Stochastic Programming in GAMS

Michael C. Ferris

Joint work with: Michael Bussieck, Martha Loewe and Lutz Westermann

University of Wisconsin, Madison

International Conference on Continuous Optimization,
Lisbon, Portugal
August 1, 2013
The problem

A furniture maker can manufacture and sell four different dressers. Each dresser requires a certain number $t_{cj}$ 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:

$$\max_{x \geq 0} \quad 12x_1 + 25x_2 + 21x_3 + 40x_4$$

(profit)

subject to

$$4x_1 + 9x_2 + 7x_3 + 10x_4 \leq 6000$$

(carpentry)

$$x_1 + x_2 + 3x_3 + 40x_4 \leq 4000$$

(finishing)

Succinctly:

$$\max_x \quad c^T x \text{ s.t. } Tx \leq d, x \geq 0$$
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!
Stochastic recourse

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

\[ \text{SP: max } c^T x + R[q^T y] \]

\[ \text{s.t. } Ax = b, \quad x \geq 0, \]

\[ \forall \omega \in \Omega : \quad T(\omega)x + W(\omega)y(\omega) \leq d(\omega), \]

\[ y(\omega) \geq 0. \]

EMP/SP extensions to facilitate these models.
Extended Form Problem

\[
\max_{x,y} c^T x + \sum_{s=1}^{65,536} \pi_s q^T y
\]

subject to

\[
T^s x - y^s \leq d, \quad s = 1, \ldots, 65,536
\]

\[
x, y^s \geq 0
\]

- Immediate profit + expected future profit
- Stochastic program with recourse
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: $x_e = (257, 0, 666, 34)$ 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!
- How to formulate model, how to solve, why it works
Models with explicit random variables

- **Model transformation:**
  - Write a core model as if the random variables are constants
  - Identify the random variables and decision variables and their staging
  - Specify the distributions of the random variables

- **Solver configuration:**
  - Specify the manner of sampling from the distributions
  - Determine which algorithm (and parameter settings) to use

- **Output handling:**
  - Optionally, list the variables for which we want a scenario-by-scenario report

Ferris (Univ. Wisconsin)  EMP/SP  ICCOPT 2013  7 / 26
Stochastic Programming as an EMP

- Separate solution process from model description

Three separate pieces of information (extended mathematical program) needed

1. **emp.info: model transformation**
   
   randvar T('c','1') discrete .25 3.6 .25 3.9 ...
   
   ... stage 2 y T profit cons obj

2. **solver.opt: solver configuration** (benders, sampling strategy, etc)
   
   4 "ISTRAT" * solve universe problem (DECIS/Benders)

3. **dictionary: output handling** (where to put all the “scenario solutions”)
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)
- Models can be solved by the extensive form equivalent, existing codes such as LINDO and DECIS, or decomposition approaches such as Benders, ATR, etc - Just change the solver
Key-idea: Non-anticipativity constraints

- Replace $x$ with $x_1, x_2, \ldots, x_K$
- Non-anticipativity: $\left(x_1, x_2, \ldots, x_K\right) \in L$ (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
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/SP allows description of compositional (nonlinear) random effects in generating $\omega$

   $\text{i.e. } \omega = \omega_1 \times \omega_2$, \quad $T(\omega) = f(X(\omega_1), Y(\omega_2))$

3. emp.info: model transformation

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

4. Generates extensive form problem with over 3 million rows and columns and 29 million nonzeros

5. Solves on 24 threaded cluster machine in 262 secs
Multi-stage model: Clear Lake

- Easy to write down multi-stage problems
- Water levels $l(t)$ in dam for each month $t$
- Determine what to release normally $r(t)$, what then floods $f(t)$ and what to import $z(t)$
- Minimize cost of flooding and import
- Change in reservoir level in period $t$ is $\delta(t)$

$$\max cost = c(f, z)$$
$$\text{s.t. } l(t) = l(t-1) + \delta(t) + z(t) - r(t) - f(t)$$

- Random variables are $\delta$, realized at stage $t$, $t \geq 2$.
- Variables $l, r, f, z$ in stage $t$, $t \geq 2$.
- Balance constraint at $t$ in stage $t$. 
Multi to 2 stage reformulation

Stage 1  Stage 2  Stage 3

Cut at stage 2
Cut at stage 3

Ferris (Univ. Wisconsin) EMP/SP ICCOPT 2013
Multi to 2 stage reformulation

Stage 1  Stage 2  Stage 3

Cut at stage 2

Ferris (Univ. Wisconsin)
Multi to 2 stage reformulation

Stage 1  Stage 2  Stage 3

Cut at stage 3
Solution options

- Form the extensive form equivalent
- Solve using LINDO api (stochastic solver)
- Convert to two stage problem and solve using DECIS or any number of competing methods

Problem with $3^{40} \approx 1.2 \times 10^{19}$ realizations in stage 2

- DECIS using Benders and Importance Sampling: < 1 second (and provides confidence bounds)
- CPLEX on a presampled approximation:

<table>
<thead>
<tr>
<th>sample</th>
<th>samp. time(s)</th>
<th>CPLEX time(s) for solution</th>
<th>cols (mil)</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>0.0</td>
<td>5 (4.5 barrier, 0.5 xover)</td>
<td>0.25</td>
</tr>
<tr>
<td>1000</td>
<td>0.2</td>
<td>18 (16 barrier, 2 xover)</td>
<td>0.5</td>
</tr>
<tr>
<td>10000</td>
<td>28</td>
<td>195 (44 barrier, 151 xover)</td>
<td>5</td>
</tr>
<tr>
<td>20000</td>
<td>110</td>
<td>1063 (98 barrier, 965 xover)</td>
<td>10</td>
</tr>
</tbody>
</table>
Chance constraints

- Use of discrete variables (in submodels) to capture logical or discrete choices (logmip - Grossmann et al)
- Chance constraints: $\text{Prob}(G(x, \xi) \leq \gamma) \geq 1 - \alpha$
- emp.info: chance E1 E2 0.95
- Use binary variable to model indicator function

$$\mathbb{E}(\mathcal{I}_{\{G(x,\xi) \leq \gamma\}}) = P(G(x, \xi) \leq \gamma) \geq 1 - \alpha$$

- Single or joint probabilistic constraints
- Reformulate as MIP (bigM) and adapt cuts

$$\max_x c^T x$$

s.t. $A^\omega x \leq b^\omega + M^\omega (1 - y_\omega), \forall \omega$

$$\sum_{\omega \in \Omega} p_\omega y_\omega \geq 1 - \alpha, \quad x \geq 0, \quad y_\omega \in (0, 1)^{|\Omega|}$$

- Optional reformulations: convex hull or indicator constraints
- chance E1 E2 0.6 viol assigns the violation probability to the variable viol
Optimization of risk measures

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

- Value at Risk (VaR) can be viewed as a chance constraint (hard):
- CVaR gives rise to a convex optimization problem (easy)

- mean-risk, semi-deviations, mean deviations from quantiles, VaR, CVaR
Example: Portfolio Model

- Coherent risk measures $\mathbb{E}$ and $CVaR$ (or convex combination)
- Maximize combination of mean and mean of the lower tail (mean tail loss):

$$\max \quad 0.2 \cdot \mathbb{E}(r) + 0.8 \cdot CVaR_\alpha(r)$$

subject to:

$$r = \sum_j v_j \cdot w_j$$

$$\sum_j w_j = 1, \quad w \geq 0$$

- Jointly distributed random variables $v$, realized at stage 2
- Variables: portfolio weights $w$ in stage 1, returns $r$ in stage 2
Example: Portfolio Model

- Coherent risk measures $\mathbb{E}$ and CVaR (or convex combination)
- Maximize combination of mean and mean of the lower tail (mean tail loss):

  $$\max \ 0.2 \times \mathbb{E}(r) + 0.8 \times CVaR_\alpha(r)$$

  $$\text{s.t.} \quad r = \sum_j v_j \cdot w_j$$
  $$\sum_j w_j = 1, \ w \geq 0$$

- Jointly distributed random variables $v$, realized at stage 2
- Variables: portfolio weights $w$ in stage 1, returns $r$ in stage 2
- Easy to add integer constraints to model (e.g. cardinality constraints)
- Alternative: mean-variance model (Markowitz)

  $$\min \ w^T \Sigma w - q \sum_j v_j \cdot w_j$$

  $$\sum_j w_j = 1, \ w \geq 0$$
Other EMP information

- emp.info: model transformation
  - expected_value EV_r r
  - cvarlo CVaR_r r alpha
  - stage 2 v r defr
  - jrandvar v("att") v("gmc") v("usx") discrete
    table of probabilities and outcomes

- Variables are assigned to $\mathbb{E}(r)$ and $\text{CVaR}_\alpha(r)$; can be used in model (appropriately) for objective, constraints, or be bounded
Reformulation details

- $\mathbb{E}(r)$ is simply a sum (probability and values generated by EMP)
- VaR is a chance constraint:

$$\mathbb{E}(I_{G(x,\xi)\leq\gamma}) = P(G(x,\xi) \leq \gamma) \geq 1 - \alpha \iff \text{VaR}_{1-\alpha}(G(x,\xi)) \leq \gamma$$

- CVaR transformation: can be written as convex optimization using:

$$\text{CVaR}_\alpha(r) = \max_{a \in \mathbb{R}} \left\{ a - \frac{1}{\alpha} \mathbb{E}(a - r)_+ \right\}$$
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 $\xi$  
  - viewed as historical data of $N$ observations of $\xi$, or  
  - generated via Monte Carlo sampling

- for any $x \in X$ estimate $f(x)$ by averaging values $F(x, \xi_j)$

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

- Nice theoretical asymptotic properties
- Can use standard optimization tools to solve the SAA problem
- $\text{EMP} = \text{SLP} \implies \text{SAA} \implies (\text{large scale}) \text{ LP}$
- Continuous distributions, sampling functions, density estimation
SAA can work well, but this is a 4 variable problem and distributions are discrete.
Continuous distributions

- Can sample from continuous distributions (configurable on options)

```
randvar d normal 45 10
sample d 9
setSeed 101
```

The second line determines the size of the sample of the distribution of the random variable $D$ to be 9.

- Variance reduction: e.g. LINDO solver provides three methods for reducing the variance: Monte Carlo sampling, Latin Square sampling and Antithetic sampling

- Can use user supplied sampling libraries

- Sampling can be separated from solution (i.e. can generate discrete approximation and then solve using existing algorithms)
## Parametric distributions supported

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Parameter 1</th>
<th>Parameter 2</th>
<th>Parameter 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta</td>
<td>shape 1</td>
<td>shape 2</td>
<td></td>
</tr>
<tr>
<td>Cauchy</td>
<td>location</td>
<td>scale</td>
<td></td>
</tr>
<tr>
<td>Chi_Square</td>
<td>deg. of freedom</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exponential</td>
<td>lambda</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gamma</td>
<td>shape</td>
<td>scale</td>
<td></td>
</tr>
<tr>
<td>LogNormal</td>
<td>mean</td>
<td>std dev</td>
<td></td>
</tr>
<tr>
<td>Normal</td>
<td>mean</td>
<td>std dev</td>
<td></td>
</tr>
<tr>
<td>StudentT</td>
<td>deg. of freedom</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Triangular</td>
<td>low</td>
<td>mid</td>
<td>high</td>
</tr>
<tr>
<td>Uniform</td>
<td>low</td>
<td>high</td>
<td></td>
</tr>
<tr>
<td>Weibull</td>
<td>shape</td>
<td>scale</td>
<td></td>
</tr>
<tr>
<td>Binomial</td>
<td>n</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>Geometric</td>
<td>p</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Logarithmic</td>
<td>p-factor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poisson</td>
<td>lambda</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What is EMP?

Annotates existing equations/variables/models for modeler to provide/define additional structure

- equilibrium
- vi (agents can solve min/max/vi)
- bilevel (reformulate as MPEC, or as SOCP)
- disjunction (or other constraint logic primitives)
- randvar
- dualvar (use multipliers from one agent as variables for another)
- extended nonlinear programs (library of plq functions)

Currently available within GAMS
Extension to MOPEC: agents solve a Stochastic Program

Buy \( y_i \) contracts in period 1, to deliver \( D(\omega)y_i \) in period 2, scenario \( \omega \).

Each agent \( i \):

\[
\begin{align*}
\min & \quad C(x_i^1) + \sum_{\omega} \pi_\omega C(x_i^2(\omega)) \\
\text{s.t.} & \quad p^1 x_i^1 + vy_i \leq p^1 e_i^1 \quad \text{(budget time 1)} \\
& \quad p^2(\omega) x_i^2(\omega) \leq p^2(\omega)(D(\omega)y_i + e_i^2(\omega)) \quad \text{(budget time 2)}
\end{align*}
\]

\[
0 \leq v \perp - \sum_i y_i \geq 0 \quad \text{(contract)}
\]

\[
0 \leq p^1 \perp \sum_i (e_i^1 - x_i^1) \geq 0 \quad \text{(walras 1)}
\]

\[
0 \leq p^2(\omega) \perp \sum_i (D(\omega)y_i + e_i^2(\omega) - x_i^2(\omega)) \geq 0 \quad \text{(walras 2)}
\]
Conclusions

- **Uncertainty is present everywhere.** We need not only to quantify it, but we need to hedge/control/ameliorate it.
- EMP model type available within GAMS, is clear and extensible, additional structure available to solver (cf COR, STOC files, etc).
- Provides SP technology to application domains (cf AMPL and Pyomo extensions).
- Add links to specialized solvers beyond current group (e.g. stochastic integer programming).
- More general stochastic structure needed at modeling level.
- Specialized methodology for probabilistic contraints.
- Create environment where advanced SP can be used by modelers.