Extended Mathematical Programming in GAMS

Jan-Hendrik Jagla jhjagla@gams.com
Michael Ferris ferris@cs.wisc.edu
Alex Meeraus ameeraus@gams.com

GAMS Software GmbH
www.gams.de
GAMS Development Corp.
www.gams.com

IFORS sandton
13-18July 2008

Agenda

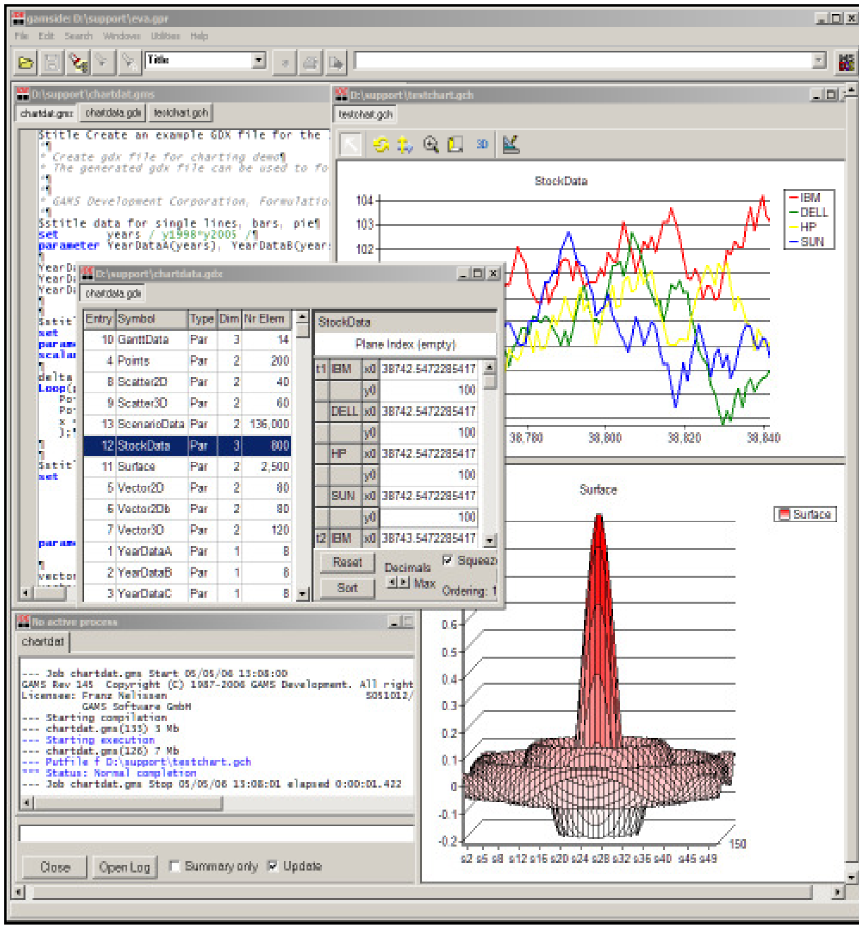
- **G**eneral **A**lgebraic **M**odeling **S**ystem
- New Solution Concepts
- **E**xtended **M**athematical **P**rogramming

GAMS

Agenda

- **G**eneral **A**lgebraic **M**odeling **S**ystem
- New Solution Concepts
- **E**xtended **M**athematical **P**rogramming

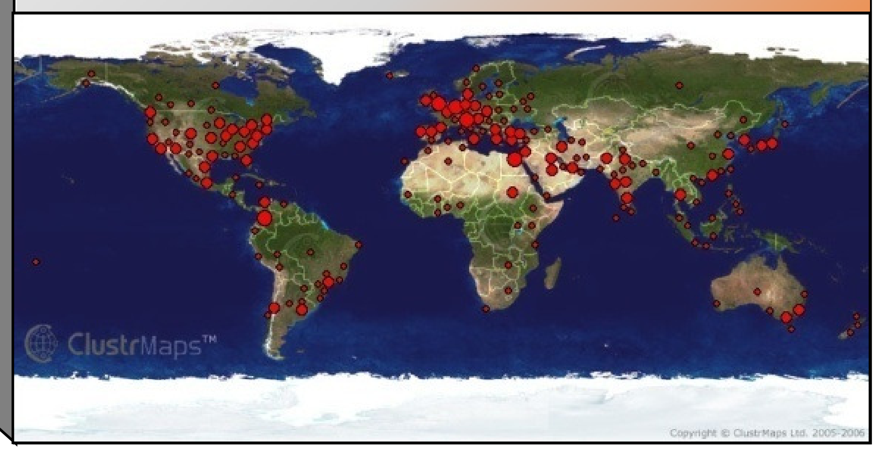
GAMS at a Glance



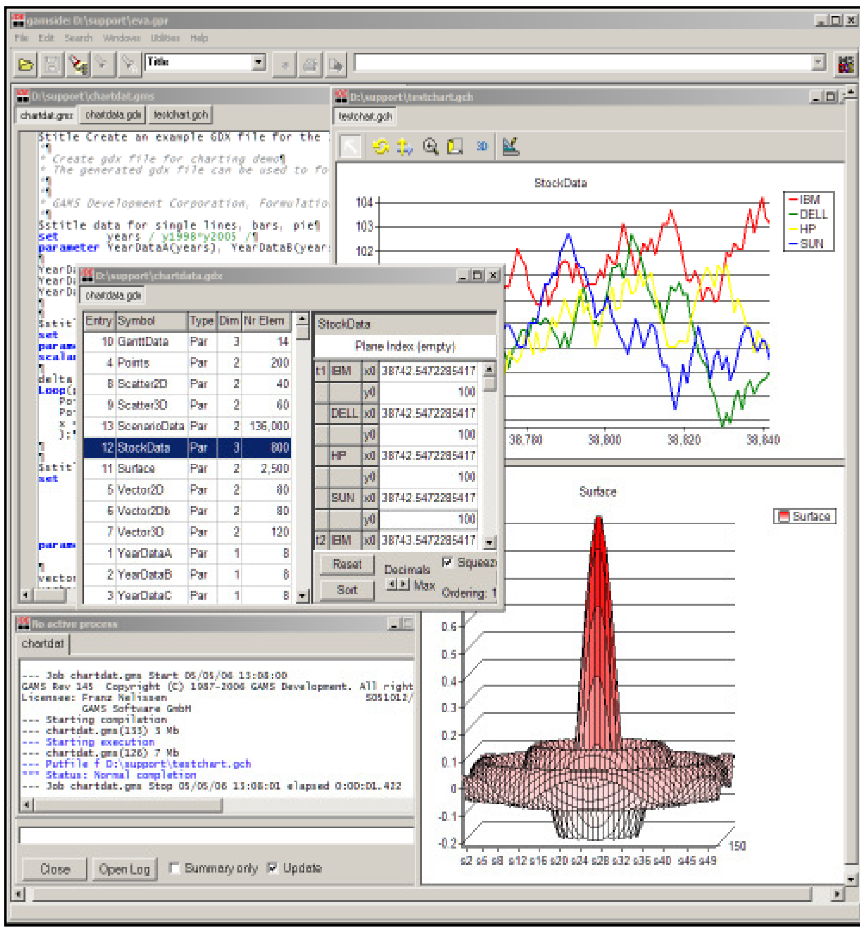
General **A**lgebraic **M**odeling **S**ystem

- Roots: World Bank, 1976
- Went commercial in 1987
- GAMS Development Corp.
- GAMS Software GmbH

- Broad academic & commercial user community and network



GAMS at a Glance



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

- Algebraic Modeling Language
- 25+ Integrated Solvers
- 10+ Supported MP classes
- 10+ Supported Platforms
- Connectivity- & Productivity Tools
  - IDE
  - Model Libraries
  - GDX, Interfaces & Tools
  - Grid Computing
  - Benchmarking
  - Compression & Encryption
  - Deployment System
  - …

Agenda

- **G**eneral **A**lgebraic **M**odeling **S**ystem
- New Solution Concepts
- **E**xtended **M**athematical **P**rogramming

Traditional but fundamental concept

Different layers with separation of

- model and data
- model and solution methods
- model and operating system
- model and interface

Current state: Model-Side

- Traditional problem format

$$\min_{x} c(x) \quad s.t.\ A_1(x) \le b_1,\ A_2(x) = b_2$$

- Support for complementarity constraints

- Interactions between models possible
  - Series of models
  - Scenario analyses / parallelized model runs
  - Iterative sequential feedback
  - Decomposition

GAMS

Current state: Solver-Side

Support of a wide collection of established MP classes through solver cluster!

➔ **Tremendous algorithmic and computational progress**

LP

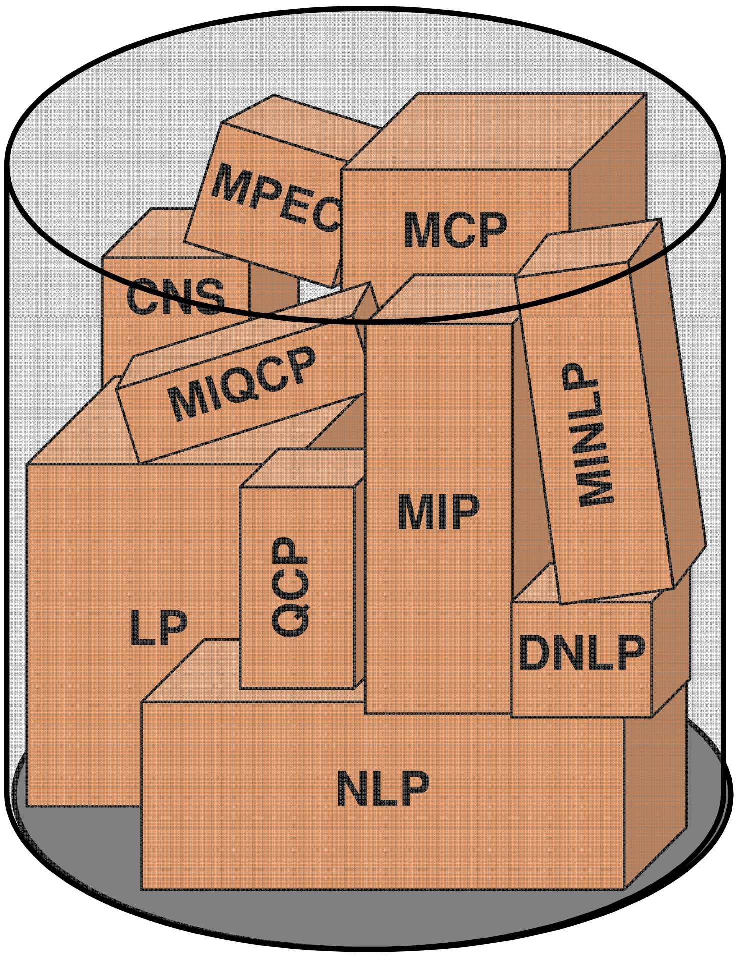
in fact only restricted by available memory

MIP

- Some (academic) problems still unsolvable
- Commercial problems mostly docile

NLP/MINLP

- Predictions are problem and data specific, global vs. local solutions



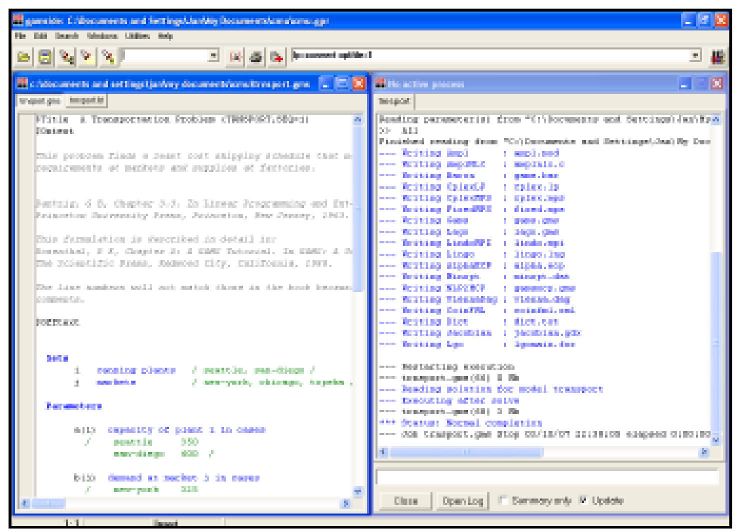
Extended Interface between Model&Solver

Model Translation Tools

- GAMS/Convert
  - GAMS → other formats/languages
  - Symbolic model translations and processing are very fast
  - Algebraic information still available (“source to source”)
  - E.g.
    - NLP2MCP

      Converts non-integer model into a scalar MCP model
    - CHull

      Creates the convex hull of a (nonlinear) disjunctive program



Extended Interface between Model&Solver

Solvers that are based upon reformulation

- 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 MP with Equilibrium Constraints (MPECs) as NLPs
  - 20+ different reformulation strategies
- GAMS/PATHNLP
  - solves NLPs as MCPs
  - internal reformulation via KKT conditions
  - requires 1^{st} and 2^{nd} order derivatives

GAMS

Extended Interface between Model&Solver

Hybrid Approaches

- traditional model representation
- additional information
  - Mathematical Programming System for General Equilibrium analysis (MPSGE)
  - Logical Mixed Integer Programming (LogMIP)
    - Reformulation and logic-based methods on Generalized Disjunctive Programs (GDP)
  - Indicator constraints (CPLEX)
    - Alternative to conventional BigM formulations

Extended Interface between Model&Solver

New Solution Concepts

- Extended Nonlinear Programs
- Embedded Complementarity Systems
- Bilevel Programs
- Disjunctive Programs

- Breakouts of traditional MP classes
- No conventional syntax
- Limited support with common model representation
- Incomplete/experimental solution approaches
- Lack of reliable/any software

What now?

Do not:
- overload existing GAMS notation right away !
- attempt to build new solvers right away !

But:
- Use existing language features to specify additional model features
- Distribute information as part of the production system
- Express extended model in symbolic form and apply existing matured solution technology

→ **Extended Mathematical Programming (EMP)**

Agenda

- **G**eneral **A**lgebraic **M**odeling **S**ystem
- New Solution Concepts
- **E**xtended **M**athematical **P**rogramming

GAMS "Solver" EMP

- Takes responsibility to offer translation services
- Uses existing language features to specify additional model features
- Expresses extended model in symbolic form and passes it to existing solution methods via embedded GAMS calls
- Reads solution back into original space
- Facilitates to write out the reformulated model ("Look and Feel")

Extended Nonlinear Programming

Soft penalization of constraints

- Model:

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

- Additional information:

```
$onecho > %emp.info%
Adjustequ
e1 sqr  5
e2 MaxZ 2
$offecho
```

```
$onecho > %gams.scrdir%empinfo2.scr
Strategy MCP
Adjustequ
e1 sqr  5
e2 MaxZ 2
$offecho
```

- EMP Tool creates the NLP model (or the MCP via KKT) :

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

Embedded Complementarity Systems

- Models with side constraints/variables:

$$\begin{aligned} \min_x \quad & f(x,y) \\ \text{s.t.} \quad & g(x,y) \le 0 \quad (\perp \lambda \ge 0) \end{aligned}$$

$$H(x,y,\lambda) = 0 \qquad (\perp y \text{ free})$$

- Additional Information:

```
$onecho > %emp.info%
dualequ H y
dualvar λ g
$offecho
```

- EMP Tool creates the MCP model:

$$\begin{aligned} \nabla_x \mathcal{L}(x,y,\lambda) \quad & \perp x \text{ free} \\ -\nabla_\lambda \mathcal{L}(x,y,\lambda) \quad & \perp \lambda \ge 0 \\ H(x,y,\lambda) = 0 \quad & \perp y \text{ free} \end{aligned}$$

ECS Example

- Rutherford, Thomas F. (http://www.mpsge.org/nlptarget/)

```
parameter
        kterm           Terminal capital stock

UTIL..              UTILITY =E= SUM(t, 10 * dfactor(t) * L(t) * LOG(C(t)/L(t)));
CC(t)..                C(t) =E= Y(t) - I(t);
YY(t)..                Y(t) =E= phi * L(t)**(1-kvs) * K(t)**kvs;
KK(t)..                K(t) =L= (1-delta)**10 * K(t-1) + 10 * I(t-1) + kinit$tfirst(t);
TERMCAP..             kterm =E= sum(tlast, (1-delta)**10 * K(tlast) + 10 * I(tlast));

model ramsey NLP Model using parameter kterm /all/;

set iter /iter1*iter20/;

kterm = kinit * power(1+g,card(t));

parameter         invest(t,iter)   Investment in successive iterations
                  kt(iter)         Terminal capital stock in successive iterations;

loop(iter,
        kt(iter) = kterm;
        solve ramsey maximizing UTILITY using NLP;
        invest(t,iter) = I.L(t);

        kterm = sum(tlast(t), K.L(tlast) * Y.L(t)/Y.L(t-1));
);
```

EMP Formulation

```
*Substitute TERMCAP of NLP by TERMCAPV (using variable KTERMV instead of parameter kterm)
TERMCAPV..  KTERMV =E= sum(tlast, (1-delta)**10 * K(tlast) + 10 * I(tlast));

*First-order-condition for terminal capital stock variable
SSTERM.. sum(tlast(t),I(t)/I(t-1) - Y(t)/Y(t-1)) =E= 0;

model ramseynlpd /UTIL,CC,YY,KK,TERMCAPV,SSTERM/;

$onecho > %emp.info%
dualequ SSTERM KTERMV
$offecho

option nlp=emp;

solve ramseynlpd maximizing UTILITY using nlp;
```

```
Extended Mathematical Programming (EMP)
----------------------------------------
--- EMP Summary  (errors=0)
    Adjusted Equations = 0
    Dual Variable Maps = 0
    Dual Equation Maps = 1
    Bilevel Followers  = 0
    Disjunctions       = 0
--- The model C:\home\distrib\tvis_alpha\convtest\emp\225a\emp.scr will be solved by GAMS
---
```

Hierarchical Models

- Bilevel Program:

$$
\begin{aligned}
\min_{x,y} \quad & f(x,y) \\
\text{s.t.} \quad & g(x,y) \leq 0, \\
& y \text{ solves } \min_{s} v(x,s) \text{ s.t. } h(x,s) \leq 0
\end{aligned}
$$

- Additional Information:

```
$onecho  > %emp.info%
Bilevel x min v h
$offecho
```

- EMP Tool automatically creates an MPEC by expressing the lower level optimization problem through its optimality conditions

Bilevel Model

Conejo A J, Castillo E, Minguez R, and Garcia-Bertrand R; Decomposition Techniques in Mathematical Programming, Springer,Berlin, 2006.

```
variables z,x1,x2,x3,x4,h1,h2,u1,u2,u3,u4,v1,v2,v3,v4;
equations defobj,defh1,defh2,a1,e1,e2;

defobj.. z =e= sqr(x1+x2-2) + sqr(x3+x4-2);
a1.. x1-x2 =e= 3;
```

Outer Problem

```
defh1.. h1 =e= sqr(u1-x1) + sqr(u2-x2) + sqr(u3-x3) + sqr(u4-x4);
e1.. 3*u1 +  u2 + 2*u3 + u4 =e= 6;
```

Inner Problem 1

```
defh2.. h2 =e= sqr(v1-x1) + sqr(v2-x2) + sqr(v3-x3) + sqr(v4-x4);
e2.. v1 +  v2+ v3 + 2*v4 =e= 7;
```

Inner Problem 2

```
model bilevel / all /
```

EMP Information File + EMP Summary Log

```
option nlp=emp;

$onecho > %emp.info%
bilevel x1 x2 x3 x4
min h1 defh1 e1
min h2 defh2 e2
$offecho

solve bilevel us nlp min z;
```

```
Extended Mathematical Programming (EMP)
----------------------------------------
--- EMP Summary  (errors=0)
    Adjusted Equations = 0
    Dual Variable Maps = 0
    Dual Equation Maps = 0
    Bilevel Followers  = 2
    Disjunctions       = 0
--- The model C:\home\distrib\tvis_alpha\convtest\emp\225a\emp.scr will be solved by GAMS
---
```

Disjunction Example

Raman & Grossmann, Comp. & Chem. Eng., 18, 7, p.563-578, 1994.

- Three jobs (A,B,C) must be executed sequentially in three steps, but not all jobs require all the stages. Once a job has started it cannot be interrupted.

- The objective is to obtain the sequence of task, which minimizes the completion time.

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

Data Definition

```
table p(j,s) processing time

    1    2    3
A   5         3
B         3   2
C   2    4

alias (j,jj),(s,ss);

parameter c(j,s)   stage completion time
          w(j,jj) maximum pair wise waiting time
          pt(j)    total processing time;
set       less(j,jj) upper triangle;

c(j,s)   = sum(ss$(ord(ss)<=ord(s)), p(j,ss));
w(j,jj) = smax(s, c(j,s) - c(jj,s-1));
pt(j)    = sum(s, p(j,s));
less(j,jj)= ord(j) < ord(jj);
```

GAMS

Basic Model Definition

```
variables t          completion time
          x(j)       job starting time
positive variable x;

equations comp(j)    job completion time
          seq(j,jj)  job sequencing j before jj;

comp(j).. t =g= x(j) + pt(j);

seq(j,jj)$(not sameas(j,jj))..  x(j) + w(j,jj) =l= x(jj);
```

Above equation is incomplete!

If (j,jj) is active then (jj,j) should be relaxed

Traditional BigM Formulation

```
binary variable y(j,jj) job precedence;

parameter big the famous big M;

big = sum(j, pt(j));
big=100000;

seq(j,jj)$(not sameas(j,jj))..

x(j)  + w(j,jj)  =l= x(jj)  + big*(      y(j,jj)  $less(j,jj)
                                   + (1-y(jj,j))$less(jj,j));

model m / all /; m.optcr=0;

solve m using MIP minimizing t;
```

EMP Disjunction Formulation

```
seq(j,jj)$(not sameas(j,jj))..  x(j) + w(j,jj) =l= x(jj);

model m / all /;

file emp / '%emp.info%' /; put emp '* EMP for example 1';
loop(less(j,jj),
   put / 'disjunction * ' seq.tn(j,jj) ' else ' seq.tn(jj,j) );
putclose;

option mip=emp;

solve m using MIP minimizing t;
```

```
* EMP for example 1
disjunction * seq('A','B') else seq('B','A')
disjunction * seq('A','C') else seq('C','A')
disjunction * seq('B','C') else seq('C','B')
```

EMP Info Syntax Summary

- `AdjustEQU equu abs|sqr|maxz|huber|... { weight { param } }`

- `DualEqu {equ var}`
- `DualVar {var equ}`

- `BiLevel {var} { MAX|MIN obj {equ} }`

- `Disjunction [NOT] var|* {equ} { ELSEIF [NOT] var|* {equ} } [ ELSE {equ} ]`

Conclusion

EMP is

- an framework for automated symbolic reformulations
- non-exhaustive and experimental

EMP needs

- **Input from other researchers !!**
  - Automate further reformulation strategies
    - More of the same, boring to some, exiting to others
    - Concurrent strategies
  - Examples from existing publications
    - EMP Library

Conclusion

EMP promotes non-traditional MP classes through:

- Automation of symbolic reformulations to avoid error-prone and time-consuming manual algebra (re)writing
- Availability of theoretical benefits to users from a wide variety
- Solutions through established and powerful solution engines
- Availability of nonstandard model information to solver developers → new algorithms/software?

→ bridge the gap between academia and industry

GAMS Beta 22.8

The GAMS Beta Distribution 22.8 is available for download

http://beta.gams-software.com

- New Solver Libraries, e.g.
  - CPLEX 11.1
  - Coin-OR Solvers
- Experimental solvers offering in-core communication
- Two new model libraries
- New utilities (gdx2xls, invert, xlstalk)
- …

Contacting GAMS

<u>Europe</u>

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

<u>USA</u>

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