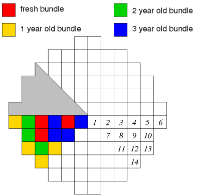
AMPL Example: Core Reload Operation

- simplified physics (neutron transport)
- maximize reactor efficiency after reload
- subject to diffusion process & safety
  $\Rightarrow$ integer & nonlinear model
- avoid reactor becoming sub-critical
- avoid reactor becoming super-heated
- look for cycles for moving bundles:
  e.g. $4 \rightarrow 6 \rightarrow 8 \rightarrow 10$
  means bundle moved from 4 to 6 to ...



AMPL Example: Core Reload Operation

- model with integer variables
  $x_{ilm} \in \{0, 1\}$
  $= 1$: node i has bundle l of cycle m
- exactly one bundle per node:
  $$\sum_{l,m} x_{ilm} = 1 \qquad \forall i \in I$$



AMPL:

- `var x {I,L,M} binary ;`
- `B1{i in I}: sum{l in L, m in M} x[i,l,m] = 1;`

www.mcs.anl.gov/~leyffer/MacMINLP/problems/c-reload.mod

Online Optimization Tools

NEOS Solvers

- solve optimization problems over internet
- connect applications & state-of-the-art solvers
- AMPL/GAMS ... input formats
- solvers for nonlinear optimization, integer, & many more!
- new NEOS-API (write software that submits jobs)
- ... and it's free!!! `www-neos.mcs.anl.gov/neos/`
- winner of the 2003 Beale-Orchard Hays Prize

OTHER OPTIMIZATION FLAVOURS

Integer Nonlinear Optimization

Nonlinear optimization with integer variables

$$\left\{ \begin{array}{lll} \underset{x}{\text{minimize}} & f(x, y) & \text{objective} \\ \text{subject to} & c(x, y) = 0 & \text{PDE constraints} \\ & x \geq 0, \;\; y \in Y & \text{variables} \end{array} \right.$$

where $y \in Y$ integer, e.g. $\{0, 1\}$, or $\{0, 1, 2, \ldots\}$ ...
Applications:

- Chemical Engineering: process synthesis, batch plant design, cyclic scheduling, design of distillation columns
- Topology optimization
- Blackout prevention in electrical power systems
- Design of nanoscale materials; accelerators ...

A Popular Integer Optimization Method

Dantzig's Two-Phase Method for MINLP Adapted by Leyffer and Linderoth

1. Convince the user that he or she does not wish to solve a mixed integer nonlinear programming problem at all!

A Popular Integer Optimization Method

Dantzig's Two-Phase Method for MINLP Adapted by Leyffer and Linderoth

1. Convince the user that he or she does not wish to solve a mixed integer nonlinear programming problem at all!
2. Otherwise, solve the continuous relaxation (NLP) and round off the minimizer to the nearest integer.

A Popular Integer Optimization Method

Dantzig's Two-Phase Method for MINLP Adapted by Leyffer and Linderoth

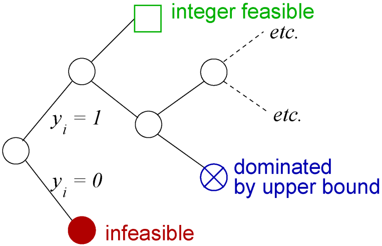
1. Convince the user that he or she does not wish to solve a mixed integer nonlinear programming problem at all!
2. Otherwise, solve the continuous relaxation (NLP) and round off the minimizer to the nearest integer.

- Sometimes a continuous approximation to the discrete (integer) decision is accurate enough in practice.
  - Yearly tree harvest in Washington
- For $0 - 1$ problems, or those in which the $|y|$ is "small", the continuous approximation to the discrete decision is not accurate enough for practical purposes.
- Conclusion: MINLP methods must be studied!

Branch-and-Bound

Solve relaxed NLP ($0 \leq y \leq 1$ continuous relaxation)
...solution value provides lower bound

- Branch on y_i non-integral
- Solve NLPs & branch until
  1. Node infeasible ... ●
  2. Node integer feasible ... □
     $\Rightarrow$ get upper bound (U)
  3. Lower bound $\geq U$ ... $\otimes$



Search until no unexplored nodes on tree

Nash & Stackelberg Games

Nash Game: e.g. electricity producers

Nash Game: non-cooperative equilibrium of several players

$$y_i^* \in \begin{cases} \underset{y_i}{\text{argmin}} & b_i(\hat{y}) \\ \text{subject to} & c_i(y_i) \geq 0 \end{cases} \qquad \text{player } i$$

- $\hat{y} = (y_1^*, \dots, y_{i-1}^*, y_i, y_{i+1}^*, \dots, y_l^*)$
- All players are equal

Nash & Stackelberg Games

Optimality conditions of Nash players ...

$$\begin{aligned} \nabla b(y) - \nabla c(y) z = 0 \\ 0 \leq z \perp c(y) \geq 0 \end{aligned}$$

where

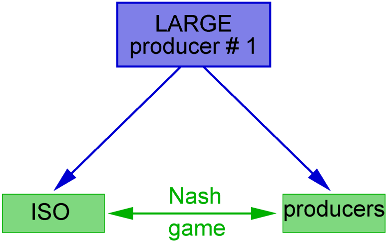
- $b(y) = (b_1(y), \ldots, b_k(y))$ & $c(y) = (c_1(y), \ldots, c_k(y))$
- $\perp$ means $z^T c(y) = 0$, either $z_i > 0$ or $c_i(y) > 0$

- Nonlinear complementarity problem (NCP)
- Robust large scale solvers exist: `PATH` [Munson]

Nash & Stackelberg Games

Single dominant player (leader) & Nash followers

$$\left\{ \begin{array}{ll} \min & f(x,y,z) \\ \text{s.t.} & h(x,y,z) = 0 \\ & 0 \le z \;\perp\; c(x,y) \ge 0 \end{array} \right.$$



Nash game parameterized in leader's variables x

Mathematical Program with Equilibrium Constraints (MPEC)
... ill-conditioned but [Anitescu, Munson & L] can do it!

Conclusions

Optimization is ubiquitous in science & engineering

- applications in engineering, operations research, games
- diverse modeling paradigms & tools exist
- large-scale nonlinear optimization tools

Open question: DOE optimization differ

- optimization over simulations (PDE, multiphysics)
- modeling languages not applicable?
  ... how to transfer optimization expertise
  ... how to transfer models/problems

THANKS FOR YOUR ATTENTION!