Global Optimization with GAMS

Lutz Westermann lwestermann@gams.com

Michael R. Bussieck mbussieck@gams.com

GAMS Software GmbH
www.gams.de

September 5th 2007

Agenda

- Global Optimization Introduction
- GAMS Global Solvers
- Performance & Quality
- Linking a new Code

Agenda

- Global Optimization Introduction
- GAMS Global Solvers
- Performance & Quality
- Linking a new Code

Algebraic Modeling Languages

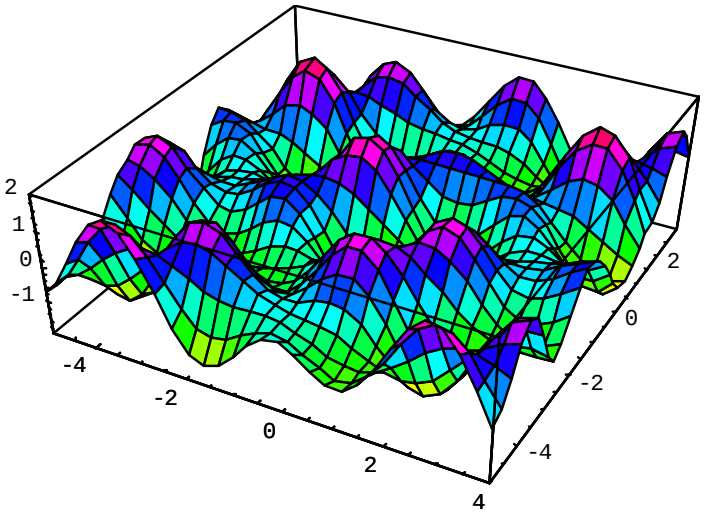
- High-level programming languages for large scale mathematical optimization problems
- Algebraic formulation
  - § Syntax similar to mathematical notation
  - § Does not contain any hints how to process it
- Do not solve optimization problems directly but call appropriate external algorithms (solvers)

Goals

- Support of decision making process
- Efficient handling of mathematical optimization problems
- Simplify model building and solution process
- Increase productivity and support maintainable models

Global Optimization (GO)

- Practical optimization problems are often nonlinear and non-convex
- They may contain disconnected feasible regions with multiple local optima
- The aim of Global Optimization is to find the best solution of all local optima



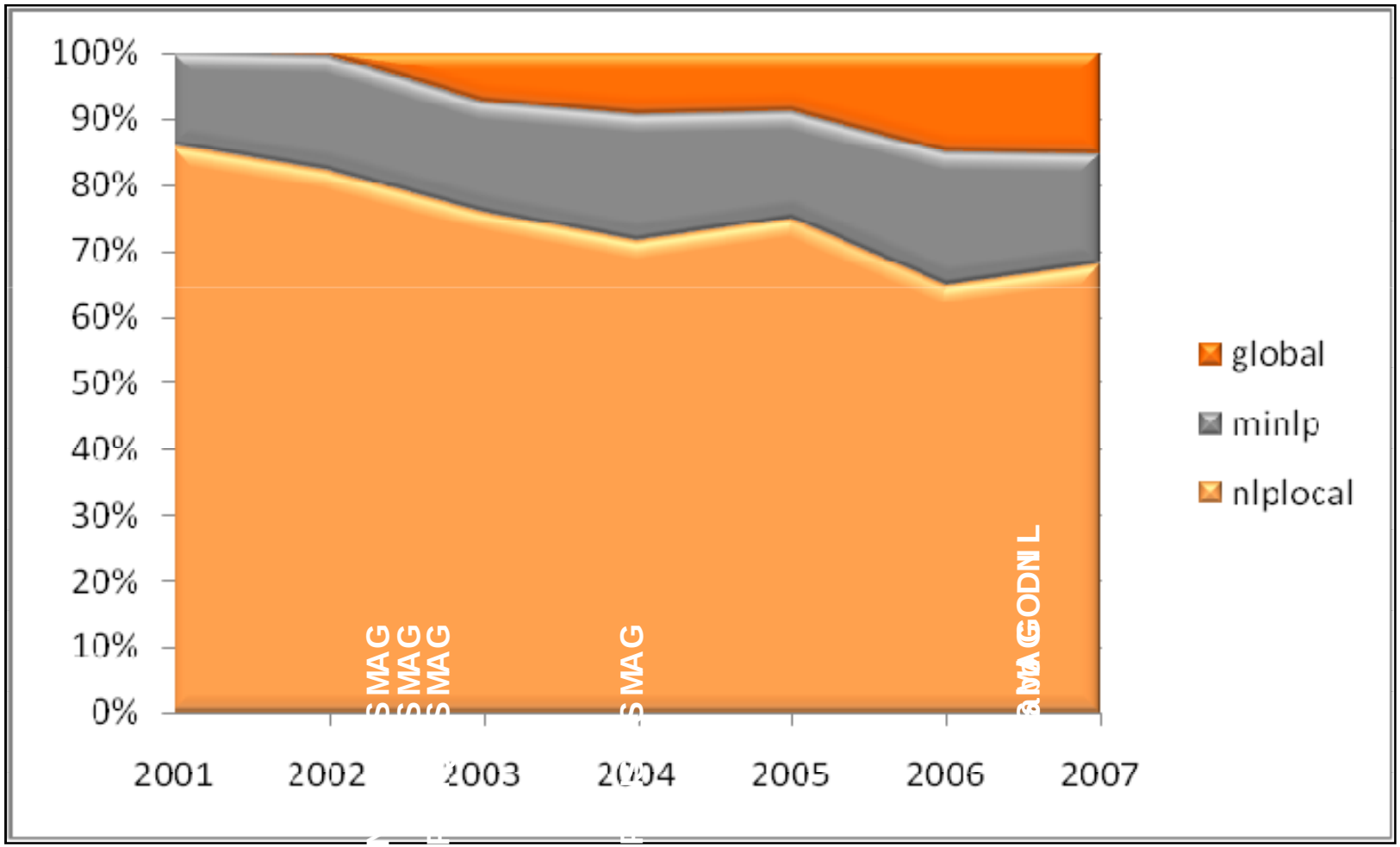
Examples for GO Applications

- Chemotherapy and radiotherapy design
- Chemical data and process analysis
- Differential equations
- Engineering design
- Environmental engineering
- Financial model development
- Laser design
- Packing and loading configuration design
- Staff scheduling
- Vehicle routing and scheduling

Agenda

- Global Optimization Introduction
- **GAMS Global Solvers**
- Performance & Quality
- Linking a new Code

Relevance of GO



GAMS Global Solvers

- **BARON** Branch-and-Reduce Optimization Navigator for proven global solutions by *The Optimization Firm*, USA
- **LGO** Lipschitz Global Optimizer by *Pintér Consulting Services*, Canada
- **OQNLP** OptQuest/NLP Multi-start Solver by *OptTek Systems/Optimal Methods*, USA
- **LINDOGlobal** MINLP solver for proven global solutions by *LINDO Systems*, USA

The solvers differ in the methods they use, in whether they find globally optimal solution with proven optimality, in the size of models they can handle, and in the characteristics of models they accept.

Optimization Methods

- **BARON**
  - Branch-and-bound algorithms
  - Range reduction techniques
  - Under-estimators for objective and constraints
- **LGO**
  - Branch-and-bound based global search
  - Stochastic sampling procedure
- **OQNLP**
  - Starts NLP solvers from a set of starting points chosen by the Scatter Search software OptQuest
- **LINDOGloba**l
  - Branch-and-cut methods
  - Breaks an NLP into a list of sub problems

Model Requirements

- **BARON**
  - LP and NLP solver for sub problems
  - Knowledge about model algebra
- **LGO**
  - Only Lipschitz-continuity of objective function
  - Black box models (external equations)
- **OQNLP**
  - Requirements of local solver used during search
  - Smooth problems
- **LINDOGlobal**
  - Local solver for nonlinear sub problems
  - Instruction list interface

Solution Quality Metrics

- **LGO**
  - Estimated statistical or Lipschitz lower bound
  - Stochastic convergence to global optimum

- **OQNLP**
  - Scatter Search ensures stochastic convergence towards the global optimum

- **BARON & LINDOGlobal**
  - Deterministic lower bound
  - Relative/absolute gap similar to MIP

    è Proven global optimum

BARON vs. LINDOGlobal

- Both find proven global optimum

- BARON
  - Requires finite bounds for all variables and nonlinear expressions to guarantee global optimum
  - Can return the *k* best solutions
  - Not tied to one sub solver

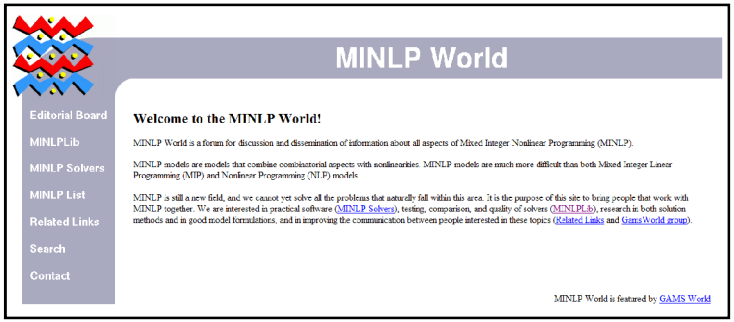
- LINDOGlobal
  - Copes with trigonometric functions
  - Handles non-smooth functions directly

Agenda

- Global Optimization Introduction
- GAMS Global Solvers
- **Performance & Quality**
- Linking a new Code

MINLP Library

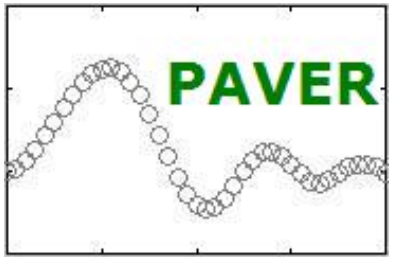
- Collection of Mixed Integer Nonlinear Programming models (~250 models)
- Large and varied set of both theoretical and practical models
- Helps algorithm developers to test their source



www.gamsworld.org/minlp/minlplib.htm

PAVER

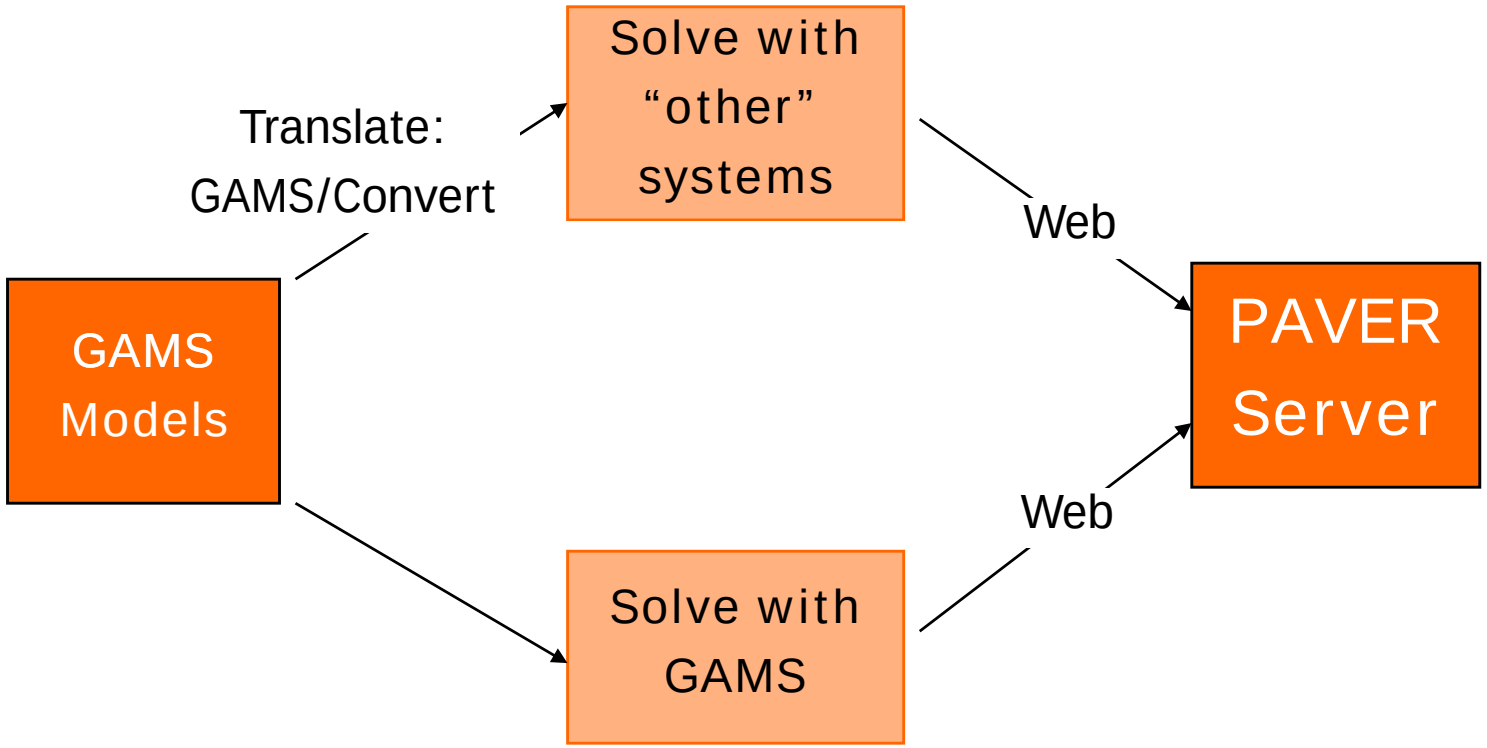
- **P**erformance **A**nalysis and **V**isualization for **E**ffortless **R**eproducibility



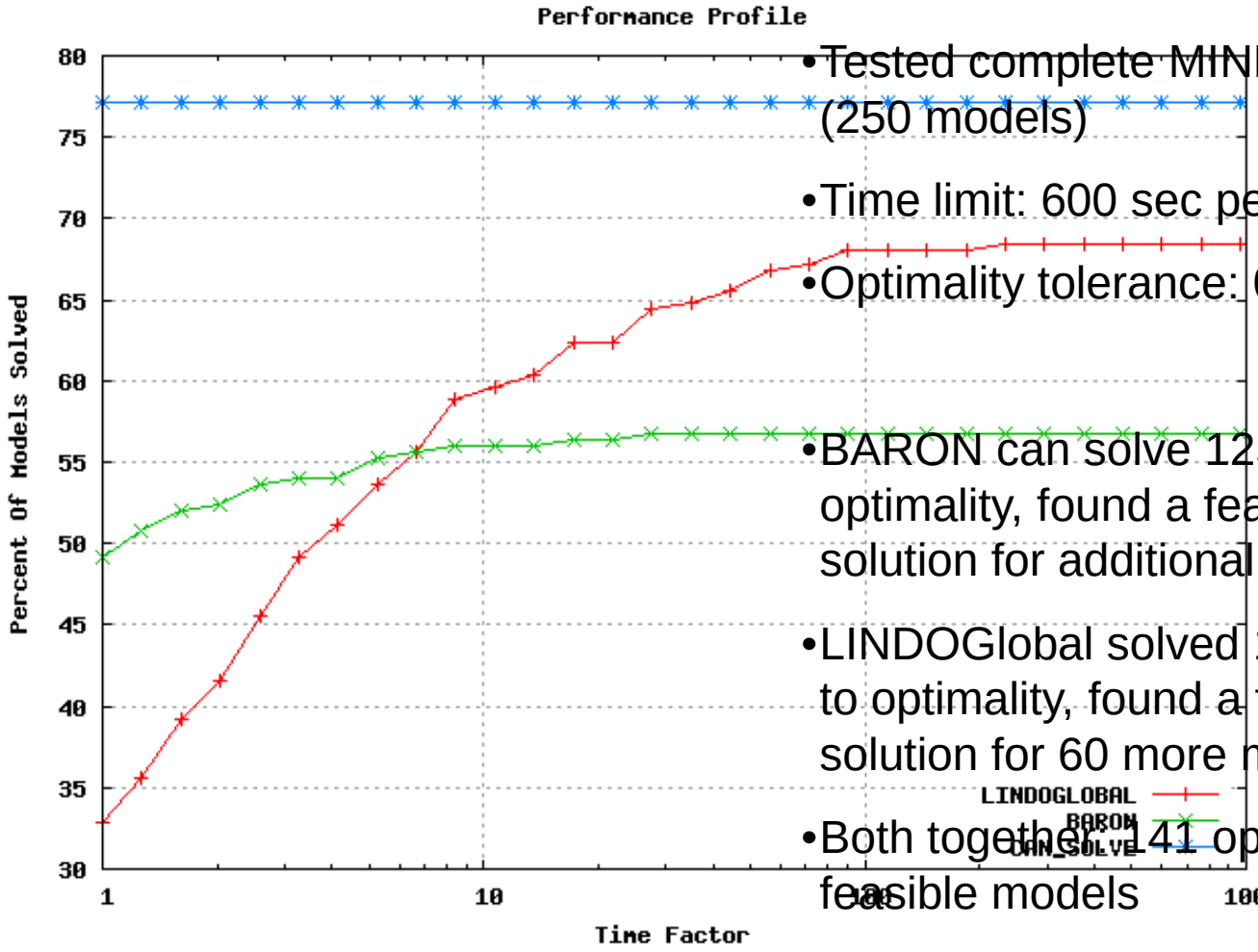
- Online server to facilitate performance testing and analysis/visualization
- Results sent via e-mail in HTML format
  - **System independent**

www.gamsworld.org/performance/paver

Open Testing Architecture



PAVER: LINDOGlobal vs. BARON



- Tested complete MINLP Library (250 models)
- Time limit: 600 sec per model
- Optimality tolerance: 0
- BARON can solve 125 models to optimality, found a feasible solution for additional 40 model
- LINDOGlobal solved 131 models to optimality, found a feasible solution for 60 more models
- Both together: 141 optimal and 53 feasible models

Benchmarking pitfalls

- Solvers may contain bugs – really!
  - Wrong solution returned
  - Wrong objective returned
  - False claims of feasibility/optimality
- Solvers will use different termination checks/tolerances
  - Difficult to compare “quality” of solutions
  - Common standard of comparison is lacking
- PAVER does not check validity of input data

GAMS/Examiner

- Purpose: to make an unbiased, independent report on the merit of points
- Points may come from GAMS or a solver
  - GAMS passes the previous solution as initial iterate
  - Solvers pass solutions back to GAMS
- Useful during solver debugging – helps pinpoint problems
  - Most checks are obvious
- Does checks on the scaled and unscaled (original) model
- All solution tolerances can be adjusted, default is tight
- Different points can be checked
- Examiner only *reports*, it doesn't *fix*

GAMS/Bench

- Tool to help facilitate benchmarking of GAMS solvers
- Compares resource usage of solvers selected by the user
- Creates problem matrix once and spawns it to all solvers
- Can create trace files used by PAVER
- Can call GAMS/Examiner for every tested solver
- Comes free with every GAMS system (depends on licensed solvers)

GAMS/Bench Output

```
Primal constraints satisfied (tol = 1e-006)
alan,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.158,0,8,9,4,24,3,NA,1,1,2.9250000000001,2.925,0.078,167,0,1,#
batch,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.158,0,74,47,24,191,22,NA,1,1,285506.510639964,285506.510639948,22.828,128068,0,26,#
batchdes,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.158,0,20,20,9,53,10,NA,1,1,167427.6571147,167427.6571147,0.234,1180,0,4,#
beuster,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.158,0,115,158,52,398,159,NA,8,3,128002.432361333,13002.3195417608,601,2904299,0,378,#
contvar,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.165,0,285,297,87,1281,530,NA,8,3,813303.696565387,392084.341377606,602.187,675054,0,3,#
csched1,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.172,0,23,77,63,174,8,NA,1,1,-30639.2578450033,-30639.2578557346,26.922,114747,0,145,#
deb10,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.173,0,130,183,11,692,432,NA,1,1,209.427811754669,209.427811754669,154.375,26218665,0,1,#
dosemin2d,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.175,0,119,166,32,4379,4080,NA,14,6,NA,NA,NA,NA,NA,NA,#
du-opt5,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.175,0,10,21,12,47,20,NA,1,1,8.0736575822132,8.0736575822132,61.296,5046,0,1,#
du-opt,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.175,0,10,21,13,47,20,NA,1,1,3.55634005788148,3.55634005788148,110.656,9075,0,1,#
eg_all_s,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.177,0,28,8,7,220,196,NA,13,3,NA,NA,601.032,0,0,0,#
eg_disc_s,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.184,0,28,8,4,220,196,NA,13,3,NA,NA,601.031,0,0,0,#
eg_disc2_s,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.19,0,28,8,3,220,196,NA,13,3,NA,NA,603.313,0,0,0,#
eg_int_s,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.197,0,28,8,3,220,196,NA,13,3,NA,NA,601.875,0,0,0,#
elf,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.204,0,39,55,24,178,30,NA,1,1,0.191666664686662,0.191666665776661,49.422,172820,0,54,#
eniplac,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.205,0,190,142,24,511,48,NA,8,3,-131806.374611992,-168871.02357847,601,6313823,0,7,#
enpro48,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.212,0,215,154,92,742,29,NA,1,1,187277.259422014,187277.259422014,52.938,244931,0,1,#
enpro56,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.213,0,192,128,73,651,24,NA,8,3,322067.383696514,263140.351531923,601,3810986,0,74,#
ex1221,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,6,6,3,17,2,NA,1,1,7.66718006881327,7.66718006881313,0.031,38,0,1,#
ex1222,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,4,4,1,9,2,NA,1,1,1.07654308333225,1.07654308333225,0.016,34,0,1,#
ex1223,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,14,12,4,40,17,NA,1,1,4.5795824024366,4.57958240243657,0.578,1953,0,2,#
ex1223a,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,10,8,4,32,9,NA,1,1,4.57958240243672,4.5795824024367,0.031,157,0,2,#
ex1223b,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,10,8,4,32,17,NA,1,1,4.57958240243672,4.57958240237498,0.125,711,0,2,#
ex1224,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,8,12,8,31,6,NA,1,1,-0.943470500622735,-0.943470500522735,0.359,3410,0,3,#
ex1225,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,11,9,6,27,2,NA,1,1,31,31,0.141,279,0,1,#
ex1226,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,6,6,3,15,2,NA,1,1,-17,-17,0.031,91,0,2,#
ex1233,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,65,53,12,221,28,NA,1,1,155010.671278216,155010.671278216,61.297,255059,0,100,#
ex1243,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.22,0,97,69,16,329,36,NA,1,1,83402.5064102569,83402.506410257,32.891,131465,0,9,#
ex1244,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.221,0,130,96,23,469,52,NA,1,1,82042.9052197479,82042.9052197471,301.61,710319,0,88,#
ex1252,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.224,0,44,40,15,118,36,NA,1,1,128893.741013226,128893.741010754,17.813,194740,0,325,#
ex1252a,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.224,0,35,25,9,94,36,NA,1,1,128893.741013318,128893.741013318,117.5,531460,0,676,#
ex1263,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.226,0,56,93,72,241,32,NA,1,1,19.6,19.6,31.422,321060,0,110,#
ex1263a,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.226,0,36,25,24,153,32,NA,1,1,19.6,19.6,6.938,76749,0,1,#
ex1264,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.226,0,56,89,68,237,32,NA,1,1,8.60000000000002,8.6,93.375,292976,0,31,#
ex1264a,MINLP,LINDOGLOBAL,CONOPT,CPLEX,39231.227,0,36,25,24,153,32,NA,1,1,8.60000000000002,8.6,1,4035,0,1,#
--- Job ex1222.gms Stop 08/30/07 06:23:04 elapsed 0:00:01.250
```

Agenda

- Global Optimization Introduction
- GAMS Global Solvers
- Performance & Quality
- Linking a new Code

Solver Links

- Standardized Solver Interface
  - Return Codes, Limits, Interrupts, …
  - Common attributes (e.g. time) through GAMS options
  - Specific options through option file

  à allows “hassle free” replacement of solvers:
  `option minlp=lindoglobal;`

- Open architecture assures seamless communication
  - IO Library (C, Fortran, Delphi) provides access to Matrix, Function/Derivative Evaluator, …

MISQP* Link

```
// EVALUATION OF PROBLEM FUNCTIONS
with mymodel
do begin
   for i := 1 to m
   do begin
      if nlfstart^[i] <> 0
      then
         neweval(nlfstart^[i],nlfend^[i],codefun^,p
      else
         equnonlinear := 0.0;
      equlinear := 0.0;
      jac := equfirst^[i];
      while jac <> nil
      do begin
         if not jac^.NL and ((objreform = 0) or (ja
         then
            equlinear := equlinear + X[VInd[jac^.co
         jac := jac^.nextrow;
         end;
      if (objreform = 1) and (i = objrow)
      then
         begin
         if direction = 0
         then
            F := (equlinear + equnonlinear - rhs[i]
         else
            F := -(equlinear + equnonlinear - rhs[i
         end;
      if (objreform = 0) or (i <> objrow)
      then
         G[EInd[i]] := equlinear + equnonlinear - r
      if ((equtype[i] = 2) and (direction = 0)) or
      then
         G[EInd[i]] := -G[EInd[i]];
      end;
   if objreform=0
   then
      begin
      if direction = 0
      then
         F:= X[VInd[objvar]]
      else
         F:= -X[VInd[objvar]];
      end;
   end;
end;
```

```
// EVALUATION OF GRADIENTS
with mymodel
do begin
   for i := 1 to m
   do begin
      if nlfstart^[i] <> 0
      then
         neweval(nlstart^[i],nlend^[i],code^,pool^,XVal^,
      jac := equfirst^[i];
      while jac <> nil
      do begin
         if jac^.NL
         then
            curgrad := evalgrad[jac^.col]
         else
            curgrad := jac^.coef;
         if ((equtype[i] = 2) and (direction = 0)) or ((e
         then
            curgrad := -curgrad;
         if (objreform = 1) and (i = objrow)
         then
            begin
            if jac^.col <> objvar
            then
               DF[VInd[jac^.col]] := curgrad / (-objcoef)
            end
         else
            begin
            if (objreform = 0) or (jac^.col <> objvar)
            then
               DG[dg2(EInd[i],VInd[jac^.col])] := curgrad
            end;
         jac := jac^.nextrow;
         end;
      if objreform = 0
      then
         begin
         if direction = 0
         then
            DF[VInd[objvar]] := objcoef
         else
            DF[VInd[objvar]] := -objcoef;
         end;
      end;
   end;
end;
```

```
// OPTIMIZATION BLOCK : CALL MISQP IN REVERSE COMMUNICATION
repeat
   if (IFAIL = 0) or (IFAIL = -1)
   then
      FUNC(F,G,X);

   if (IFAIL = 0) or (IFAIL = -2)
   then
      GRAD(DF,DG,X);

   MISQP(ENum,
         ME,
         MMAX,
         VNum,
         NINT,
         NBIN,
         NMAX,
         MNN,
         X[0],
         F,
         G[0],
         DF[0],
         DG[0],
         U[0],
         XL[0],
         XU[0],
         B[0],
         OpACC,
         OpACCQP,
         OpMAXIT,
         OpMAXPEN,
         OpMAXUND,
         OpRESOPT,
         OpNONMON,
         OpMAXNDE,
         OpIPRINT,
         OpMODE,
         OpIOUT,
         IFAIL,
         RW[0],
         LRW,
         IW[0],
         LIW,
         LW[0],
         LLW);
until IFAIL >= 0;
```


*Prof. Schittkowski, University of Bayreuth

BARON Input

```
MODULE: NLP;
VARIABLES  x2,x3,x4,x5,x6;
EQUATIONS  e2,e3,e4;

e2: x3^2 + x4^3 + x2 == 6.24264068711929;
e3:  - x4^2 + x3 + x5 == 0.82842712474619;
e4: x2*x6 == 2;

OBJ: minimize  (-1 + x2)^2 + (x2 - x3)^2 + (x3 - x4)^3
               + (x4 - x5)^4 + (x5 - x6)^4;

STARTING_POINT{
x2: -1;
x3: 2;
x4: 1;
x5: -2;
x6: -2;
}
```

LINDOGlobal Input

```
BEGINMODEL     TEST
VARIABLES
  x2    -1E30        -1     1E30   C
  x3    -1E30         2     1E30   C
  x4    -1E30         1     1E30   C
  x5    -1E30        -2     1E30   C
  x6    -1E30        -2     1E30   C
OBJECTIVES
  OBJ             MINIMIZE
    EP_PUSH_NUM   -1
    EP_PUSH_VAR   x2
    EP_PLUS
    EP_SQR
    EP_PUSH_VAR   x2
    EP_PUSH_VAR   x3
    EP_MINUS
    EP_SQR
    EP_PLUS
    EP_PUSH_VAR   x3
  …
```

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