Stochastic Optimization: Solvers and Tools

Michael R. Bussieck
MBussieck@gams.com

GAMS Software GmbH
http://www.gams.de

Stochastic Optimization in the Energy Industry
Aachen, Germany, April 19-20, 2007

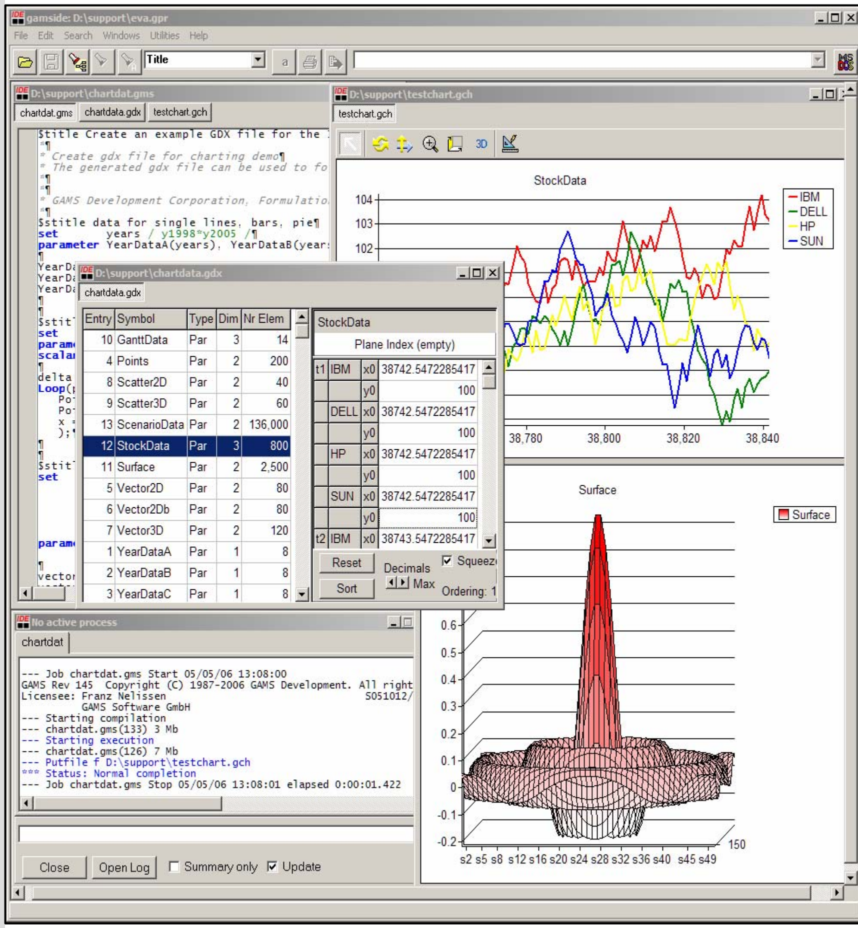
GAMS Development / GAMS Software

- Roots: **Research project** World Bank 1976
- Pioneer in **Algebraic Modeling Systems** used for economic modeling
- Went **commercial** in 1987
- **Offices** in Washington, D.C and Cologne

- Professional **software tool provider**
- Operating in a **segmented niche market**
- Broad **academic** & **commercial** user base and network

GAMS

GAMS at a Glance

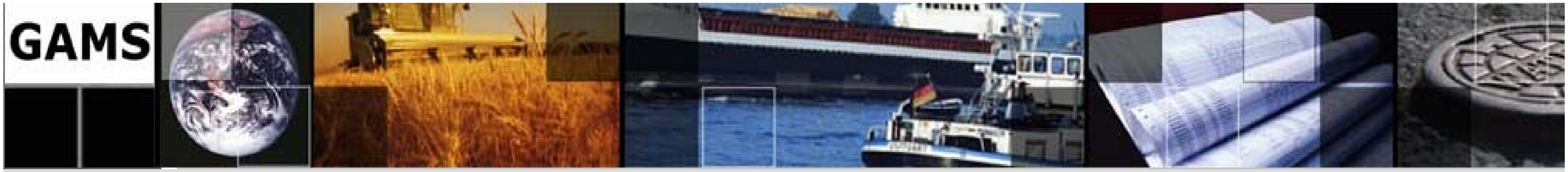


General **A**lgebraic **M**odeling **S**ystem:

Algebraic Modeling Language, Integrated Solver, Model Libraries, Connectivity- & Productivity Tools

Design Principles:

- Balanced mix of declarative and procedural elements
- Open architecture and interfaces to other systems
- Different layers with separation of:
  - model and data
  - model and solution methods
  - model and operating system
  - model and interface



AML and Stochastic Programming (SP)

- Algebraic Modeling Languages/Systems good way to represent optimization problems
  - Algebra is a universal language
  - Hassle free use of optimization solvers
  - Simple connection to data sources (DB, Spreadsheets, …) and analytic engines (GIS, Charting, …)
- Large number of (deterministic) models in production
  - Opportunity for *seamless* introduction of new technology like Global Optimization, Stochastic Programming, …
  - AML potential framework for SP

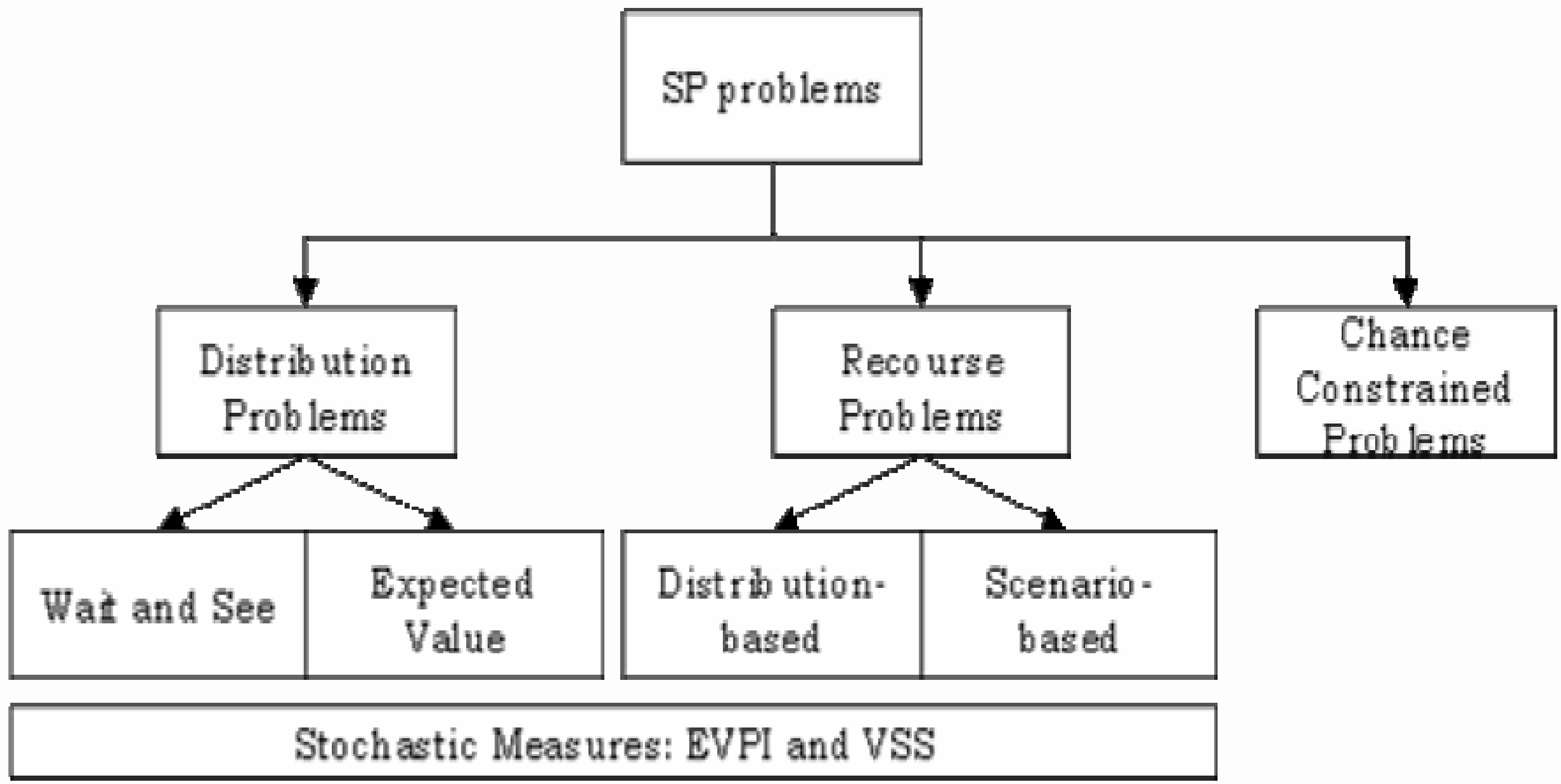
GAMS

Stochastic Programming Claims and ‘*Facts*'

- Lots of application areas (Finance, Energy, Telecommunication)
- Mature field (Dantzig ’55)
- Variety of SP problem classes with specialized solution algorithms (e.g. Bender’s Decomposition)

- Compared to deterministic mathematical programming (MP) small fraction
  - Only 0.2% of NEOS submission to SP solvers
- No/few commercially supported solvers for SP
- Various frustrations with industrial SP projects

Some Stochastic Programming Classes



Source: G. Mitra

GAMS

Chance Constraints

$$Z_{CCP} = \min cx$$

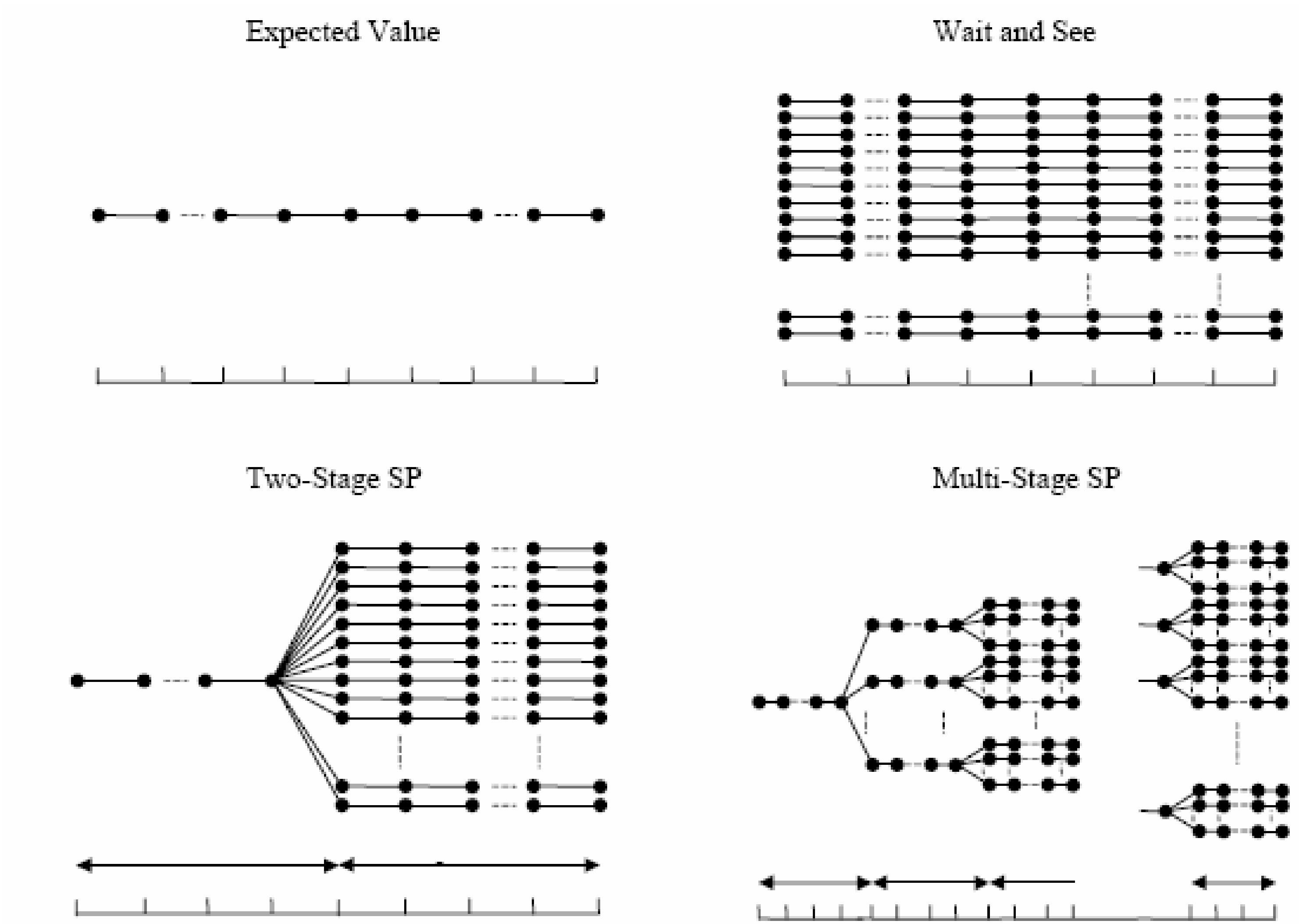
$$s.t.$$

$$A_0 x = b_0$$

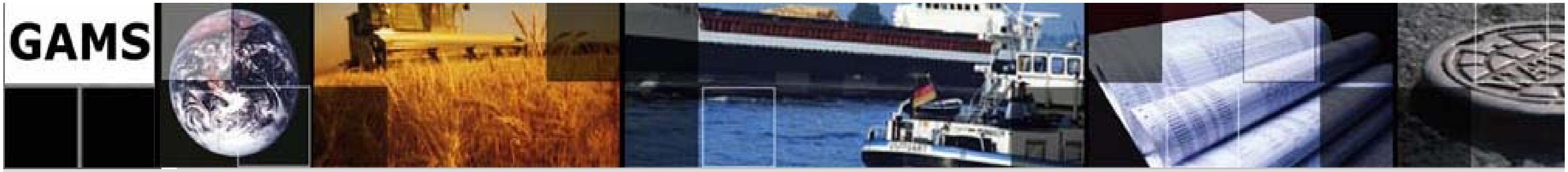
$$P\left\{A_i x \geq h_i\right\} \geq \beta_i \qquad i = 1..I, \qquad \beta_i \in [0,1]$$

- β is the reliability level

Dynamic Representation



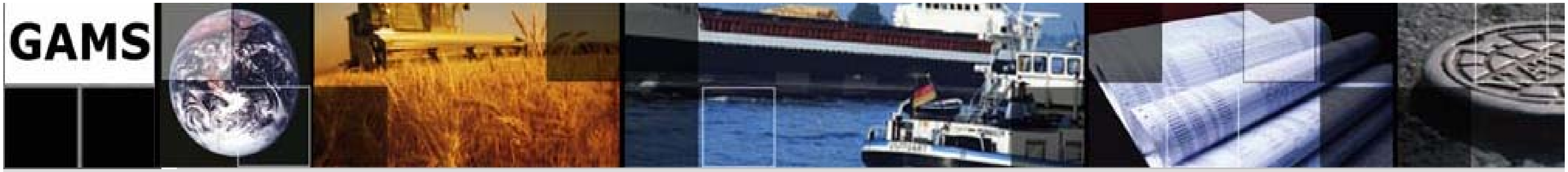
Source: G. Mitra



Two-Stage SLP with recourse

$$
\begin{array}{rrcrcrl}
\min\ z & = & cx & + & E\ f^{\omega} y^{\omega} & & \\
s/t & & Ax & & & = & b \\
 & & -B^{\omega} x & + & D^{\omega} y^{\omega} & = & d^{\omega} \\
 & & x, & & y^{\omega} & \geq & 0, \quad \omega \in \Omega.
\end{array}
$$

- x first-stage decision variables
- y^w second-stage decision variables
- c, f^w objective coefficients
- A,b first-stage coefficients
- B^w,D^w,d^w second-stage random parameters



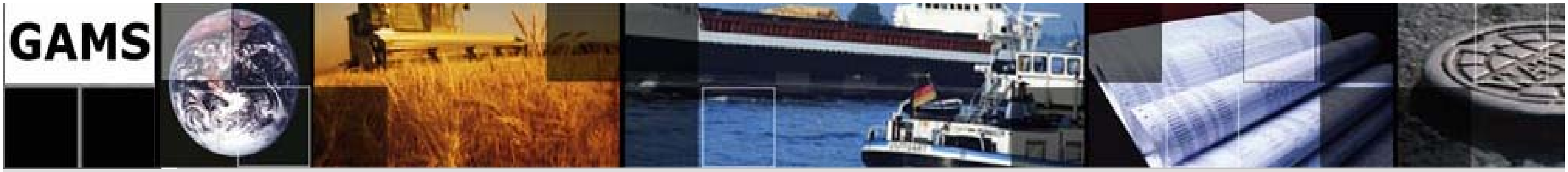
Multi-Stage SLP with recourse

$$Z_{HN} = \min_{x_1} \left\{ c_1 x_1 + E_{\xi_2}\left[\min_{x_2} c_2 x_2 + E_{\xi_3|\xi_2}\left[\min_{x_3} c_3 x_3 + \ldots + E_{\xi_T|\xi_{T-1}|\ldots|\xi_2} \min_{x_T} c_T x_T \right]\right]\right\}$$

subject to:

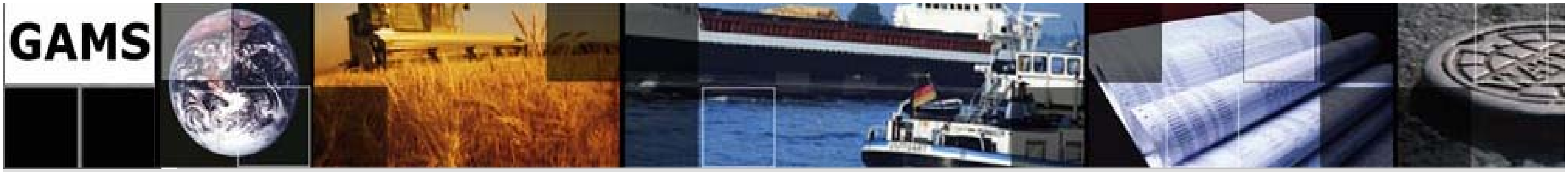
$$\begin{array}{lllll} A_{11}x_1 & & & & = b_1 \\ A_{21}x_1 + & A_{22}x_2 & & & = b_2 \\ A_{31}x_1 + & A_{32}x_2 + & A_{33}x_3 & & = b_3 \\ \vdots & & & \ddots & \vdots \\ A_{T1}x_1 + & A_{T2}x_2 + & A_{T3}x_3 + & \ldots & + A_{TT}x_T = b_T \end{array}$$

$$\ell_t \le x_t \le u_t;$$



Discrete distribution (Two-stage SLP)

$$
\begin{array}{rcccccccccc}
\min\ z & = & & & & & & & & & \\
s/t & & & & & & & & & & \\
cx & + & p^1 f y^1 & + & p^2 f y^2 & + & \cdots & + & p^W f y^W & & \\
Ax & & & & & & & & & = & b \\
-B^1 x & + & D y^1 & & & & & & & = & d^1 \\
-B^2 x & & & + & D y^2 & & & & & = & d^2 \\
\vdots & & & & & & \ddots & & & & \vdots \\
-B^W x & & & & & & & + & D y^W & = & d^W \\
x, & & y^1, & & y^2, & & \ldots, & & y^W & \geq & 0,
\end{array}
$$



Deterministic Equivalent (DE)

- Discrete Distribution/Scenarios
- Standard (Mixed Integer Nonlinear) Linear Program with block structure (opportunity for special purpose algorithms)
- Non-anticipativity (or locking) constraints
  - Explicit by new constraints (Split-variable representation maintains block structure)
  - Implicit through variable substitution
- Size equivalent to number of possible realizations (i.e. exponential in the number of random variables)
- Size of LP can quickly explode

Personalized Software Overview

- SP Solvers
  - Standard LP Solvers
  - SP solvers based on decomposition
- Modeling Tools
  - SMPS (core, time, stochastic)
  - Extension to AML
  - APIs
- Others
  - Scenario (tree) management
  - Visualization of results

Stochastic Programming Solvers

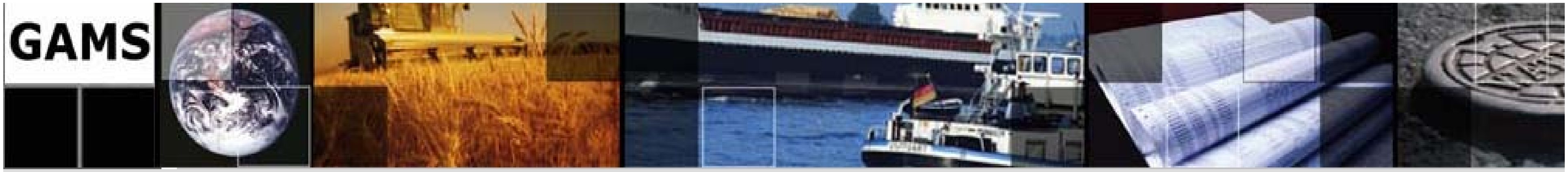
- LP solver
  - Interior point methods seem to be better than simplex
- All other ready to use solver (e.g. NEOS) for *n*-stage SP with recourse:
  - DECIS, Infanger (2-stage)
  - FortSP, OptiRisk
  - OSL/SE, IBM (discontinued)
  - Academic codes: MSLiP (Gassmann), bnbs (Altenstedt), ddsip (Schultz et. al) (2-stage MIP), …
- SLP-IOR (Mayer/Kall) started in 1992
  - Model management system for SLP
  - Rich solver library (21) for 18 SP model types

DECIS

- Solves two-stage stochastic linear programs with recourse
- Two-stage decomposition (Benders)
- Stores only one instance of the problem and generates scenario sub-problems as needed
- LP engine for sub-problems CPLEX or MINOS
- Solution Strategies
– Universe problem (all scenarios)
– Sampling: Crude Monte Carlo/Importance sampling

SP Modeling Tools

- Stochastic MPS
  - make it easy to convert existing deterministic LP into SLP
  - Add information about dynamic and stochastic structure.
- Core file (deterministic problem)
- Time file (map core file dynamic structure into stages)
- Stoch file (information about the random variables)
- SMPS format is extremely flexible exceeding the capability of any existing stochastic programming solver
- Difficult for a human to manage



SP Modeling Tools cont.

- Algebraic Modeling Languages
  - Similar to SMPS:
    - Supply time/stage map
    - Supply stochastic information
  - Integrated in language: AIMMS, GAMS, Mosel
  - Extensions: SAMPL, SMPL, StAMPL
- API
  - OSL/SE (discontinued)
  - COIN-OR Stochastic Modeling Interface (SMI) https://projects.coin-or.org/Smi
- Gear towards generating and solving DE

MPL

```
INDEX       I  =   1..7;

DECISION
        h[I,T] ;
        b[I,T];
        d[I,T];
        c[T] ;

DATA
        C0 = 50000;                    ! Initial amount of cash
        U = 10;                        ! Coupon
        G   = 0.005;                   ! Transaction cost rate
        V   = 0.15;                    ! Diversification limit

MODEL
       obj = max SUM(T=4: c[T]);

SUBJECT TO

BAL[T,I] WHERE (T<4) AND (I<7):
                    H[I,T] = H[I+1,T-1] + B[I,T] - S[I,T] ;

BAL[T,I] WHERE (T<4) AND (I=7) :
                    H[I,T] = B[I,T] ;

BAL[T,I] WHERE (T=4) AND (I<7) :
                    H[I,T] = H[I+1,T-1] ;

....
```

AMPL

```
set I:= 1 .. 7;

var   h{i in I, t in T}  >=0;
var   b{i in I, t in T}  >=0;
var   d{i in I, t in T}  >=0;
var   c{t in T}          >=0;

param C0:= 50000;              # Initial amount of cash
param U = 10;                  # Coupon
param G   = 0.005;             # Transaction cost rate
param V   = 0.15;              # Diversification limit

maximize obj: sum {t inT: t=4} c[t];

subject to bal{t in T: t< 4,i in I: i <7}:
                    h[i,t] =h[i+1, t-1]+ b[i,t] - d[i,t];

subject to bal {t in T: t< 4,i in I: i = 7}:
                    h[i,t] = b[i,t];

subject to bal{t in T: t=4,i in I: i<7}:
                    h[i,t] = h[i+1,t-1];

....
```

SMPL

```
TIME
     T : =   1..4;

SCENARIO
     Scen:= 1..8;

PROBABILITIES
     Prob[Scen]:= FILE("ScenProb.pro");

RANDOM DATA
     PR[T,I,Scen]    := DATABASE("tbl_Prices","Price");

STAGES:
   AGGREGATION:
   1: 1..1;
   2: 2..2;
   3: 3..3;
   4: 4..4;

TREE
    BINARY;
```

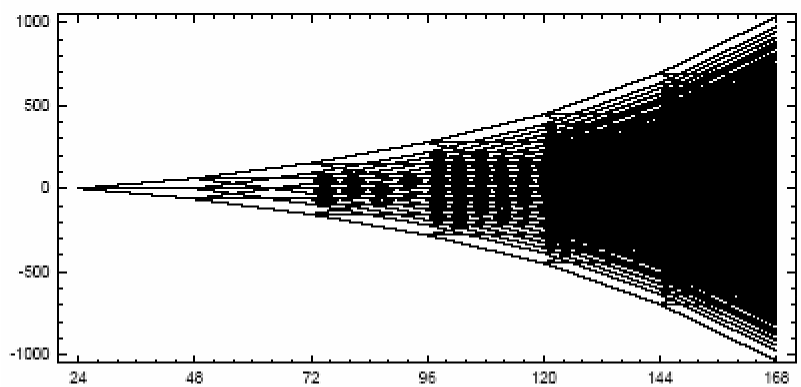
SAMPL

```
set T : =  4;

scenarioset  Scen:= 1..8;

probability param Prob{Scen};

random param PR{T, I, Scen};

stages theStages :=

    1  {t IN T: t=1}
    2  {t IN T: t=2}
    3  {t IN T: t=3}
    4  {t IN T: t=4};

tree theTree:=twostage(2);
```

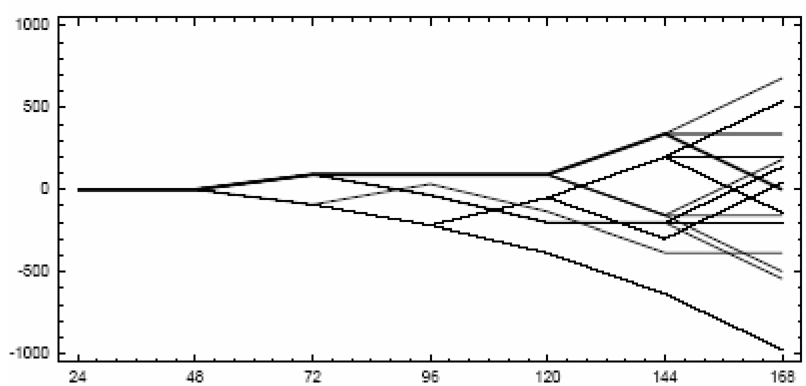
Other Tools and Frameworks

- Scenario (tree) management
  - Generation
    - From random variables to scenarios
    - SAMPL, SMPL, AIMMS, Mosel tools for tree generation
  - Reduction
    - ScenRed, (Römisch et. al.)
- Framework
  - SLP-IOR
  - SPInE (Stochastic Programming Integrated Environment), OptiRisk
    - *Special purpose* scenario generators
    - Connection to SAMPL and SMPL
    - FortSP SP solver

ScenRed (Römisch et. al.)



Original load scenario tree



Reduced load scenario tree / backward

- Find good approximation of original scenario tree of significant smaller size.

Observations

- Skills for SP Application
  - Application domain expert
  - Modeling expert
  - Statistic/Stochastic expert (scenario generation)
  - Algorithmic expert
- Focus has shifted from solving SP to represent/model SP
- Little progress in specialized solver technology
- GUPOR: *Large Scale Stochastic Programming: Still an unsolved problem in OR* (Birge, 2006)

GAMS Approach to SP

- Emphasis on performance:
  - Support 64bit OS to overcome memory issues
  - Efficient out-of-core solver runs
  - Reduce SP problem size (GAMS/ScenRed)
  - Make professional special purpose SP solvers available
    - GAMS/DECIS *(GAMS/OSL-SE)*
  - Utilize emerging grid computing technology
  - High performance data exchange
- Minor modifications to GAMS language
  - Emphasize functionality over beauty

Wait-and-see Model

- Independent problems → Solve in parallel
- GAMS Grid Facility
  - Supports SMP and various grid engines

A pool of connected computers managed and available as a common computing resource

- Effective sharing of CPU power
- Massive parallel task execution
- Scheduler handles management tasks
- E.g. Condor, Sun N6 Grid Engine, Globus
- Can be rented or owned in common

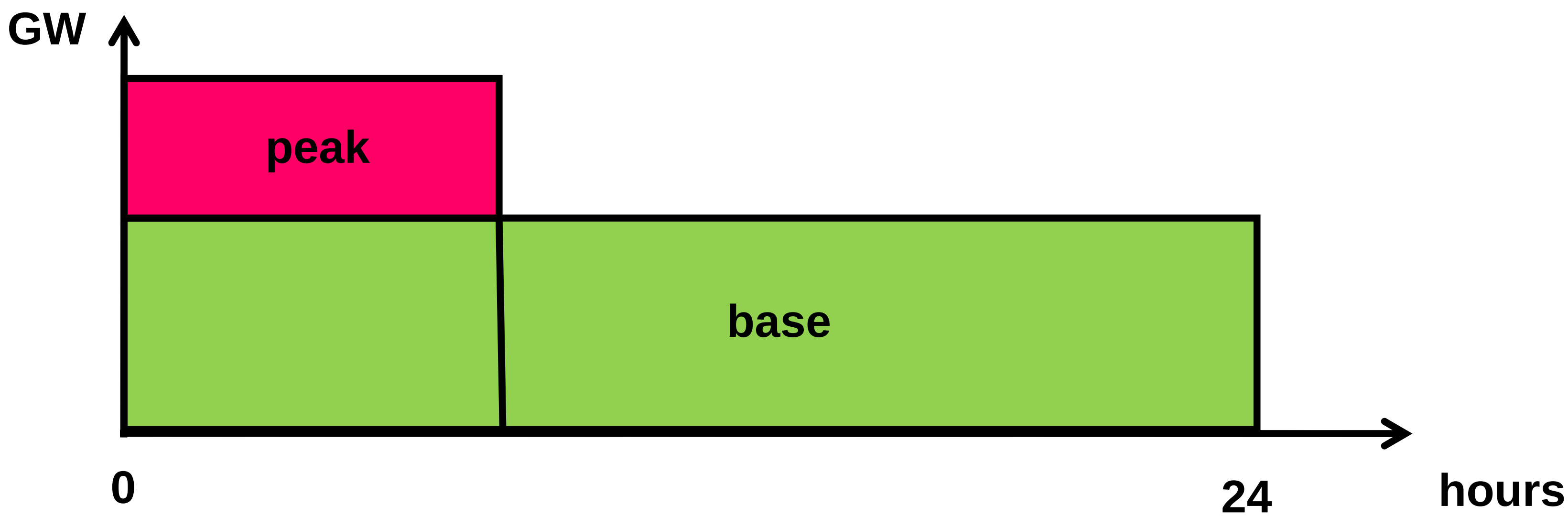
Examples

- Too few (lots of repeats)
- Too simple
- Difficult to reproduce
- Important steps in example missing
  - Creating stochastic data/scenarios
  - Analyzing results (vast amount of data)

- Power Expansion Problem (GAMS/DECIS, AIMMS)
  - Simple, small, demand scenarios given
  - But at least reproducible

Power Expansion Problem

- Determine capacity expansion for different power plants (coal, hydro, nuclear) based on uncertain demand
- Cost structure: capital cost + operating cost
- Demand given as load block:

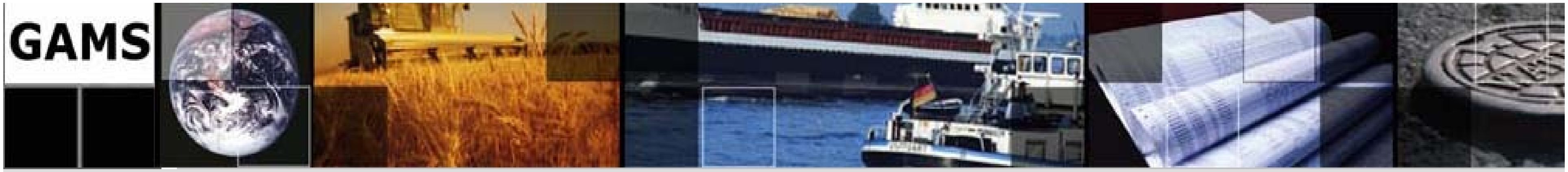


- Additional constraint: No nuclear in peak.
- Recourse: Purchase electricity

GAMS

Conclusion

- Stochastic Programming still challenging and developing field
  - GUPOR: *Uncertainty: An OR Frontier* (Greenberg, 2006)
- Lack of solution technology for *n*-stage SLP limits the dissemination of SP
- Reduce the skill level for SP applications
- Collection of comprehensive & reproducible examples could help to *spread the word*



Contacting GAMS

Europe:

GAMS Software GmbH
Eupener Str. 135-137
50933 Cologne
Germany
Phone: +49 221 949 9170
Fax: +49 221 949 9171
http://www.gams.de

USA:

GAMS Development Corp.
1217 Potomac Street, NW
Washington, DC 20007
USA
Phone: +1 202 342 0180
Fax: +1 202 342 0181
http://www.gams.com