# GAMS
# Recent Enhancements

**Jan-Hendrik Jagla** jhjagla@gams.com

**Michael Bussieck** mbussieck@gams.com

**GAMS Software GmbH** www.gams.de

**Saarbrücken, September 5th 2007**

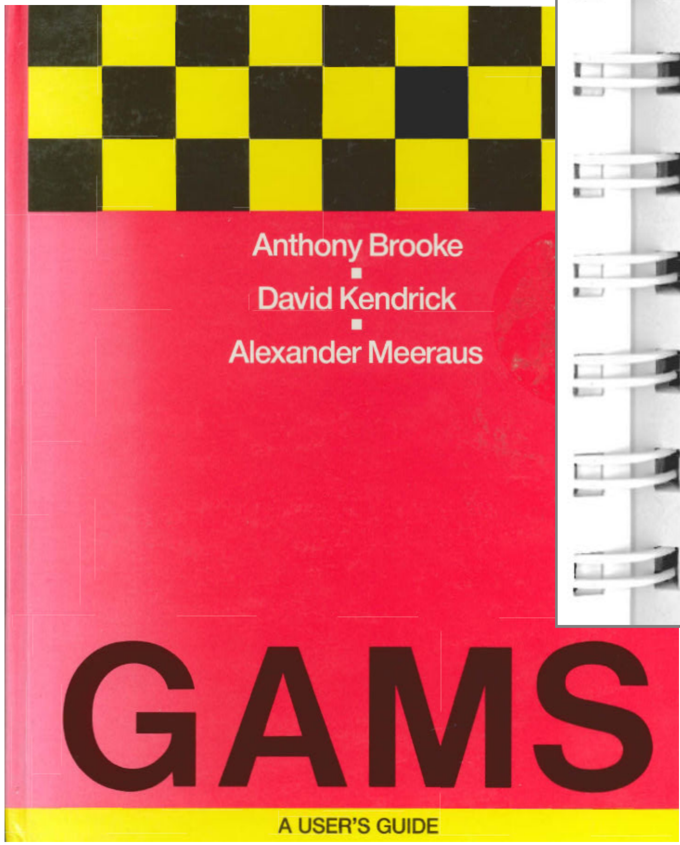
# Agenda

- Then and now
- Recent Enhancements
- Extending Algebraic Modeling
- Well then?


# Agenda

- Then and now
- Recent Enhancements
- Extending Algebraic Modeling
- Well then?


# Then …




In Table 17.1 we list sizes and attributes of representative models that are "large" in the sense that they are near the limit of what is practical on a personal computer, along with the model generation time (GAMS) and solution time (solver), both in minutes. These examples were run on an 8 MHz AT with an 80287 coprocessor and 640K of RAM. The times shown are to give you a rough idea of what is possible: these are not precisely controlled benchmarks, and we have a host of performance improvements in mind for the near future.

**Table 17.1:** Problem Characteristics

| *Name* | *Number of Rows* | *Number of Columns* | *Number of Nonzeroes* | *Generation Time*[a] | *Solution Time*[a] | *Iterations* | *Solver* |
|---|---|---|---|---|---|---|---|
| DINAMICO | 318 | 425 | 4156 | 3.0 | 30.1 | 628 | MINOS |
| SARF | 532 | 542 | 3949 | 37.7 | 115.8 | 2775 | MINOS |
| FERTD[b] | 458 | 2968 | 7252 | 11.4 | 28.3 | 1368 | ZOOM |
| CAMCGE[c] | 243 | 280 | 1356 | 0.8 | 7.0 | 189 | MINOS |
| GANGES[d] | 274 | 357 | 1405 | 1.8 | 7.3 | 187 | MINOS |
| YEMCEM[e] | 168 | 258 | 953 | 0.9 | 7.6 | 600 | ZOOM |
| EGYPT[f] | 281 | 618 | 3168 | 4.0 | 25.3 | 1551 | ZOOM |

[a] Measured in minutes.
[b] The problem is too big for MINOS. ZOOM was used instead.
[c] A nonlinear problem. 63% of the non-zeroes are nonlinear.
[d] A nonlinear problem. 58% of the non-zeroes are nonlinear.
[e] A mixed binary problem, with 55 binary variables (solved with a relative termination criterion of 10%).
[f] A linear problem, solved using XMP which is contained within ZOOM.

**GAMS Users Guide (1988)**

GAMS

# … and now

| | Type | s in 1988 | s in 2007 | Improvement Factor |
|---|---|---|---|---|
| camcge | NLP | 468 | 0.031 | 15097 |
| dinamico | LP | 1986 | 0.125 | 15888 |
| egypt* | MIP | 1758 | 0.015 | 117200 |
| fertd* | MIP | 2382 | 0.062 | 38419 |
| ganges | NLP | 546 | 0.109 | 5009 |
| sarf | LP | 9210 | 0.139 | 66259 |
| yemcem* | MIP | 510 | 0.140 | 3643 |

à Hardly predictable how much performance gain comes from hardware and how much from software

* MIP 1988 solver ZOOM, 2007 solver CPLEX

GAMS

# Agenda

- Then and now
- **Recent Enhancements**
- Extending Algebraic Modeling
- Well then?


# Release Notes

**New Solvers**

- **COIN-OR Solvers** **(http://www.coin-or.org/)**
  - MINLP solver: CoinBonmin
- **AlphaECP**
  - MINLP solver
  - Extended Cutting Plane method by T. Westerlund and T. Lastusilta (Abo Akademi University, Finland)
- **LINDOGlobal**
  - finds proven optimal solutions to non-convex MINLP
  - Global Optimization Solver from LINDO Systems,Inc

**Improvements**

- **BARON, CONOPT, CPLEX, MOSEK, XPRESS,…**

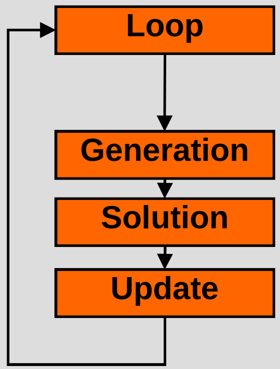

# Multiple Threads

- **CPLEX**
  - **parallel extension** for B&B and interior point solver
  - **concurrent optimizer**
  - academic license includes 4 threads
- **MOSEK**
  - **parallel extension** for the interior solver comes free of charge
  - **concurrent optimizer**
- **XPRESS**
  - **parallel extension** for B&B and interior point solver
  - academic license includes 4 threads
- **XA (XAPAR)**

# Grid Computing

- pool of connected computers managed and available as common computing resource
- e.g. Condor, Sun Grid Engine

```
Loop(p(pp),
   ret.fx = rmin +(rmax-rmin)
     /(card(pp)+1)*ord(pp) ;
   Solve minvar min var using miqcp;
   xres(i,p)         = x.l(i);
   report(p,i,'inc') = xi.l(i);
   report(p,i,'dec') = xd.l(i)
  );
```



# Minor Changes to Model

gamside: C:\Documents and Settings\Jan\My Documents\presentation\2007-07-EURO-Prag\Workshop\meanvar_grid\meanvar.gpr

File Edit Search Windows Utilities Help

c:\documents and settings\jan\my documents\presentation\2007-07-euro-prag\wor...

meanvar.gms

```
   xres('var',p)      = v.l;
   xres('status',p)   = var1.modelstat;
   vmin = v.l; );

Loop(p(pp),
   v.fx = vmin + (vmax-vmin)/(card(pp)+1)*ord(pp) ;
   Solve var1 maximizing m using nlp ;
   xres(i,p)          = x.l(i);
   xres('mean',p)     = m.l;
   xres('var',p)      = v.l;
   xres('status',p)   = var1.modelstat; );

Display xres;
```

C:\Documents and Settings\Jan\My Documents\presentation\2007-07-EURO-Prag\Wo...

meanvar.gms | meanvar_edited.gms

```
   xres('var',p)      = v.l;
   xres('status',p)   = var1.modelstat;
   vmin = v.l; );

$if not set grid $set grid 0
parameter handle(p) Grid handle;

if(not %grid%,
   Loop(p(pp),
      v.fx = vmin + (vmax-vmin)/(card(pp)+1)*ord(pp)
      Solve var1 maximizing m using nlp ;
      xres(i,p)          = x.l(i);
      xres('mean',p)     = m.l;
      xres('var',p)      = v.l;
      xres('status',p)   = var1.modelstat; );
else
   var1.solvelink=3;
   Loop(p(pp),
      v.fx = vmin + (vmax-vmin)/(card(pp)+1)*ord(pp)
      Solve var1 maximizing m using nlp ;
      handle(p) = var1.handle );
   Repeat
      loop(p(pp)$handlecollect(handle(p)),
         xres(i,p)          = x.l(i);
         xres('mean',p)     = m.l;
         xres('var',p)      = v.l;
         xres('status',p)   = var1.modelstat;
         display$handledelete(handle(p)) 'trouble del
         handle(p) = 0 ) ;
      display$sleep(card(handle)*0.2) 'sleep some tim
   until card(handle) = 0 or timeelapsed > 100;
   xres(i,p(pp))$handle(p) = na;
);
Display xres;

execute_unload "portfolio.gdx" xres;
```

# SUNgrid

www.network.com

- On-demand grid computing service operated by Sun Microsystems
- Access to enormous computing power over Internet
- Opteron-based servers with 4 GB of RAM per CPU
- Solaris 10 OS, and Sun Grid Engine 6 software.
- $1 per CPU-hour
- **GAMS Distribution 22.5 available**
- **250 free CPU hours to new users**

www.gams.com/sungrid

à F. Nelissen: “Grid Computing in Finance using an Algebraic Modeling System”, Thursday 2:30-3pm, Room A2 4 1.32

# Agenda

- Then and now
- Recent Enhancements
- **Extending Algebraic Modeling**
- Well then?


# Algebraic Modeling Languages

- Problem format is old/traditional

$$\min_{x} f(x) \text{ s.t. } g(x) \leq 0, h(x) = 0$$

- Limited support for symbolic reformulations

  – logical constructs
  – constraints tightening, softening
  – stochastic programming

è Automated symbolic reformulations will gain importance


# Symbolic Reformulations and GAMS

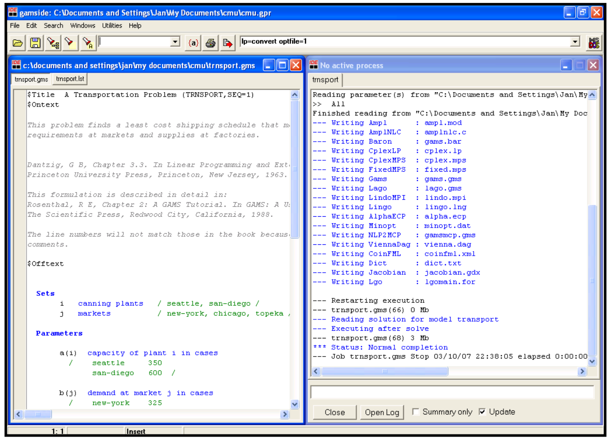
- GAMS/DECIS
  - solves two-stage stochastic linear programs with recourse
  - two-stage decomposition (Benders)
  - stores only one instance of the problem and generates scenario sub-problems as needed
  - solution Strategies (Universe problem/Importance sampling)

- GAMS/NLPEC
  - solves MPECs as NLPs
  - 23 different reformulation strategies

- GAMS/PATHNLP
  - solves NLPs as MCPs
  - internal reformulation via KKT conditions
  - requires 1st and 2nd order derivatives

# Symbolic Reformulations and GAMS

- GAMS/Convert
  - model translation tools
  - GAMS à other formats/languages
  - makes use of infix notation available
  - GAMS è ENLP
  - NLP2MCP
  - CHull è LOGMIP



Option GAMS

- converts indexed input model into scalar model
- “standardized” model format
- easier to deal with
- confidentiality issues
- nonlinearity information is still available

# NLP2MCP

- Michael C. Ferris and Jeffrey D. Horn (1998)

  "NLP2MCP: Automatic conversion of nonlinear programs into mixed complementarity problems"

  – Reformulate NLP via KKT into MCP model
  – Requires 1st derivatives

- Original approach
  – individual tool that translates an indexed model
  – tough approach due to potential beastiness of NLPs

- New approach
  – integrated in GAMS/Convert
  – result is the source of a scalar MCP model


# NLP2MCP

- Why convert to MCP
  - Second order information implicitly available
  - Remove “superbasic” dependence
  - Exploit multiplier information
  - Incorporate into MPEC

- Likely that MCP solver will find a solution
  - Solution is only guaranteed to be feasible for the original problem
  - In the convex case, every KKT point corresponds to a global solution of the NLP


# Extended Nonlinear Programming (ENLP)

Reference is paper by R.T. Rockafellar (1999)

- Classical problem

$$\begin{aligned} \min_{x_1,x_2,x_3} \quad & \exp(x_1) \\ \text{s.t.} \quad & \log(x_1) = 1 \\ & x_2^2 \leq 2 \\ & x_1/x_2 = \log(x_3), 3x_1 + x_2 \leq 5, x_1 \geq 0, x_2 \geq 0 \end{aligned}$$

- Soft penalization of constraints

$$\begin{aligned} \min_{x_1,x_2,x_3} \quad & \exp(x_1) + 5\,\|\log(x_1) - 1\|^2 + 2\max(x_2^2 - 2, 0) \\ \text{s.t.} \quad & x_1/x_2 = \log(x_3), 3x_1 + x_2 \leq 5, x_1 \geq 0, x_2 \geq 0 \end{aligned}$$

# Extending NLPs automatically

- Motivation
  - automates tedious algebra writing
  - avoids mistakes
  - infeasibility analysis
  - real world applications always need a solution
- Implement a “solver” that
  - extends an NLP based on user provided information
  - passes the modified model to an NLP solver
  - reads the solution back
  - is aware of a class of different extension functions


# GAMS “Solver” ENLP

- Converts input model to a scalar NLP or MCP
- Reads ENLP information file to extend the model

```
$onecho > %gams.scrdir%enlpinfo.scr
e1 sqr  5
e2 MaxZ 2
$offecho
```

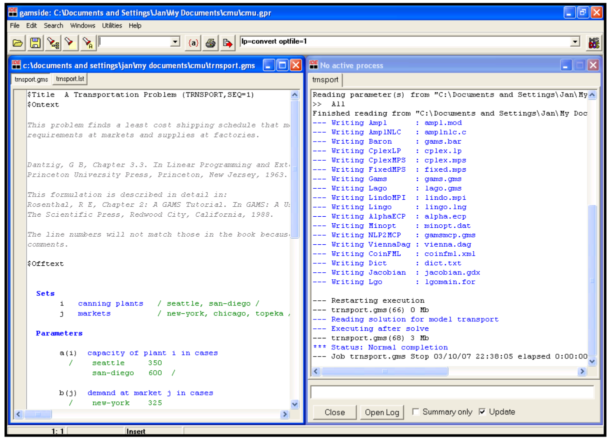
- Solves the created ENLP / EMCP
- Reads solution back
- Solver options
  - SolveasMCP — ENLP will be converted to and solved as MCP
  - SubSolver — subsolver to run
  - SubSolverOpt — optfile value to pass to the subsolver
  - Terminate — terminate after generating scalar GAMS model
  - …

# Symbolic Reformulations and GAMS

- GAMS/Convert
  - model translation tools
  - GAMS à other formats/languages
  - makes use of infix notation available
  - GAMS
  - NLP2MCP
  - Chull

è **ENLP**

è **LOGMIP**

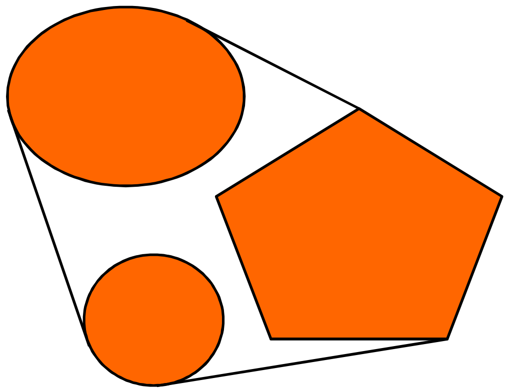


# Convex Hull

- Convex Hull reformulation of linear and nonlinear models with disjunctions
- User provides disjunction information

```
file dj2 / '%gams.scrdir%loginfo.scr' /;  dj.nd=0; dj.nw=0; dj.lw=0;
put dj2 '* convex hull for example 1';
loop(lt(j,jj),
   put / 'disj ' y.tn(j,jj) ' ' seq.tn(j,jj) ' else ' seq.tn(jj,j));
putclose;
```

- Result is a scalar GAMS model representing the Convex Hull




# Logical Mixed Integer Programming (LogMIP)

- Developed by Aldo Vecchietti and Ignacio E. Grossmann

  http://www.logmip.ceride.gov.ar/

- Add-on modeling framework to formulate models with disjunction and logic propositions

- Language Compiler

- Solution algorithms and techniques for solving linear and nonlinear disjunctive programming problems

- Add-on tool as of GAMS Distribution 21.7 (April 2005)


# LogMIP Example

$$\min Z = T$$

$$\begin{aligned}
s.t. \quad & T \geq x_1 + 8 \\
& T \geq x_2 + 5 \\
& T \geq x_3 + 6
\end{aligned}$$

$$\begin{bmatrix} Y_1 \\ x_1 - x_3 + 5 \leq 0 \end{bmatrix} \vee \begin{bmatrix} \neg Y_1 \\ x_3 - x_1 + 2 \leq 0 \end{bmatrix}$$

$$\begin{bmatrix} Y_2 \\ x_2 - x_3 + 1 \leq 0 \end{bmatrix} \vee \begin{bmatrix} \neg Y_2 \\ x_3 - x_2 + 6 \leq 0 \end{bmatrix}$$

$$\begin{bmatrix} Y_3 \\ x_1 - x_2 + 5 \leq 0 \end{bmatrix} \vee \begin{bmatrix} \neg Y_3 \\ x_2 - x_1 \leq 0 \end{bmatrix}$$

$$T, x_1, x_2, x_3 \geq 0$$

$$Y_k \in \{true, false\}, k = 1,2,3.$$

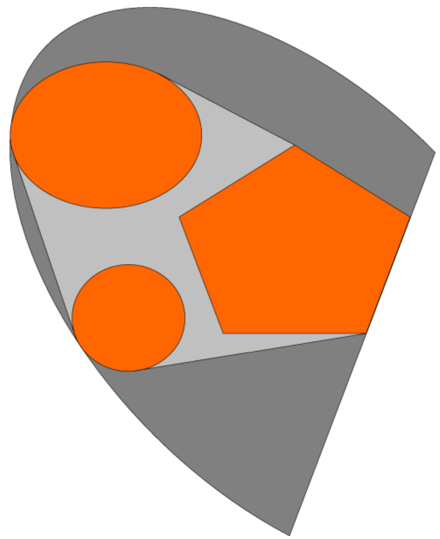
| Stage / Job | 1 | 2 | 3 |
|---|---|---|---|
| A | 5 | - | 3 |
| B | - | 3 | 2 |
| C | 2 | 4 | - |

```
$onecho  > %gams.scrdir%logdisj.scr
Disjunction d(j,jj);
d(j,jj) with lt(j,jj) is
if pr(j,jj)
   then seq(j,jj);
   else seq(jj,j);
endif;
$offecho
```

**Raman & Grossmann (1994)**

# LogMIP “Solver”

- Reformulation
  - Convex Hull (only linear)
  - Big M (only linear)
  - Need MIP solver



- Logic based Method
  - Logic-based Outer Approximation
  - Solves a series of NLP and MIP sub-problems
  - Needs NLP an MIP solver


# CPLEX Indicator Constraints

- New way of expressing relationships among variables
- Specify binary variable to control whether or not a constraint takes effect
- Can be numerically more robust and accurate than conventional Big M formulations

```
file copt / cplex.opt /; put copt '* inicators for example 1'
loop(lt(j,jj),
   put / 'indic ' seq.tn(j,jj) '$' y.tn(j,jj) yes
       / 'indic ' seq.tn(jj,j) '$' y.tn(j,jj) NO   );

m.optfile=1; option mip=cplex;

solve m using MIP minimizing t;
```

GAMS

# Agenda

- Then and now
- Recent Enhancements
- Extending Algebraic Modeling
- **Well then?**


# Well then?

- GAMS already supports a wide collection of established mathematical programming classes

  (LP,MIP,QCP,MIQCP,NLP,DNLP,MINLP,MCP,MPEC,CNS,Global)

- New research breaks out of traditional MP classes
  - broaden algebraic modeling
  - proven research add-ons to GAMS
  - engaged in systematic modification of constraints (activate, deactivate, soften, tighten)
  - But! Each approach has its individual intermediate format

GAMS

# Automatic Reformulation Framework

à Need of an integrated framework for automated mathematical programming reformulations

- provide new facilities for seamless integration of new model types
- benefit models with constructs like
  - disjunctions
  - indicator constraints
  - extended nonlinear programs
  - conditional value at risk
  - stochasticity (chance constraints)

# Evolution in the GAMS way

0. – try as research code
   – committed to backward compatibility
   √

1. – analyze big picture
   – generalize
   √

2. – implement sub-language
   – does it proof itself?
   →22.6

3. – define standard
   – integrate in GAMS language
   – work with solver developers
   ?


# 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

info@gams.de
support@gams-software.com

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

sales@gams.com
support@gams.com