Optimization: beyond the normal

Michael C. Ferris

Joint work with: Michael Bussieck, Jan Jagla, Lutz Westermann and Roger Wets

Supported partly by AFOSR, DOE and NSF

University of Wisconsin, Madison

Lunchtime seminar
February 24, 2012

The problem

A furniture maker can manufacture and sell four different dressers. Each dresser requires a certain number t of man-hours for carpentry, and a certain number t_{fj} of man-hours for finishing, $j = 1, \ldots, 4$. In each period, there are d_c man-hours available for carpentry, and d_f available for finishing. There is a (unit) profit $\bar{c}_j$ per dresser of type j that's manufactured. The owner's goal is to maximize total profit:

$$\max_{x \geq 0} \; 12x_1 + 25x_2 + 21x_3 + 40x_4 \qquad (\textit{profit})$$

subject to

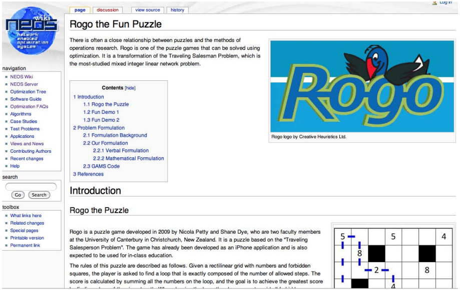
$$4x_1 + 9x_2 + 7x_3 + 10x_4 \leq 6000 \qquad (\textit{carpentry})$$
$$x_1 + x_2 + 3x_3 + 40x_4 \leq 4000 \qquad (\textit{finishing})$$

Succinctly:

$$\max_x \; c^T x \text{ s.t. } Tx \leq d, x \geq 0$$

Show me on a problem like mine

- Solution is $(4000/3, 0, 0, 200/3)$, value $\$18,667$
- Repeated solutions of multiple (different) problems enables "understanding" of the solution space (or sensitivity)
- NEOS wiki (www.neos-guide.org) or try out NEOS solvers (www.neos-solvers.org) for extensive examples



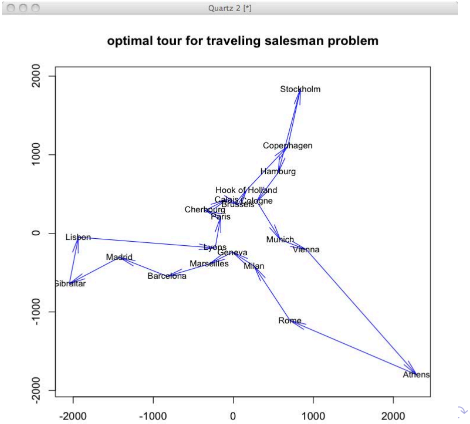
- Sudoku, etc

Make it work in/enhance my environment

- In practice: need (large scale) data, problem/model transformations, access to solution features
- Modeling systems (AIMMS, AMPL, ... , GAMS, ...) provide some of these needs from an optimization perspective
- Open source, libraries, interfaces to Excel/Matlab/R

Traveling salesman in R:

- ...
- wgdx (fnData, data)
- gams("tspDSE.gms")
- stat ← list(name='modelstat')
- v ← rgdx(fnSol, stat)
- ...
- R commands for graphics output



Is your time estimate that good?

- The time for carpentry and finishing for each dresser cannot be known with certainty
- Each entry in T takes on four possible values with probability $1/4$, independently
- 8 entries of T are random variables: $s = 65,536$ different T's each with same probability of occurring
- But decide "now" how many dressers x of each type to build
- Might have to pay for overtime (for carpentry and finishing)
- Can make different overtime decision y^s for each scenario s - recourse!

Extended Form Problem

$$\min_{x,y} -c^T x + \sum_{s=1}^{65,536} \pi_s q^T y$$

subject to

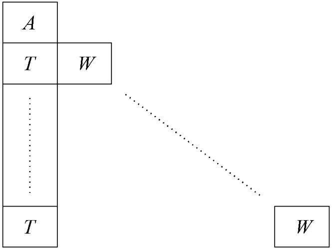
$$T^s x - y^s \leq d, \quad s = 1, \ldots, 65,536$$
$$x, y^s \geq 0$$

- Immediate costs + expected *future costs*
- Stochastic program with recourse

Stochastic recourse

- Two stage stochastic programming, x is here-and-now decision, recourse decisions y depend on realization of a random variable
- $\mathbb{R}$ is a risk measure (e.g. expectation, CVaR)

$$\begin{aligned} \text{SP:} \min \quad & c^\top x + \mathbb{R}[q^\top y] \\ \text{s.t.} \quad & Ax = b, \quad x \geq 0, \\ \forall \omega \in \Omega: \quad & T(\omega)x + W(\omega)y(\omega) \leq d(\omega), \\ & y(\omega) \geq 0. \end{aligned}$$



Computation methods matter!

- Problem becomes large very quickly!
- Lindo solver defaults: 825 seconds
- Lindo solver barrier method: 382 seconds
- CPLEX solver barrier method: 4 seconds (8 threads)

We do this! How to formulate model, how to solve, why it works

How to generate the model

1. May have multiple sources of uncertainty: e.g. man-hours d also can take on 4 values in each setting independently: $s = 1,048,576$
2. emp.info: model transformation

```
randvar T('c','1') discrete .25  3.60 .25  3.90 .25  4.10 .25  4.40
randvar T('c','2') discrete .25  8.25 .25  8.75 .25  9.25 .25  9.75
randvar T('c','3') discrete .25  6.85 .25  6.95 .25  7.05 .25  7.15
randvar T('c','4') discrete .25  9.25 .25  9.75 .25 10.25 .25 10.75
randvar T('f','1') discrete .25  0.85 .25  0.95 .25  1.05 .25  1.15
randvar T('f','2') discrete .25  0.85 .25  0.95 .25  1.05 .25  1.15
randvar T('f','3') discrete .25  2.60 .25  2.90 .25  3.10 .25  3.40
randvar T('f','4') discrete .25 37.00 .25 39.00 .25 41.00 .25 43.00
randvar d('c')     discrete .25 5873. .25 5967. .25 6033. .25 6127.
randvar d('f')     discrete .25 3936. .25 3984. .25 4016. .25 4064.

stage 2 y t d cost cons obj
```

3. Generates extensive form problem with over 3 million rows and columns and 29 million nonzeros
4. Solves on 24 threaded cluster machine in 262 secs

Sampling methods

But what if the number of scenarios is too big (or the probability distribution is not discrete)? use sample average approximation (SAA)

- Take sample $\xi_1, \ldots, \xi_N$ of N realizations of random vector ξ
  - viewed as historical data of N observations of ξ, or
  - generated via Monte Carlo sampling
- for any $x \in X$ estimate $f(x)$ by averaging values $F(x, \xi_j)$

$$\text{(SAA):} \ \min_{x \in X} \left\{ \hat{f}_N(x) := \frac{1}{N} \sum_{j=1}^{N} F(x, \xi_j) \right\}$$

- Nice theoretical asymptotic properties
- Can use standard optimization tools to solve the SAA problem
- EMP = SLP $\implies$ SAA $\implies$ (large scale) LP

Convergence

N	Time(s)	Soln	Profit
1000	0.6	(265,0,662,34)	18050
2000	1.0	(254,0,668,34)	18057
3000	1.6	(254,0,668,34)	18057
4000	2.3	(255,0,662,34)	18058
5000	3.1	(257,0,666,34)	18054
6000	3.9	(262,0,663,34)	18051
7000	5.0	(257,0,666,34)	18054
8000	6.1	(262,0,663,34)	18048
9000	7.3	(257,0,666,34)	18051
1m	262.0	(257,0,666,34)	18051

SAA can work well, but this is a 4 variable problem and distributions are discrete

What do we learn?

- Deterministic solution: $x_d = (1333, 0, 0, 67)$
- Expected profit using this solution: \$16,942
- Expected (averaged) overtime costs: \$1,725
- Extensive form solution: $x_s = (257, 0, 666, 34)$ with expected profit \$18,051
- Deterministic solution is not optimal for stochastic program, but more significantly it isn't getting us on the right track!
- Stochastic solution suggests large number of "type 3" dressers, while deterministic solution has none!

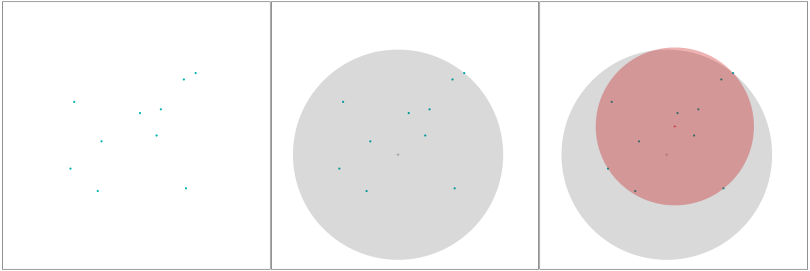
Continuous distributions: Newsboy problems

N	Derand		SAA	
	Mean	Stdev	Mean	Stdev
2	16.85	2.185	16.94	3.615
5	14.84	1.369	14.92	2.791
10	14.23	1.127	14.57	2.248
20	14.03	0.797	14.18	1.635
100	14.01	0.100	14.48	0.745

1. As the sample size N increases, the optimal solutions obtained by both methods converge to the true solution, i.e. 14
2. For a given sample size N, new sampling method (derand) is always (slightly) closer to the true solution
3. But standard deviation of the optimal solutions obtained by derand is significantly smaller than the SAA method

Circle cover problem

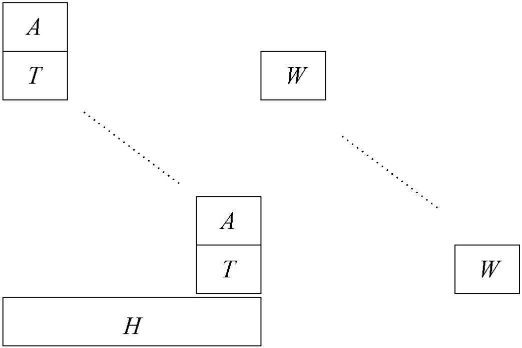
Given a set of points find the location (x, y) of the center of the circle with minimum radius that covers all points (coverage problem)



- This is an example of a nonlinear program (second order cone program).
- What if the points are only known by distribution?
- Notion of robust optimization (all points in enclosing circle) or a stochastic programming formulation or chance constraints

Key-idea: Non-anticipativity constraints

- Replace x with $x_1, x_2, \ldots, x_K$
- Non-anticipativity: $(x_1, x_2, \ldots, x_K) \in L$ (a subspace) - the H constraints



Computational methods exploit the separability of these constraints, essentially by dualization of the non-anticipativity constraints.

- Primal and dual decompositions (Lagrangian relaxation, progressive hedging , etc)
- L shaped method (Benders decomposition applied to det. equiv.)
- Trust region methods and/or regularized decomposition

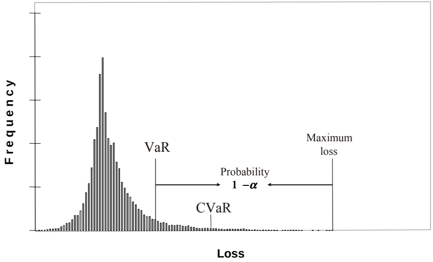
Risk Measures

$\overline{CVaR}_\alpha$: mean of upper tail beyond α-quantile (e.g. $\alpha = 0.95$)

- Classical: utility/disutility $u(\cdot)$:

$$\min_{x \in X} f(x) = \mathbb{E}[u(F(x, \xi))]$$

- Modern approach to modeling risk aversion uses concept of risk measures



- mean-risk, semi-deviations, mean deviations from quantiles, VaR, CVaR
- Römisch, Schultz, Rockafellar, Urasyev (in Math Prog literature)
- Much more in mathematical economics and finance literature
- Optimization approaches still valid, different objectives

Example: Portfolio Model (core model)

- Determine portfolio weights w_j for each of a collection of assets
- Asset returns v are random, but jointly distributed
- Portfolio return $r(w, v)$
- Minimize a "risk" measure

$$\begin{array}{ll} \max & 0.2 * \mathbb{E}(r) + 0.8 * \underline{CVaR}_\alpha(r) \\ \text{s.t.} & r = \sum_j v_j * w_j \\ & \sum_j w_j = 1,\ w \geq 0 \end{array}$$

- Jointly distributed random variables v, realized at stage 2
- Variables: portfolio weights w in stage 1, returns r in stage 2
- Coherent risk measures $\mathbb{E}$ and $\underline{CVaR}$

Additional techniques requiring extensive computation

- Continuous distributions, sampling functions, density estimation
- Chance constraints: $Prob(T_i x + W_i y_i \geq h_i) \geq 1 - \alpha$ - can reformulate as MIP and adapt cuts (Luedtke) empinfo: chance E1 E2 0.95
- Use of discrete variables (in submodels) to capture logical or discrete choices (logmip - Grossmann et al)
- Robust or stochastic programming
- Decomposition approaches to exploit underlying structure identified by EMP
- Nonsmooth penalties and reformulation approaches to recast problems for existing or new solution methods (ENLP)
- Conic or semidefinite programs - alternative reformulations that capture features in a manner amenable to global computation

Conclusions

- Optimization helps understand what drives a system
- Uncertainty is present everywhere (the world is not "normal")
- We need not only to quantify it, but we need to hedge/control/ameliorate it
- Modeling, optimization, and computation embedded within the application domain is critical