Contents

Chapter 1

Language Features

1.1 Put file question

Suppose I have a model that I want to run t = 1 ... T times, where on each model run a price increases by a fixed increment. Further suppose there are 2 output files that have data sent to them via a put statement eg. land.out and water.out for each. I want to set up a looping procedure where my model is within the loop and my price increases by a fixed constant within each loop. Is there any syntax that would allow me to include the value of,t, in the name of output files. For example if T = 2 I would end up with land1.out water1.out land2.out water2.out
as my put files?
Answer from rutherford@colorado.edu:

```
set     i /1*4/;

file    ktmp  /temp.out/;  ktmp.lw=0;
file    kcopy /fcopy.bat/; kcopy.lw=0;

loop(i,
        put ktmp, 'Results for A in iteration ',i.tl;
        putclose;
        putclose kcopy, 'copy temp.out A.',i.tl,' >nul'/;
        execute 'fcopy.bat >nul';

        put ktmp, 'Results for B in iteration ',i.tl;
        putclose;
        putclose kcopy, 'copy temp.out B.',i.tl,' >nul'/;
        execute 'fcopy.bat >nul';
);
```

1.2 How to transform a parameter

If i have a parameter p(k) /50, 75, 80/ where k=1,..,3
How can I use GAMS to create a parameter p(?) /50, 75, 80, 80, 80 ,80 ,80 ,80.....80/ which is of dimenion i=1,...,N so that I can multiply this parameter by X(i) in an equation? I don't want to have to write out the values of p(?) by hand.
Answer from adrud@arki.dk:

```
Try
set L all indices for P / i1*i50/;
set I(L) indices with data / i1*i3/;
```

```
parameter p(L) / i1 50, i2 75, i3 80 /;
Loop(L$(not I(L)), P(L) = P(L-1); );

If you do not want to define the set I and if you are sure the initial data
only appear in the first position of P then you may try

Loop(L$(not P(L)), P(L) = P(L-1); );

which simply says that if P does not have a value, take the previous.
```

Answer from :

```
The following using the gams sameas command should work
parameter pnew(i)  new p values;
set  mapping(i,k)  tells which i's to associate with which k's
;
mapping(i,k)$(sameas(i,k) or ord(i) gt card(k))=yes;
pnew(i)=sum(mapping(i,k),p(k));
```

1.3 Endogenous variable becomes exogenous

Under the assumption that I want to run only one .gms file, consider the following:
First step: A variable X is endogenous. I run the model to get the optimal value for X.
Second step: The optimal value of X is exogenous in order to derive the optimal value for a variable Y.
Answer from adrud@aki.dk:

```
After you have determined the value of X, for example using a SOLVE
statement,
you add the line

X.FX = X.L;

and X if now fixed at the solution value. If you want to free it again,
you must change the lower and upper bounds bact to their original values,
for example

X.LO = -INF; X.UP = +INF;
```

1.4 Indexing a variable with subsets

```
Why can't I index a variable with a subset of it's index set? An example:

set V /1*20/
W /1,5,9,13,17/;
parameter A(V);

Now I would like to be able to use A(W) in an equation
```

Answer from n.n:

```
Have a look a the example below:
$eolcom !
set v / 1*20 /,
w(v) subset of V / 1,5,7/;
```

```
variable a(v) ! a is indexed with DOMAIN v
parameter b(v)
equation c(v)
```

V is used as a domain and cannot be changed any more. But one can use a(w) or b(w) and gams can ensure that you will not violate the domain because w is a subset of V. You can even define an equation over a dynamic set.

```
    c(w).. ....
```

Note that c was declared c(v) but the equation is defined with c(w). You can change w as you like (make it empty) and when you solve a model that contains c, it will only generate the equation relevant to w.

1.5 Using Subsets

I have a set of regions, and want to compactly define a subset and that subset's complement in the set. The following works, but IF I change DOERS by deleting B, I need to remember to add B to NOERS, what I always forget. Is there a way to set this up so one change will do it?

```
SETS REG All regions /A, B, C, D/,
DOERS(REG) Those who do /A, B/,
NOERS(REG) Those who do not /C, D/ ;
```

Answer from n.n:

Try something like this:

```
SETS REG All regions /A,B,C,D/
DOERS(REG) /A,B/
NOERS(REG);
NOERS(REG)$(NOT DOERS(REG))=YES;
```

Thus, NOERS is everything in REG that isn't a member of DOERS.

Answer from n.n:

This problem is commonly encountered. The key idea here is to use a dynamic set if that is possible.Here is a solution:

```
set r a big set /a,b,c,d/
s(r) a subset of r /a,b/
c(r) the complement of s -- a dynamic set;
c(r) = yes$(not s(r));
```

The rub here is that because c is dynamic, you may not subsequently declare a parameter using c in the domain -- you need to use set r.

1.6 Using Aliases

I am trying to create the following constraints in gams.

```
x11 =g= .7(x11 + x21)
x21 =g= .25(x11 + x21)
x12 =g= .3(x12 + x22)
x22 =g= .6(x12 + x22)
```

Naturally, I'd like to say

```
MYEQN(I,J).. X(I,J) =G= PARAM1(I,J)*SUM(I, X(I,J));
```

where param1 is a table of the decimal values listed above. This, of course, leads to the error statement "set already under control". How do I get around this without having to create an equality constraint and a variable for each j that would look like

```
MYEQN2(J).. WHY(J) =E= SUM(I, X(I,J));
```

so that I could use the WHY(J) in place of the sum in MYEQN?

Answer from n.n:

This error does not occur if you do it this way:

```
SET I /1*2/;ALIAS (I,J,K);
VARIABLES X(I,J),Z;
TABLE PARAM1(I,J)
    1       2
1   0.7     0.3
2   0.25    0.6
EQUATIONS DUMMY, MYEQN(I,J);
MYEQN(I,J).. X(I,J) =G= PARAM1(I,J)*SUM(K,X(K,J));
DUMMY.. Z =E= 1;
MODEL M /ALL/;
SOLVE M USING LP MINIMIZING Z;
```

1.7 Sets in the LOOP construct

When you're using an index set to define a loop, is there any way to define a subset of the index set which contains only the last element of the set? Something that would correspond to the current value of the iteration index in a FORTRAN do-loop? As far as I can see, when GAMS executes the k-th iteration of the loop, the controlling index set contains the first k elements of the set. Here's why I need just the last element. Suppose you have a table x(i,j), where i runs from 1 to 3, and j from 1 to 2. I want to generate equations in a GAMS model, two per iteration, such that at iteration k I get

```
LAM * x(k,j) =l= sum(i, z(i) * x(i,j)), for all j.
```

(where LAM and the zs are GAMS variables).For example when k = 2 you want

```
LAM * x(2,1) =l= sum(i, z(i) * x(i,1))
LAM * x(2,2) =l= sum(i, z(i) * z(i,2))
```

Here's a toy GAMS program which tries to do this. Note that the LP is unbounded: the only reason to compile it is to generate the equations.

```
$TITLE LP test problem
$eolcom;
```

```
$OFFUPPER
OPTIONS LIMCOL = 100, LIMROW = 100;
set i / 1, 2, 3 / ;
set j / 1, 2 / ;
alias(index,i) ;
set k(index) /2/;
set k(index);
table x(i,j) data initialization
    1   2
1   10  11
2   21  22
3   31  32

variables lam, z(i) ;
equation xeq(i,j) adding-up constraint ;
xeq(k,j) .. lam * x(k,j) =l= sum(i, z(i) * x(i,j)) ;
model lptest /all/ ;
; first try to solve the model using the value
; of k established above, i.e. 2.
solve lptest using lp minimizing lam ;
; now try to do it in the course of a loop.
loop(index,
    k(index)=yes;
    solve lptest using lp minimizing lam;
     );
```

The first SOLVE (the one outside the loop) works fine, and generates the equations listed above. But the looping attempt fails. When you get to iteration k = 2, the index set has values 1 and 2 so I generate not two, but four equations. (and 6 in iteration 3). I've spent I don't know how much time trying to generate just the two equations each iteration, but I just can't do it.

Answer from adrud@arki.dk:

You have a problem with a set in a loop, where the loop looks like:

```
loop(index,
    k(index)=yes;
* do something
    );
```

In this loop, the set k changes from each time the loop is executed, and the difference is that one extra element is added each time. Try:

```
loop(index,
    k(index)=yes;
    display k;
    );
```

to see this. If you want k to contain only the current element from index, then you must "clean up" during each loop, for example like

```
loop(index,
    k(k) = no;
    k(index)=yes;
    display k;
    );
```

The line k(k) = no; will erase the content of k no matter what was there before. A small problem can occur if k has not been initialized; in this case you can use the construct

```
loop(index,
    k(index)=yes;
    display k;
    k(index)=no;
    );
```

1.8 Lag and lead circular

The following code is used to get a value for the number of two types of employees who are currently working a shift. one type works eight hour shifts and of course, the other four hours. Is there a way to do this without "spelling it out"? If so, please tell me what it is.

```
PTONSHIFT(S).. PTOS(S) =E= PTH(S) + PTH(S--1) + PTH(S--2) + PTH(S--3);
FTONSHIFT(S).. FTOS(S) =E= FTH(S) + FTH(S--1) + FTH(S--2) + FTH(S--3)
               +FTH(S--4) + FTH(S--5) + FTH(S--6) +FTH(S--7);
```

Answer from n.n:

Try this:

```
* partimers work 4 hour shifts, fulltimers 8, Rangers never sleep;
SETS PHOURS /1*3/, FHOURS /1*7/;
PTONSHIFT(S).. PTOS(S) =E= PTH(S) + SUM(PHOURS,PTH(S--ord(PHOURS)));
FTONSHIFT(S).. FTOS(S) =E= FTH(S) + SUM(FHOURS,FTH(S--ord(FHOURS)));
```

Answer from n.n:

Try the following:

```
ALIAS (alt_s, s);
PTONSHIFT(S).. PTOS(S) =e= SUM(alt_s$((ord(alt_s) ge (ord(s) - 3))
        and (ord(alt_s) le ord(s))), pth(alt_s));
FTONSHIFT(S).. PTOS(S) =e= SUM(alt_s$((ord(alt_s) ge (ord(s) - 7))
        and (ord(alt_s) le ord(s))), pth(alt_s));
```

1.9 Creating a subset to use as an index

I am having trouble creating a subset to use as an index in an equation. I have created a set AT /0*30/ which represents tree age in years. Later in my program I would like to calculate the expected net revenue(per acre) from trees aged AT in time T over the trees remaining lifetime(up to 30years). Thus I would like to sum over all of the tree ages greater than or equal to the current tree age in T. For instance:

```
ENR(3,T) = EPTC*YLD(3)-PC(3) + EPTC*YLD(4)-PC(4) + ......
ENR(5,T) = EPTC*YLD(5)-PC(5) + EPTC*YLD(6)-PC(6) + ...
```

My problem is what to use as the index for the summation in the generalized

```
equations ENR(AT,T).. NR(AT,T)=E=SUM(?,(EPTC*YLD(AT)-PC(AT));
```

Answer from n.n:

Have a look on this example:

```
SET T Time period /1990*2040/,
A Tree age /0*15/;
ALIAS (T,TT), (A,AA);
PARAMETER
ENR(A,T) Expected net revenue of tree age A in year T
P(T) Present value price for year T,
YIELD(A) Yield for age A,
YEAR(T) Numeric value of year T
AGE(A) Numeric value of age A;
YEAR(T) = 1990 + ORD(T) - 1;
AGE(A) = ORD(A) - 1;
* 3% interest rate, and a 7% depreciation rate on yield:
P(T) = EXP( -0.03 * (ORD(T)-1));
YIELD(A) = EXP( -0.07 * (ORD(A)-1));
ENR(A,T) = SUM((AA,TT)$( (AGE(AA) GE AGE(A)) *
(YEAR(TT)-YEAR(T) EQ AGE(AA)-AGE(A)) ),P(TT) * YIELD(AA) );
DISPLAY P, YIELD, ENR;
```

Answer from n.n:

I had some time ago the same problem, the solution was application of exceptional handling $ operator. The following example worked and should solve your problem. I do not know the matter of your model, but I think that there are three possible forms for the equation you mentioned.

```
set AT/0*30/,T/0*60/;
alias (AT,ATA);
scalar EPTC /0.1/;
parameter YLD(AT), PC(AT), NR(AT,T);
* Data for test
YLD(AT)=20;
PC(AT)=1;
* The difference in following forms is in the indexes
* Formula 1,
NR(AT,T)=SUM( ATA$(ORD(ATA) GE ORD(AT)), EPTC*YLD(ATA)-PC(ATA));
DISPLAY NR;
* Formula 2
NR(AT,T)=SUM( ATA$(ORD(ATA) GE ORD(T)), EPTC*YLD(ATA)-PC(ATA));
DISPLAY NR;
* Formula 3 difference in the second index of NR matrix
* To use it you need to change NR definition to NR(AT,AT)
*NR(ATA,AT)=
* SUM( ATA$(ORD(ATA) GE ORD(AT)), EPTC*YLD(ATA)-PC(ATA));
*DISPLAY NR;
equation ENR(AT,T);
```

```
* The equation - formula 1, note that the right side has no
* relation to T and the equation will be the same for all T
ENR(AT,T).. NR(AT,T) =E=SUM(ATA$(ORD(ATA) GE ORD(AT)), EPTC*YLD(ATA)-PC(ATA) );
* The equation - formula 2
* ENR(AT,T).. NR(AT,T) =E=
* SUM(ATA$(ORD(ATA) GE ORD(T)), EPTC*YLD(ATA)-PC(ATA) );
* The equation - formula 3
ENR(ATA,AT).. NR(ATA,AT) =E=SUM(ATA$(ORD(ATA) GE ORD(AT)), EPTC*YLD(ATA)-PC(ATA) );
```

1.10 Index for the maximum value of a set

I have a problem of choosing indexes for the maximum value of a set.

```
Set i /1*5/, j /1*10/
```

a(i,j) is a set of value on i and j. Let M = smax((i,j), a(i,j)) . How do I
get the index for i and j which lead to value M?

I tried to use the following approach, and it failed.

```
loop((i,j),
    index = i.tl$(M eq a(i,j));
    jindex = j.tl$(M eq a(i,j));
    );
```

Answer from rrosenthal@monterey.nps.navy.mil:

Try this:

```
$inlinecom { }
{GAMS does not have an argmax function, which is
requested here. Here is how to do the
equivalent with a dynamic set that I'll call maximizer.}
sets i /i1 * i5/, j /j1 * j4/, maximizer(i,j);
{Notice I changed Yan's labels, because I don't
like to see labels mistaken for numbers.}
table a(i,j)
    j1  j2  j3  j4
i1  12  34  10  7
i2  33  16  5   18
i3  17  22  12  31
i4  24  1   29  29
i5  14  13  34  -9
;
scalar maxa ;
maxa = smax( (i,j), a(i,j) ) ;
maximizer(i,j) = yes$( a(i,j) eq maxa ) ;
display maximizer ;
{You can use this dynamic set as index, in calculations and equation
definitions (but not, yet, in declarations.}
```

1.11 Exception handling on indexes

Here I am having a small problem. I will explain though a trivial example.

```
SETS I /1*20/;
PARAMETERS A(I);
A('1') = 100 ;
```

Remaining A(I)'s should be just at an increment of 5

```
A(I$(I NE 1) = A(I) + 5;
```

This does not work.

Answer from n.n:

The condition that you need to use is:

```
(ORD(I) NE 1)
```

i.e., "if this is not the first element in the index set I". You can als use:

```
SETS I /1*20/;
PARAMETERS A(I);
A('1') = 100 ;
* Remaining A(I)'s should be just at an increment of 5
loop(i$(not sameas(i,"1")),A(I) = a(i-1) +5) ;
display a;
```

1.12 Introducing another dimension

Consider the following table of feedstuffs(f) and nutrient contents(nu)

```
table ingred(f,nu)
            energy       protein    fibre     drymatter
ms          180          2.5          5         -28
gs          180          4.8          9         -35
cl          128          3.8        4.4         -21
ha          430          8.6         26         -86
```

In a regional model the nutrient contents are initially set equal in all subregions. Therefore I don't include the 3rd dimension(r) yet. In a later stage of the model I need to change the above table, ie the quality of the feedstuffs in some municipalities. Using a $condition to indicate the regions for which the values of the table are to be changed I need to come up with a third dimension (r). When I use this command

```
table newingr(f,nu);
newingr(f,nu)$(sum(iff, perdiff2(iff,r)) GE 0) = ingred(f,nu) * .9;
                                      $149
display newingr;
```

I get an error message (uncontrolled set entered as constant). I know that it is not correct but I do not know how handle this problem. If someone could help me I would appreciate that very much.

Answer from rutherford@colorado.edu:

This is far and away the most common error by new GAMS programmers. The reason GAMS rejects the statement is that you refer to an uncontrolled set inside the condition. It is like writing:

```
x(i) = y(i)  when k = 1
```

Notice that there is no reference to k on the left-hand side of this equation. It still might make sense if k were an index controlled by the algorithm, but this would be akin to having your assignment statement shown above inside a loop. Although the code you provided was somewhat sparse, I believe that what you wanted to write was:

```
parameter newingr(f,nu,r);
newingr(f,nu,r) = ingred(f,nu);
newingr(f,nu,r)$(sum(iff,perdiff2(iff,r)) ge 0) = ingred(f,nu) * .9;
```

(This is only a guess.)I hope that you see the problem now. Set r appears in the conditional expression, but it is not controlled by the indices of the assignment, nor (I assume) does it appear in a loop running over the statement. I must admit that this seems to be a difficult concept to teach to students, even those who have had all sorts of mathematics courses. The problem is that in a math course you can get away with the occasional non-sensical assignment statement and still get an A on the paper. When you are writing computer code, just one inconsistent statement stops you cold. The need for precision comes as a shock to many graduate students.

1.13 $ON/OFFEMPTY error

In my model I have a SET that can be empty. I received a message like this:

```
  67                / /;
****                  $460
460  Empty data statements not allowed. You may want to use $ON/OFFEMPTY
```

Where should I put that $ option?

Answer from rutherford@colorado.edu:

Here is an example of how the $onempty command can be used:

```
set j /1*3/;
$onempty
set i(j) / /;
parameter a(j);
a(j)$(not i(j)) = 1;
display a;
```

You can insert $offempty to have the compiler revert to the default syntax in which empty data fields generate a compile-time error.

1.14 What is the difference between the **-operator and the power function?

What is the difference between the **-operator and the power function?

Answer from n.n:

If you use the power function the exponent must be an integer, which is not required if you uses the ** - operator. However, with the ** operator the exponent must be a positive number to avoid an compilation error. So:

```
scalar test;
test = (-3)**2;
* note: this is not the same as -3**2,which
* will be treated as -(3**2)
display test;
```

will give you an error:

```
**** EXECUTION ERROR 10 AT LINE 4 .. ILLEGAL ARGUMENTS IN ** OPERATION
```

This formulation will work:

```
scalar test;
test = power(-3,2);
*note: here we could also use sqr(-3)
display test;
```

1.15 Very large numbers

I am having a problem with a simple computation. I am trying to raise 9211 to the power 8.752. This looks pretty simple - too simple even for a pocket calculator. But I get the error message "overflow in ** operation" when I type the following:

```
parameter at ;
at = 9211**8.752 ;
```

The result (from a pocket calculator) is 4.9614995 E+34. What could be wrong?

Answer from adrud@arki.dk:

In order to run the same way on all kinds of hardware GAMS has a limit on the size of numbers around Everything larger than that will be translated into UNDEF. If you have intermediate results like this inside a model, try to scale it. I assume that 9211 is the value of a variable; scale it by 1000 or even better with 10000 so you will get 9.211**8.752 = 2.75e8 or 0.9211**8.852 = 0.487.
You can use VAR.SCALE(..) = 10000; in your GAMS program.

Answer from rrosenthal@monterey.nps.navy.mil:

You give new meaning to what I used to think of as "huge models." I thought only the astronomers had to deal with such big numbers. Remember, computers have only finite precision, so you are flirting with disaster when you try to do computations with orders of magnitude this high. For example, in the next step of your program, suppose you needed to compute at+1, and then say you needed to divide by ((at+1)-at). Try computing ((at+1)-at) on your pocket calculator; I'll bet you get 0 not 1. You can see it's not easy to play around with the 35th decimal place. My suggestion is to rescale the units, so that you avoid the problem. Try to make the first six or so decimal places significant for you. By the way, if computing a number over 1E+34 causes trouble for a GAMS assignment statement, we can hardly imagine how much worse things could have gotten if you had reached the solver.

Answer from n.n:

GAMS can handle numbers in the range between 10 E-30 10 and 10 E +29.

1.16 Using the determinant

I have a matrix A(I,J) whose entries aij are functions of unknown but bounded variables X(K). That is: aij = F(X(k)). Each X(K) are bounded between 0 and 1. I want to MAXIMIZE the determinant of A(I,J). I do not have an explicit expression for the determinant of A(I,J) (my matrices may vary in size) but I have an algorithm to calculate the determinant of a matrix. This algorithm is given in the model library of GAMS and is called GAUSS.71. I would like to modify this algorithm to solve the problem of maximizing the determinant of A(I,J). This algorithm is written to be solved once, and I need the objective function to be the determinant. I cannot define equations inside the LOOP. Maybe there is a way to call this algorithm as part of the model. What do you think?

Answer from n.n:

Using the determinant directly as defined in GAUSS is difficult because of the permutations involved in the Gaussian elimination. They could give rise to binary variables with all the derived difficulties. It is therefore interesting to know if your matrix has any special structure, in particular if it is symmetric and possibly positive definite. For a positive definite symmetric matrix A you can use the following trick:

A = L * L' (where ' = transpose). L is the Cholesky factorization.
Det(A) = Det(L)**2 so you can maximize Det(L) = PROD(I, L(I,I)).

The relationship between A and L can be written as a set of equalities.

1.17 Indexing connundrum

I'd like to define a set that I could use for indexing and comparing. To wit, if I have a set:

```
 I  /1*26/;
and an alias(I,J);
I'd like to be able to do something like:

$(I LT J) instead of $(ord(I) LT ord(J)).
```

The model under scrutiny has 214 ords and 60 cards in it, hence the interest.

Answer from adrud@arki.dk:

You can create a two dimensional set with the legal combinations:

```
set ij(i,j) Legal i-j combinations;
ij(i,j) = yes$(ord(i) lt ord(j));
```

and then replace you conditions by

```
   $ij(i,j)
```

But you cannot avoid to use ord the first time. Set elements are text strings and can not be ordered (without implying some translation of characters into numbers).

1.18 How to transform a parameter

If i have a parameter p(k) /50, 75, 80/ where k=1,..,3

How can I use GAMS to create a parameter

p(?) /50, 75, 80, 80, 80 ,80 ,80 ,80.....80/ which is of dimenion i=1,...,N

so that I can multiply this parameter by X(i) in an equation?

I don't want to have to wtite out the values of p(?) by hand.

Answer from mccarl@masked.tamu.edu:

The following using the gams sameas command should work

```
parameter pnew(i)  new p values;
set  mapping(i,k)  tells which i's to associate with which k's
;
mapping(i,k)$(sameas(i,k) or ord(i) gt card(k))=yes;
pnew(i)=sum(mapping(i,k),p(k));
```

Answer from adrud@arki.dk:

Try

```
set L all indices for P / i1*i50/;
set I(L) indices with data / i1*i3/;
parameter p(L) / i1 50, i2 75, i3 80 /;
Loop(L$(not I(L)), P(L) = P(L-1); );
```

If you do not want to define the set I and if you are sure the initial data only appear in the first position of P then you may try

```
Loop(L$(not P(L)), P(L) = P(L-1); );
```

which simply says that if P does not have a value, take the previous.

1.19 Subsets and assignments

The toy example below tries to define a table with 2 indices and then make a parameter assignment to MUINC using just one of them. If I try to pick out the desired index via the subset declaration dbx(db) the assignment fails; while if I "parametrize" MUINC as in the commented-out lines, it works.

I think I can see in a general way what's going on --- there must be a distinction between a 1-element subset and a single label --- but can anyone tell me if there's a way to do this which does not require parametrizing quantities like MUINC?

```
======================== example ============================

$ONDOLLAR
$COMMENT ;

SETS
db distance blocks /db1,db2/
dbx(db) chosen distance block /db1/
zbeta demand model coeff name /cf/
  ;


TABLE BETA(ZBETA,DB) demand coefficients
             DB1       DB2
   cf      -112700    -200800
   ;



; This one fails
PARAMETER MUINC marginal utility of income;
MUINC=-beta("cf",dbx);

; this one works
; PARAMETER MUINC(dbx) marginal utility of income;
; MUINC(dbx)=-beta("cf",dbx);
```

Answer from adrud@arki.dk:

If you know a set only has one element then the standard trick is to sum = over the set, i.e.

```
MUNIC = sum(dbx, -beta("cf",dbx) );
```

Answer from mccarl@masked.tamu.edu:

The question is what are you saying algebraically

you are saying a scalar equals a vector

```
    x=y(i)
```

gams wond do this because y1 may equal 4 and y2 equal 5 so what should x equal 4 or 5 yo be valis you mist resolve the subscript

the one that worked says x(i) = y(i) so the subcripts can be matched up

Answer from oyvind.hoveid@nlh10.nlh.no:

The GAMS compilator tells you that set dbx is uncontrolled, so you have to control it:

```
MUINC = - SUM(dbx, beta("cf",dbx));
```

Answer from saddy@cba.ua.edu:

The real reason why the you see the difference is because of set control in assignments. In the two cases you define MUINC and make assignment as follows:

```
PARAMETER MUINC marginal utility of income;
MUINC=-beta("cf",dbx);

PARAMETER MUINC(dbx) marginal utility of income;
MUINC(dbx)=-beta("cf",dbx);
```

The second works because dbx is the controlling set in the assignment. In English, this reads as "marginal utility of income over the chosen distance block". Since beta is defined over zbeta and db, GAMS needs to know which of them is the controlling set for the assignment. To use the first specification, you have to change it to identify the specific element within set dbx as follows:

```
PARAMETER MUINC marginal utility of income;
MUINC=-beta("cf","db1");
```

1.20 Different variable types within one set

Is there a concise way to declare some of the decision variables within a class, say, binary and the rest of the class continuous? For example, can we achieve the following notional declarations and how (that are illegal in the form shown below)?

```
BINARY VARIABLES
VARCLASS(A)$(ORD(A) LT CARD(A)/2) the first half are binary;
POSITIVE VARIABLES
VARCLASS(A)$(ORD(A) GE CARD(A)/2) the rest are continuous;
```

Can we declare a class of variables to be continuous at some point in the code and redeclare them to be binary later?

Answer from rutherford@colorado.edu:

Try this:

```
positive variables x(a);
binary variables xb(a);
```

```
equation xblink(a);
xblink(a)$bin(a).. x(a) =e= xb(a);
```

This will add some dimensions, but it is probably of no consequence for the computational time. Notice that the only xb(a) variables which appear in the model will be those for which bin(a)=yes. You have declared xb over the entire set a, but the only variables which appear in the model will be the ones which appear in the generated xblink equations.

1.21 Representing parameters as fixed variables

It is sometimes convenient to represent GAMS parameters as a fixed variable, like:

```
SET   i ;
PARAMETER     parm1(i)  ;
VARIABLE    PARM(i);
PARM.FX(i) = parm1(i);
```

I suppose a little more memory is needed for models in terms of 'PARM' than those those in terms of 'parm1'. But is the difference significant (if this doubles the number of variables)? And will the model be more difficult to solve (NLP - CONOPT)?.

Answer from adrud@arki.dk:

The penalty one pays for using a fixed variable instead of a parameter depends on the nonlinear expressions the fixed variable is used in. The expressions may be much larger (and function evaluation much slower) using variables instead of parameters.You can have it both ways here. If you set set <modelname>.holdfixed = 1; GAMS will treat fixed variables as constants; they will not be passed on to the solver and the nonlinear expressions the solver evaluates will be the same as if you uses parameter parm1 instead of the variable parm. If you don't set holdfixed = 1, CONOPT can still remove these fixed variables from the model it solves during its presolve stage, and make simplifications to the nonlinear expressions.

1.22 Computing of cost components

Is there a way to calculate individual cost components of for an objective function in GAMS? I successfully set up a program in GAMS and obtain the optimal solutions (including values for decision variables and the objective funcation). The objective function has SIX cost components and I need to investigate the contribution of each to the total objective function value. How can I write a procedure/section in the original program so that when the program is solved, the values for each individual cost components in the objective function are also achieved?

Answer from n.n:

Write an equation defining each equation with a "slack" variable which absorbs the row activity. Then redefine the total cost function as the sum of its components- if it is separable. Otherwise keep the objective function as is. The GAMS report will report the values of each cost equation's slack value which will be the component you desire.

Answer from n.n:

The method of writing separate equations for each objective function component works, but I have found it to be burdensome on the solution algorithm sometimes if the objective components are nonlinear (hence the new constraints are nonlinear). If you only want to know the components AFTER solution, then define a parameter for each component, and calculate the value of that parameter from the solution point variable values. A merit of this method is that the component values must only be calculated once as a parameter, but if they are introduced as variables they must (roughly) be calculated for every iteration of the solution. Of course, in the parameter calculation you have to refer to the variables with the "level" or ".L" suffix. Roughly speaking:

```
VARIABLE X(T);
OBJ(T) =E= C1(X(T)) + C1(X(T)) ... C6(X(T));
SOLVE MODEL BLAH MINIMIZING OBJ;
PARAMETERS CALCC1(T), CALCC2(T) ... CALCC6(T);
CALCC1(T) = C1(X.L(T));
..
DISPLAY CALCC1, CALCC2 ... CALCC6;
```

Answer from n.n:

There has been some discussion about the best way of computing intermediate terms in the objective function (nonlinear case), so they were immediately available for report purposes. The example was

```
ODEF.. OBJ =E= C1(...) + C2(...) ... CN(...);
```

where C1,..,CN are nonlinear expressions. The alternative formulation is of course to have the intermediate variable VC1,...VCN with defining equations

```
EC1 .. VC1 =E= C1(...);
...
ECN .. VCN =E= CN(...);
NEWODEF.. OBJ =E= VC1 + VC2 + ... + VCN;
```

The usefulness of this reformulation is, unfortunately, algorithm specific. MINOS does not like this formulation because the nonlinearities are moved from the objective to several constraints. CONOPT, on the other hand, does not mind since the extra variables and constraints as long as the intermediate variables, VC1,...,VCN are declared free. CONOPT essentially does the natural elimination, but you will still get the value of the intermediate variables back in the GAMS solution.

1.23 Divide by zero error in line..

I have written a simple equation that uses the .Lo data file as follows.

```
A.Lo(m,c) = sum( (k,r), (S(m,c) * P(k,r)*(-1)) / (Q(k,m)) );
```

where S is a matrix with positive numbers and zero elements, and where P is a negative number matrix, and Q has positive numbers for its elements. GAMS will load the model and this equation, but it will not solve, typing this note in the .lst file :

" 0 divide by zero error in line 445 "

445 is the line of this code fragment above. I was wondering if this equation is written wrong; that is, i am not sure i can combine all of these matrix in one equation.

Answer from n.n:

Since there is only one division, and you divide by a parameter then some of the elements of Q must be zero. The easiest in cases like this is to create a set of the 'bad' positions in Q, for example:

```
SET Zeros(k,m) Bad elements in Q;
Zeros(k,m) = YES$(Q(k,m) eq 0);
Display Zeros;
```

This is often better than displaying Q. If a whole row or column is missing you may not see so easily. Let GAMS do the work for you!

1.24 Interpretation of marginals

I am still unsure how to interpret the MARGINAL value for a constraint. Here I have set up a very simple optimization problem, hoping that the MARGINAL would have some relationship to the lagrange multiplier. It doesn't seem to. In the larger problem I am working on, I am concerned about isolating which constraints are binding and which are not. CONST2 is a binding constraint, and has a non- zero MARGINAL, but are these two facts coincidental, or related?

```
objective .. z =e= 30-20*x+10*(x**2);
const2 .. z =g= 15 + 10*x
;
model quadratic / objective /
model quad2 / objective, const2 /;

solve quadratic using nlp minimising z;
solve quad2 using nlp minimising z;
```

The unconstrained solution is z=20,x=1, and the constrained solution is :

```
**** OBJECTIVE VALUE 21.3397
```

where the equation stats show:

```
LOWER LEVEL UPPER MARGINAL
---- EQU OBJECTIVE 30.000 30.000 30.000 0.577
---- EQU CONST2 15.000 15.000 +INF 0.423LOWER LEVEL UPPER MARGINAL
---- VAR Z -INF 21.340 +INF .
---- VAR X -INF 0.634 +INF .
```

What is the interpretation of this 0.423, in a problem where the lagrange multiplier =0? When the same problem is run with an additional binding constraint z=30, the marginals take the following form:

```
LOWER LEVEL UPPER MARGINAL
---- EQU OBJECTIVE 30.000 30.000 30.000 -0.577
---- EQU CONST1 30.000 38.660 +INF .
---- EQU CONST2 15.000 15.000 +INF 1.577
```

Why does an additional binding constraint take a zero MARGINAL value?

Answer from n.n:

The Lagrange multiplier isn't zero. By using the same symbol in your objective and constraint, you have effectively set up the problem

```
min z
s/t z = 30 - 20x + 10x^2
z >= 15 + 10x,
```

which has two constraints. The Lagrange multipliers on these two constraints at optimality are .577 and .423, just as shown by GAMS. On the second question, there is no reason why the Lagrange multiplier on a binding constraint should be nonzero. It can be either zero or nonzero. (if the constraint is nonbinding, though, the multiplier must be zero)

1.25 GAMS-Solver Communication

Which information are passed to the solver if I do multiple solves?

Answer from adrud@arki.dk:

The GAMS manual mentions that GAMS will use the previous solution as much as possible. This may sound rather inaccurate; the background is the following: When a SOLVE statement is executed, i.e. when GAMS generates a model, it will use the current value and basis status of all variables and equations that appear in the model, and pass this information on to the solver. If the basis appear to be good measured by the number of good variables relative to the number of constraints, then GAMS will ask the solver to use the basis, and otherwise it should be ignored, see BRATIO in one of the appendices. If you design you model properly GAMS will automatically restart in the best way. However, there are some pitfalls. If you have a dynamic model with a variable X(T,I) and T is changed dynamically from one SOLVE to the next then GAMS cannot reused previous solution values; the value of X('1','ABC') may be known, but the model uses X('2','ABC'). If you remove the T index from the model GAMS may restart more intelligently and solution times may be reduced dramatically. Another example is in decomposition: You alter between model A and model B and both models depend on a variable, say X. When you solve B it will start from the optimal X value from A and when you solve A again it will start from the optimal X value from B. By proper choice of the names of variables you may transfer variables from one model to another or you may keep the models independent so that a second solve of a model not affected by an intermediate solve of another model.

1.26 How is the basis information passed to the solver?

Does anyone know just what information is passed to the solver from GAMS besides initial values (e.g. x.l) of variables? If other information is passed, can this information be "reset" in the event that multiple solves are called. I ask this question because my model has multiple solves and it seems that some of the latter solves are failing (ending with the message: THE PROBLEM IS UNBOUNDED (OR BADLY SCALED)) because of the state of the system left by earlier solves. I'm quite certain that the error message isn't due to the problems actually being unbounded.

Answer from n.n:

GAMS passes the solver information about the marginals (EQU.M), but (so far as I know) not the exact numerical values of the multipliers. The NLP code uses this information(together with the level values and bounds) to install the initial basis. In many cases, if you are solving a sequence of unrelated cases, you may wish to "recenter" the model before each solve. I usually do this with an include file:

```
loop(scenario,
     <assign parameter values for current scenario>
     $INCLUDE bench.gms ! set benchmark values
     solve model using ...
     <extract summary report information>
    ); ! end of loop over scenarios
```

Answer from erwin@gams.com:

In order to let a solver reconstruct a basis GAMS passes on primal values and dual "indicators". It is good enough to know whether or not a variable is basic (this corresponds to a zero marginal) or non-basic (nonzero marginal; the levels can then be used to find out whether a variable is nonbasic at lowerbound or at upperbound). For NLP's the story is a little bit more complicated due to superbasics. This indicator function of the marginals is one of the reasons why EPS is important: this signals a nonbasic with marginal that is numerically zero (a form of degeneracy). If you add variables and equations to the model between two solves the default for these new rows and columns is basic (marginal=0). When the number of zero marginals becomes too large it may be better for the solver to start from scratch and crash a basis, because the model has changed a lot. This is where the obscure BRATIO option plays its role: if the number of zero marginals becomes too large the model is supposed to have changed too much and the basis is rejected. (I think the manual is wrong here, where it says in appendix C: "The use of this basis is rejected if the number of basic variables is smaller than BRATIO times the size of the basis").This simple mechanism works wonderfully, and is portable over different algorithms, ie. a basis created by BDMLP can be used by MINOS etc.Setting BRATIO to 1 will cause the basis to be rejected, and BRATIO=0 will cause the basis to be accepted unconditionally. Another way of forcing the basis to be rejected is to set all (or most of the) marginals to zero (don't forget the marginals of the equations).

Answer from adrud@arki.dk:

The GAMS manual mentions that GAMS will use the previous solution as much as possible. This may sound rather inaccurate; the background is the following: When a SOLVE statement is executed, i.e. when GAMS generates a model, it will use the current value and basis status of all variables and equations that appear in the model, and pass this information on to the solver. If the basis appear to be good measured by the number of good variables relative to the number of constraints, then GAMS will ask the solver to use the basis, and otherwise it should be ignored, see BRATIO in one of the appendices. If you design you model properly GAMS will automatically restart in the best way. However, there are some pitfalls. If you have a dynamic model with a variable X(T,I) and T is changed dynamically from one SOLVE to the next then GAMS cannot reused previous solution values; the value of X('1','ABC') may be known, but the model uses X('2','ABC'). If you remove the T index from the model GAMS may restart more intelligently and solution times may be reduced dramatically. Another example is in decomposition: You alter between model A and model B and both models depend on a variable, say X. When you solve B it will start from the optimal X value from A and when you solve A again it will start from the optimal X value from B. By proper choice of the names of variables you may transfer variables from one model to another or you may keep the models independent so that a second solve of a model not affected by an intermediate solve of another model.

1.27 Error: endogenous $operation not allowed

My model looks for optimal transport prices (p) and optimal supply of transportation services (Q). In one of the equations I want to define the marginal benefit of extra supply (MB) as follows:

```
IF N1/Q <= 10 THEN MB = 0.5 * N2/(Q*Q)
IF N1/Q >10 THEN MB = 0.25 * N2/(Q*Q)
```

where N1 and N2 are parameters and MB and Q are variables. I modeled this as:

```
EQ.. MB =E= ( 0.25 * N2/(Q*Q) ) $ (N1/Q gt 10)
+ ( 0.5 * N2/(Q*Q) ) $ (N1/Q le 10);
```

Solving the model using DNLP results in error 53 : endogenous $ operation not allowed. Does anybody know whether there is a way to solve the problem ?

Answer from adrud@arki.dk:

The expression:

```
EQ.. MB =E= ( 0.25 * N2/(Q*Q)) $ (N1/Q gt 10)
+ ( 0.50 * N2/(Q*Q)) $ (N1/Q le 10);
```

will probably result in an unsolvable DNLP model, even if we could implement it. The problem is that MB is not a continuous function of Q. The function value jumps when N1/Q crosses the value 10. The DNLP solvers in GAMS are really NLP solvers that just try to do their best on models with discontinuous derivatives - not discontinuous functions. There are two alternative approaches:

Make the model continuous. Once the model is continuous you can often formulate it using MAX, MIN, or ABS functions to represent the two components of the equation. This results in a DNLP model that you DNLP solver may or may not be able to solve. Use a MINLP solver (DICOPT++). You will need a binary variable that is zero when N1/Q gt 10 and 1 otherwise, and the equation can then be formulated exactly as shown in terms of MB, Q, and the binary variable.

Answer from n.n:

Your condition:

```
IF N1/Q <= 10 THEN MB = 0.5 * N2/(Q*Q)
IF N1/Q >10 THEN MB = 0.25 * N2/(Q*Q)
```

where N1 and N2 are parameters and MB and Q are variables is a disjunction that requires the use of 0-1 variables. One way of representing this constraint is as follows:

```
Q = Q1 + Q2
N1*y <= 10*Q1
N1*(1-y) => 10*Q2
0<= Q1 <=QU*y
0<= Q2 <=QU*(1-y)
A = 0.5*y + 0.25(1-y)
MB = A * N2/(Q*Q)
```

where y 0-1 var; Q1, Q2, A, are new continuos variables; QU upper bound on Q. Note that Q1, Q2 are simply disaggregated variables of Q. I assumed lower bound of Q is zero. An alternative way of representing the first five equations above, and not having to introduce Q1 and Q2, is:

```
N1 <= 10*Q + M*(1-y)
N1 => 10*Q - My
```

where M is a valid "big M" parameter that renders the corresponding inequality redundant. Assuming the lower bound of Q is zero, a "good" selection of M is M=10*QU. Since this problem is small it is likely this second alternative should be sufficient.

Answer from rutherford@colorado.edu:

You need to determine which of the following characterizes your formulation:

-discontinuous derivatives
-discontinuous functions

If it (1), then you can most likely solve the problem as an NLP or MCP. If it is (ii), then you need to use the integer programming code, DICOPT. Without more information about the model formulation, I cannot tell you which avenue to explore.

Answer from rrosenthal@monterey.nps.navy.mil:

Putting aside for the moment my concern as to why would you want to use this function, let's talk about the GAMS. This is one of the more common GAMS mistakes that I see among students. Following Buckminster Fuller, I won't again use the word "mistake" but rather "learning experience." Lucky you, your learning experience is particularly instructive. To understand the reason why GAMS would not accept your equation definition is to understand the difference between modeling and solving. GAMS's job as a modeling language is to tell the solver what model you want it to solve. The solver's job is to solve. Here, the meaning of "solve" is to assign primal and dual values to decision variables. (At least that's the way an operations researcher would say it. If you talk like an economist, the solver's job is to "assign levels and marginals to endogenous variables." That language doesn't quite roll of the tongue for me, but let's get back to your learning experience.) Now, you are using GAMS to define an equation as part of the modeling process. Dollar operators in equation definitions are one of the nicer tools GAMS puts at your disposal for modeling. The problem is that inside your dollar operator is a decision variable, but GAMS can't know what value the decision variable will have. That's not GAMS's job. It's the solver's job. But the solver can't do its job until GAMS tells it what model to solve. And GAMS can't model the way you want it to because it doesn't know how to evaluate the (endogenous) variable. And so on. This is why the GAMS compiler complains when you put a variable inside a dollar condition (and why the error message mentions the word "endogenous.") By the way, if you put Q.L inside the dollar operator in the equation definition, you would have a different kind of learning experience. In that case, there would be nothing wrong with the GAMS syntax, but it still would not accomplish what you were trying to do. What would happen is GAMS would look up the current level of Q and execute the equation generation according to the dollar condition that applies to that level. To truly model if-then-else logic, as you would like, you need to use binary variables. A good, readable reference for how to it (and other tricks) is the integer programming chapter in Linus Schrage's Lindo book. Another is HP Williams' book on math programming modeling. The skill of converting logical constraints to algebraic constraints is well worth developing, and it is totally independent of whether you implement your models with GAMS or any other tool. With all due respect to the brilliant people who provide us with great solvers, I think you need to make your problem easier before calling upon a solver. Try to reformulate your model's functions without discontinuities. Try to reformulate it with continuous derivatives, or failing that, try to reformulate it as a linear

1.28 Reporting solutions from numerous runs

Is there a way to generate a report based on the solutions from numerous runs? Suppose a GAMS model has been run 100 times and 100 optimal solutions have been achieved. Suppose also that the objective function has three cost components, which have also been obtained for each run. Can we write a procedure/section in the GAMS model so that when we rerun the model 100 times, a separate file containing only the solutions is generated at the same time? For example, when the model is solved, the separate file contains four columns: the first column lists all the 100 objective function values, the other columns contain all the cost components in the objective function. Something like:

```
Obj.F. Val.   CostTerm1   CostTerm2    CostTerm3    CostTerm4
100           50          30           20           0
.             .           .            .            .
......        ...         ...          ...          ...
300           200         50           0            50
```

Answer from cs76@andrew.cmu.edu:

Have a look on this model:

```
*    An example of using the put statement in conjunction with
*    multiple solve statements in a loop
*    This particular example fragment will try all possible
*    "up" branches on the binary variables y(i,j),
*    to find the one that improves the LP relaxation the most
*    Note: in the model I'm working on at the moment, some
*    of the subproblems may be infeasible.  In these cases,
*    use option file of PRESOLVE -1, to turn off OSL presolve.
*    The presolve terminates the loops when finding an infeasible
*    subproblem, and we just want to keep going...
*    I've left the code to handle infeasible subproblems in the example,
*    just for fun, even though nothing is infeasible in this example.
*    By Craig Schmidt, cs76@andrew.cmu.edu
*    Should produce the following output file:
*        1.000000     2.000000     5.300000
*        1.000000     3.000000     8.200000
*        1.000000     4.000000     8.400000
*        1.000000     5.000000     3.000000
*        2.000000     1.000000     8.200000
*        2.000000     3.000000     4.400000
*        2.000000     4.000000     7.800000
*        2.000000     5.000000     0.500000
*        3.000000     1.000000     8.500000
*        3.000000     2.000000     8.000000
*        3.000000     4.000000     4.400000
*        3.000000     5.000000     6.400000
*        4.000000     1.000000     6.800000
*        4.000000     2.000000     4.200000
*        4.000000     3.000000     7.400000
*        4.000000     5.000000     6.100000
*        5.000000     1.000000     7.300000
*        5.000000     2.000000     4.700000
*        5.000000     3.000000     6.500000
*        5.000000     4.000000     5.100000
*    my standard settings to turn off output
$offsymxref offsymlist offuellist offuelxref
SETS
     i       tasks                / 1 * 5 /;
alias (i,j,);
alias (i,l);
alias (i, ip);
alias (j, jp);
```

```
*    -----------------------------------------------------------    *
TABLE    c(i,j) objective cost coefficients
               1   2   3   4   5
          1   6.7 5.3 8.2 8.4 3.0
          2   8.2 5.4 4.4 7.8 0.5
          3   8.5 8.0 2.2 4.4 6.4
          4   6.8 4.2 7.4 9.6 6.1
          5   7.3 4.7 6.5 5.1 0.3;
*  parameters for loops
   parameter zerooneobj;

VARIABLES
     y(i,j)          y=1 if task i comes before task j
     objfn           linear objective function;
*    define binary variables
     binary variable y;
      y.fx(i,i) = 0.0;
EQUATIONS
eq1
eq2(i,j)
eq3(i,j,l)
eq4(i,j,l);
*    demo model that doesn't really mean anything
*    this model is kind of based on the minimum cost acyclic
*    graph, but you could still get 4-cycles or 5-cycles
eq1  .. objfn =e= Sum(i, Sum(j, c(i,j)*y(i,j)));
eq2(i,j)$(ord(i) lt ord(j))..  y(i,j) + y(j,i) =l= 1;
eq3(i,j,l)$((ord(i) lt ord(j)) and (ord(j) lt ord(l)))..
      y(i,j) + y(j,l) + y(l,i) =l= 2;
eq4(i,j,l)$((ord(i) lt ord(j)) and (ord(j) lt ord(l)))..
      y(j,i) + y(i,l) + y(l,j) =l= 2;
MODEL M6      / all /;
M6.optfile=1;
*    open output to a file
file output /rightall.out/;
output.nd = 6;
*    send put statements to file specified by output,
*    which is leftall.out in this case
put output;
loop(ip,
loop(jp$(y.lo(ip,jp) ne y.up(ip,jp)),
*           try branching on binary variable y, fixing to 1
y.fx(ip,jp) = 1;
solve M6 using rmip minimizing objfn;
*           store the objective function in parameter zerooneobj
zerooneobj = objfn.l;
*           this branch may result in an infeasible
*           set objective to 9999 unless model was optimal
if(M6.modelstat ne 1, zerooneobj = 9999;);
*           send output to file
put ord(ip), ' ', ord(jp),  ' ', zerooneobj /;
*           return lower bound of y(ip,jp) to original value of 0
y.lo(ip,jp) = 0;
       );
     );
putclose output;
```

1.29 Generating a matrix using the loop statement

The problem encountered is during the generation of a matrix using the LOOP statement.Here is an example of the input....

```
LOOP(nalp, N(nalp+0,nalp)=NUM('0'));
LOOP(nalp, N(nalp+1,nalp)=NUM('1'));
LOOP(nalp, N(nalp+2,nalp)=NUM('2'));
```

It works okay for the first statement but then the second and third statements become restricted to the size of the matrix N(nalp * nalp). Here is an example of an output.....

```
----     53 PARAMETER D            matrix D
           0           1           2           3           4
0       1.000
1       4.000       1.000
2       3.000       4.000       1.000
3                   3.000       4.000       1.000
4                               3.000       4.000       1.000
5                                           3.000       4.000
```

It has missed the rows 6 and 7 which contain the values '3' and '4'. The question is if there is an efficient way to create the matrix (ie without using the table statement) that works 'fully' (i.e. produce a matrix of size 8*6 as it should in the above example). The LOOP statement used seems to be restricted in this manner.

Answer from n.n:

The problem comes from the fact that the lag/lead operator cannot address an element outside the range of the driving set. The lag/lead operator is RELATIVE to the current element. Have a look on this example:

```
set l / 0*2 /, nalp / 0*8 /;
parameter num(l) / 0 1, 1 4, 2 3 /
n1(nalp,nalp) has 8 columns instead of 6
n2(nalp,nalp) seems to be what you want;
loop(l, n1(nalp+(ord(l)-1),nalp ) = num(l)); display n1;
loop(l, n2(nalp+(ord(l)-1),nalp-2) = num(l)); display n2;
```

1.30 Solves in a loop

I have a model that I need to solve many times, until some convergence criterion is satisfied. I can do this in a GAMS loop by first defining an index set larger than the number of iterations I'll need, then looping on that index set. Then after each SOLVE I can check whether convergence is attained by doing something like ABORT $ (cvg le crit) "DONE" But when the condition is true it induces an immediate exit, so you can't do any post-processing with the results at convergence. Is there a way around this? For example, something like an "exit" statement which would terminate the loop and go on to whatever follows it? Or can anyone think of another way to get the same effect? (I though of re-defining the index set in the loop, but I don't think this can be done).

Answer from rutherford@colorado.edu:

Here is a template program:

```
set iter /iter1*iter100/;
file kcon /con/;
scalar tol convergence tolerance /0.9/
solved flag for completion /0/;
loop(iter$(not solved),
    tclose kcon, "Iteration :",iter.tl;
    lved = yes$(uniform(0,1) gt tol);
);
```

1.31 Looping GAMS, solving subsets of equations (periods) in a dynamic model

We are trying to formulate a dynamic model which only solves for few (possibly one) time period at a time. The LOOP statement could then iterate until all time periods have been solved. Suppose we have a dynamic model following the usual form, based on a time set T. We define our reference parameters over all T, and declare our variables, and equations over all T. In order to LOOP and solve for a subset of the periods, we planned to construct a "dynamic" set (in GAMS parlance) whose membership would change during the run to indicate which periods are to be solved in each loop. Below is a partial listing showing our unsuccessful approach.

```
SETS
    T Time periods /1994*2000/ {annual steps}
    TFIRST(T) First period
    TLAST(T) Last period
    CURRT(T) Current set of time periods to be solved
    CURRTFIRST(T) First period in current set
    CURRTLAST(T) Last Period in current set
    AGE AGE OF VEHICLES OR EQUIPMENT /0*3/
    C CONVERSION PROCESSES /CGASO, CM85/
    CDUR(C) CONVERSION PROCESSES USING VINTAGED DURABLE CAPITAL STOCK
        / { Begin with only vehicles having vintaged stock }
           CM85
        /
    I SUPPLY-DEMAND REGIONS /USA/
    TFIRST(T) = YES$(ORD(T) EQ 1); {set first period}
    TLAST(T) = YES$(ORD(T) EQ CARD(T)); {set last period}
    TSHOW(T) = TLAST(T) + TFIRST(T);
    SCALAR LOOPTN Number of Current Period T;
*=============================================================================
* VARIABLES and SOLVER EQUATIONS
*=============================================================================
POSITIVE VARIABLES
    STK(T,I,AGE,CDUR) STOCK OF EQUIPMENT USED BY AGE AND TYPE
    STKTOTAL(T,I) TOTAL STOCK OF EQUIPMENT USED
;
VARIABLE
    OBJ "COSTS MINUS BENEFITS ($ MILLIONS / DAY)";
EQUATIONS { DECLARE equations over all T, actually DEFINE only for subset CURRT }
STKEQ(T,I,AGE,CDUR) STOCK EVOLUTION
STKEO(T,I,AGE,CDUR) INITIAL STOCK AGE DISTRIBUTION
STKTYPEEQ(T,I,CDUR) BALANCE CAPITAL STOCK POP AND CAPITAL SERVICES
DEMAND
STKTOTEQ(T,I) ACCOUNT TOTAL STOCK POP
OBDEF OBJECTIVE FUNCTION DEFINITION;
{ Capital Stock Vintage Equation: Equipment Not Scrapped get one period older }
STKEQ(CURRT+1,I,AGE+1,CDUR)$CFEAS(CURRT+1,I,CDUR)..
    STK(CURRT+1,I,AGE+1,CDUR) =E= STK(CURRT,I,AGE,CDUR)*(1-STKSCRAPRT(AGE,CDUR));
**** $199 Error comes right here:
{ Total Stock Equation: Must be enough equipment to satisfy durable-equip services }
{ demand of each type }
STKTYPEEQ(CURRT,I,CDUR)$CFEAS(CURRT,I,CDUR)..
    SUM(AGE $(ORD(AGE)>1), STK(CURRT,I,AGE,CDUR))/STKDATA("STKPERBGD",CDUR) =E=
        CONV(CURRT,I,CDUR); { Number of AFVs of each type }
{ Accounting equation adding up all stock in each year }
STKTOTEQ(CURRT,I)..
    SUM((AGE,CDUR) $(ORD(AGE)>1), STK(CURRT,I,AGE,CDUR)) =E=
        STKTOTAL(CURRT,I); { Number of AFVs of all types }
{ Initial vehicle age-type distribution constraint }
STKEO(CURRTFIRST,I,AGE,CDUR)$CFEAS(CURRTFIRST,I,CDUR)..
    STK(CURRTFIRST,I,AGE,CDUR) =G=
    STKTOTAL(CURRTFIRST,I) * STKAGEPCTi(AGE,CDUR);
{ Initial Min Fraction of AFVs of each type }
```

Answer from rutherford@colorado.edu:

```
$ontext
The model below illustrates how to solve a dynamic economic model one
period at a time. It is based on the same input data as the RAMSEY test
problem from the GAMS library. I put this problem together to respond to
Paul Leiby's request for help. Notice that Paul's problem with looping
over a set appearing in a model's equations is entirely avoided here --
set T does not appear in the current-period equilibrium model. The
intra-period equilibrium problem presented here is trivial. The point of
the model is to illustrate:

- How to define a single period problem using temporary state
    variables.
- How to update state variables within the loop over time periods
- How to retrieve period-by-period values for reporting.
- How to compare equilibrium outcomes from a fully dynamic equilibrium
  model with a corresponding recursive model.

$OFFTEXT
* Retrieve the RAMSEY model from the GAMS library and
* process it:
$call gamslib ramsey
```

```
$include ramsey.gms
*-----------------------------------------------------------------------
* Resolve the dynamic model without lower bounds on consumption
* and upper bounds on investment:
C.LO(T) = 0.05;
I.UP(T) = +INF;
SOLVE RAMSEY MAXIMIZING UTILITY USING NLP;
*-----------------------------------------------------------------------
* Give the program a new title:
$TITLE A Recursive Version of Manne’s Ramsey Test Problem
*-----------------------------------------------------------------------
* Define the recursive model:
VARIABLES
    C_T CURRENT PERIOD CONSUMPTION (TRILLION RUPEES PER YEAR)
EQUATIONS
    CC_T CURRENT PERIOD OUTPUT BALANCE;
* Declare some parameters which will change over time:
SCALAR  AL_T CURRENT LABOR SUPPLY INDEX,
    K_T CURRENT CAPITAL,
    MPS MARGINAL PROPENSITY TO SAVE
;
* The "model" here consists of assigning a value to C_T. (Adding
* some sectoral detail, for example, would make the intra-period
* model less trivial.)
CC_T.. C_T =E= AL_T*K_T**B * (1 - MPS);
MODEL RECURSIVE /CC_T/;
* Marginal propensity to save calibrate to the benchmark values:
MPS = I0 / (I0 + C0);
* Declare some parameters for reporting:
PARAMETER INVEST, CONSUM, CAPITAL;
* Extract solution values from the dynamic equilibrium:
INVEST(T, "DYNAMIC") = I.L(T);
CAPITAL(T,"DYNAMIC") = K.L(T);
CONSUM(T, "DYNAMIC") = C.L(T);
CONSUM("UTILITY","DYNAMIC") = UTILITY.L;
CONSUM("UTILITY","RECURSIVE") = 0

* Initialize the capital stock:
K_T = K0;
LOOP(T,
    * Set labor productivity for the current period:
    AL_T = AL(T);
    CAPITAL(T,"RECURSIVE") = K_T;
    SOLVE RECURSIVE USING NLP MAXIMIZING C_T;
* Extract the solution values for reporting. During the initial
* periods, investment is determined as a fixed fraction of output. In
* the final period, we impose the lower bound on investment as a
* fraction of the capital stock. (This maintains comparability with the
* dynamic equilibrium.)
    IF (TLAST(T),
        INVEST(T,"RECURSIVE") = G * K_T;
        CONSUM(T,"RECURSIVE") = AL_T * K_T**B - G * K_T;
        ELSE
        INVEST(T,"RECURSIVE") = MPS * AL_T * K_T**B;
        CONSUM(T,"RECURSIVE") = C_T.L;
        );
* Compute utility:
    CONSUM("UTILITY","RECURSIVE") = CONSUM("UTILITY","RECURSIVE") +
    BETA(T) * LOG(CONSUM(T,"RECURSIVE"));
    * Update the capital stock:
    K_T = K_T + MPS * AL_T * K_T**B;
);
DISPLAY INVEST, CONSUM, CAPITAL;
```

1.32 Reducing the size of the listing file

Does anyone know how to completely shut off output from GAMS models to the *.lst file?

Answer from n.n:

You can reduce the listing file to the barest minimum by setting the following in the input file.

```
$offlisting offsymxref offsymlist
OPTION LIMROW = 0, LIMCOL = 0, SOLPRINT = OFF ;
```

You can further completely shut off output from the GAMS model by sending it to /dev/null on most Unix machines or NUL on PCs. For example:

```
gams trnsport o=/dev/null
```

will do the trick on a Unix platform while running trnsport.gms

1.33 Reverse loops in GAMS

Is it possible in a LOOP statement of GAMS to execute BACKWARD RECURSIVE statements, i.e. to traverse the members of the driving set of the loop in the reverse order?

Answer from n.n:

There is no syntax designed expecially for this, so you will want to avoid doing this if possible. If you must use a reverse loop, here is an example of how one can set this up. There also some goodies in there that I just couldn’t help including.

```
set i / 1 * 10 /;
alias (i,ri);
set revi(i,ri);
* the idea is to populate the sw-ne diagonal of revi.
* then looping through i in forward order gives us
* i in reverse order by taking the second index of revi
* this will work, but it is very slow, o(n^2)!
* revi(i,ri)$(ord(i) + ord(ri) eq n+1) = yes;
* this is much faster, o(n)
revi(i,i+[card(i)-2*ord(i)+1]) = yes;
scalars  n, halfpi, csum, er;
n = card(i);
```

```
halfpi = arctan(inf)
parameter c(i);
c(i) = cos((ord(i)-1)/(n-1) * halfpi);
file fp / reverse.out /;
put fp;
loop {revi(i,ri)$(ord(i) le 10),
      put ri.tl:8, c(ri):16:8 /;
     };
put /;
* reverse sums are also useful in some computations
csum = 0;
loop {revi(i,ri),
       csum = csum + c(i);
     };
put /;
er = sin(halfpi) - (csum/n * halfpi);
put 'forward sum       : ', csum:22:14 /;
put 'integration error : ', er:22:14 /;
csum = 0;
loop {revi(i,ri),
      csum = csum + c(ri);
     };
put /;
er = sin(halfpi) - (csum/n * halfpi);
put 'reverse sum       : ', csum:22:14 /;
put 'integration error : ', er:22:14 /;
put /;
putclose fp;
* just a note on efficiency:  the loop over revi goes very fast,
* since we are taking the first index, i, in order,
* and each row or revi has only one element.
* however, accessing c(ri) in reverse order is not so efficient,
* and may be a problem for very large i.
* to see this work, increase the dimension of set i,
* run the model using profiling, and check out the times:
*
* gams revloop.gms profile 1
* list revloop.lst
```

1.34 Equations in Loop

Given the case that some equations follow a structure which could be described as:

```
for n = 1 to z
for k=1 to n
    sum(c, x(c,q,t) * lad(c)) $ (ord(q)=k and ord(t)=n)    =e=
          u(q,t) $ (ord(q)=k and ord(t)=n)
    next k
    next n
```

where
c, q, t are sets,
x, u   variables and
lad   a parameter.

GAMS does not allow the definiton of equations within any kind of loop, but I would still like to use a structure similar to the one above in order to save a lot of writing. Can anybody help me with translating this double for/next loop into a proper GAMS code?

Answer from mccarl@masked.tamu.edu:

Try the following

```
set q /1*120/
alias(q,n,t);
set q1(n) particular q element in use
set t1(t) particular t element in use;
```

in your equations definitions statement

```
 constraint(q,t)
```

in your equations specification statement

```
 constraint(q1,t1)..
      sum(c, x(c,q1,t1) * lad(c))     =e=   u(q1,t1)
```

then

```
loop(q,
    q1(n)=no;
    q1(q)=yes;
       loop(t$(ord(t) le ord(q)),
          t1(n)=no;
          n1(t)=yes;
        solve the model
    ));
```

under that only the constraint q1,t1 will be active

Answer from adrud@arki.dk:

It seems as if you want to write the following block of equations:

```
eqname(q,t)$(ord(q) le ord(t)) ..
   sum(c, x(c,q,t) * lad(c) ) =e= u(q,t);
```

The condition on the equation name limits the equation to be generated to those for which ord(q) (=k) is less than ord(t) (=n).

1.35 Put file question

I have a question about put files.

Suppose I have a model that I want to run t = 1 ... T times, where on each model run a price increases by a fixed increment.

Further suppose there are 2 output files that have data sent to them via a put statement eg. land.out and water.out for each.

```
I want to set up a looping procedure where my model is within the loop and
my price increases by a fixed constant within each loop.

Is there any syntax that would allow me to include the value of,t, in the
name of output files.

For example if T = 2 I would end up with

land1.out
water1.out
land2.out
water2.out

as my put files?
```

Answer from rutherford@colorado.edu:

```
set     i /1*4/;

file    ktmp  /temp.out/;  ktmp.lw=0;
file    kcopy /fcopy.bat/; kcopy.lw=0;

loop(i,
        put ktmp, 'Results for A in iteration ',i.tl;
        putclose;
        putclose kcopy, 'copy temp.out A.',i.tl,' >nul'/;
        execute 'fcopy.bat >nul';

        put ktmp, 'Results for B in iteration ',i.tl;
        putclose;
        putclose kcopy, 'copy temp.out B.',i.tl,' >nul'/;
        execute 'fcopy.bat >nul';
);
```

1.36 Slacks

```
Is it possible to have control over the slack variables in a GAMS
program? Specifically, can a nonzero coefficient be entered for a slack
variable in the objective equation? Let's say the decision-maker can sell
some of his resource endowment (RHS) as well as use it in a productive
activity (variable). How could this be modeled?
```

Answer from rutherford@colorado.edu:

```
Replace         f(x) =g= rhs;

by              f(s) + s =g= rhs;

with s.lo = 0;

Then use s however you wish.
```

Answer from mccarl@masked.tamu.edu:

```
All you do is explicitly add the slack and then put in whatever you want
```

```
ie given
%
   max cx
       ax <= b
       ex >= f
        x >=0

make the problem

   max cx + rs + zq
       ax +  s       = b
       ex      -  q  = f
        x ,  s    q   >=0

for normal slacks r=z=0
but you can define r and z to what ever values you want

note the constraints them become equalities

also note that in the >= equations you enter a slack with a - coef
(unlike in rutherfords suggestion which allows constraint violation)
```

1.37 A sorted Table

How can I get a sorted table?

Answer from n.n:

Try this little model:

```
$TITLE GAMS Program Illustrating How to Produce Sorted Output
$ontext
This program illustrates how to take a vector of values defined on a domain
which is an unordered GAMS set and produce a output report in which the values
are listed in decending order. Thomas F. Rutherford, University of Colorado
$offtext

* Here is the "input:"
SET G /A,B,C,D,E,F/; ALIAS (G,GG)
PARAMETER V(G) Random values to be sorted for illstration;
OPTION SEED=1001;
V(G) = UNIFORM(0,1);


$ontext
The program takes this data and generates the following output:
---- 29 PARAMETER SORTED Sorted list of values for display
VALUE
1.A 0.8970
2.D 0.6751
3.C 0.1731
4.E 0.1336
5.B 0.0742
```

```
6.F 0.0464

Because the random number generated is seeded, the output from thisprogram
will be the same on any platform which runs GAMS. $offtext

* The code for doing the sorting is as follows:
SET I An ordered set larger than G /1*1000/;
PARAMETER SORTED(I,G,*) Sorted list of values for display
RANK(G) Rank order of item in the list;
RANK(G) = SUM(GG$(V(GG) GE V(G)), 1);
SORTED(I,g,"value") = V(G)$(RANK(G) EQ ORD(I));
* Make the display use 4 decimals with list format
OPTION SORTED:4:2:1;
DISPLAY SORTED;
```

1.38 String manipulation in GAMS

I am working on an application where I want to get information that is embedded in set element names,chop it up and display it in a different form in my reports. For example, B0795XT is an element of a set from which I want to extract and report the following information:

```
Mode : B
Date (mm/yy) : 07/95
Item type : XT
```

Is there any way to do this within GAMS?

Answer from n.n:

GAMS has no direct string manipulation capability. The GAMS example below will produce the report you wanted. I used the capability to overwrite the put file. I was lucky that you wanted it in the form mm/yy. This trick would not have worked for yy/mm. Also note that @ is an operator that works on expressions as well. For example you could use @(x-2) etc.

```
set i some item names / B0795XT, B0795AT /
file out; put out;
put / ’Type one’ / ;
loop(i,
    put / ’- Mode : ’ @20 i.tl:1
    / ’- Date (mm/yy) : ’ @20 i.tl:5 @19 i.tl:3 @19 ’ ’ @22 ’/’
    / ’- Item Type : ’ @15 i.tl:7 @15 ’ : ’
    /
     );
put / ’Type two’ / ;
loop(i,
    put / @20 i.tl:1 @1 ’- Mode : ’
    / @20 i.tl:5 @19 i.tl:3 @22 ’/’ @1 ’- Date (mm/yy) : ’
    / @15 i.tl:7 @1 ’- Item Type : ’
        /
     );
```

Results:

```
Type one
- Mode : B
- Date (mm/yy) : 07/95
- Item Type : XT
- Mode : B
- Date (mm/yy) : 07/95
- Item Type : AT
Type two
- Mode : B
- Date (mm/yy) : 07/95
- Item Type : XT
- Mode : B
- Date (mm/yy) : 07/95
- Item Type : AT
```

1.39 PUT-ing the element text of created subsets

In the attached example, I want to use the "element text" corresponding to the SUBSET CF (below) of CASES. Unfortunately, if I just list the elements when I declare the subset BCF(CASES) (from which CF is derived) the element text is blanked out. Is there a way to get around this? I want to avoid having two copies of the element text each time as I have done in the (working) example attached.

```
SET CASES      All possible cases including the benchmark
    /
    BCH        Benchmark,
    TRD_STD        Trade Standards,
    PRO_STD    Process standards,
    S_CLEAN    South Cleanup,
    N_CLEAN    North Cleanup
    /,
        BENCH(CASES) benchmark alone
        / BCH   /,
    BCF(CASES) Benchmark plus counterfactuals to do now
    /
    BCH        Benchmark,
    TRD_STD        Trade Standards,
    PRO_STD    Process standards
    /,
***
***  If I just list the elements in BCF (which seems quite natural)
***  the "element text" gets overwritten to nothing.
***
    CF(CASES) Counterfactuals  ;
CF(CASES) = BCF(CASES) - BENCH(CASES) ;
ALIAS( CF, CFCTL) ;
* Create Tables
FILE LTXTABLE /tables.tmp/ ;
PUT LTXTABLE ;
LOOP( CFCTL,
* TeX File identifier line
PUT "% % % %  Tables for experimment " CFCTL.TL "  % % %" //;
*  summary table header
PUT "\caption{" CF.TE(CFCTL) " Summary\label{" CFCTL.TL "}}" // ;
***
*** The previous statement does PUT the correct text IF
*** I add the element text to the declaration of the SUBSET BCF
***
* I PUT the data here
   ) ;
PUTCLOSE LTXTABLE ;
```

Answer from rutherford@colorado.edu:

```
Here is a solution to this question. I use a controlled loop in which
both references to the set CASES are specified. This way we can use the
element text for CASES while also referring in the same loop to the
subset CFCTL. I came across this syntax last year, and find it is much
more flexible than loop(set$subset(set). Why not produce a truly general
purpose TeX table facility which can be called "blind". Something like:
$libinclude textable <item>.See
http://www.gams.com/contrib/gams2txt/gams2txt.htm for hints -- we include
all of the GAMS source code for these tools, so it should not be too
difficult to mimic. (If you restrict it to 2-dimensional items, you might
also look at the gnuplot interface for ideas.

*   In the following example, I want to use the "element text"
*   corresponding to the SUBSET CF (below) of CASES. Unfortunately,
*   if I just list the elements when I declare the subset BCF(CASES)
*   (from which CF is derived) the element text is blanked out. Is
*   there a way to get around this?
*
*   I want to avoid having two copies of the element text each time
*   as I have done below.
SET CASES      All possible cases including the benchmark
    /
    BCH        Benchmark,
    TRD_STD        Trade Standards,
    PRO_STD    Process standards,
    S_CLEAN    South Cleanup,
    N_CLEAN    North Cleanup
    /,
    BENCH(CASES) benchmark alone
       / BCH   /,
    BCF(CASES) Benchmark plus counterfactuals to do now
    /BCH, TRD_STD, PRO_STD /,
***  If I just list the elements in BCF (which seems quite natural)
***  the "element text" gets overwritten to nothing.
***
    CF(CASES) Counterfactuals  ;
CF(CASES) = BCF(CASES) - BENCH(CASES) ;
ALIAS( CF, CFCTL) ;
* Create Tables
FILE LTXTABLE /tables.tmp/ ;
PUT LTXTABLE ;
LOOP( CASES(CFCTL),
* TeX File identifier line
PUT "% % % %  Tables for experimment " CFCTL.TL "  % % %" //;
*  summary table header
PUT "\caption{",CASES.TE(CFCTL)," Summary\label{" CFCTL.TL "}}" // ;
***
*** The previous statement does PUT the correct text IF
*** I add the element text to the declaration of the SUBSET BCF
***
* I PUT the data here
   ) ;
PUTCLOSE LTXTABLE ;
```

1.40 The gams225?-Subdirectories

When I run my GAMS model from a temp subdirectory, other directories are created in the directory in which I am running. These directories are named 225a, 225b, 225c, etc. and contain either one file called gamsparm.scr or several files with the gams*.scr name and a gamsnext.bat - file. My question is, do I need to keep these 225a (etc.) directories, or canI delete them from my computer as they take up space?

Answer from n.n:

These directories are only needed while your GAMS job is running. If you have those directories still there you either used 'gamskeep' instead of 'gams' or your job crashed for some strange reason. You can deletes all these directories (.deltree 225?).

1.41 The gams225?-Subdirectories

When I run my GAMS model from a temp subdirectory, other directories are created in the directory in which I am running. These directories are named 225a, 225b, 225c, etc. and contain either one file called gamsparm.scr or several files with the gams*.scr name and a gamsnext.bat - file. My question is, do I need to keep these 225a (etc.) directories, or canI delete them from my computer as they take up space?

Answer from n.n:

These directories are only needed while your GAMS job is running. If you have those directories still there you either used 'gamskeep' instead of 'gams' or your job crashed for some strange reason. You can deletes all these directories (.deltree 225?).

1.42 Calling GAMS from Fortran

Using available fortran routines (adaptive random search) I can easily generate different input data for a NLP/MINLP problems. For every input data I'd like to solve the problem using GAMS programs (already written!). My question is, how to make a GAMS call inside a fortran routine?

Answer from rutherford@colorado.edu:

You need to have a system call to do this -- this works fine on the PC, under DOS or Win95. It also works fine with Unix Fortran. The basic command is something like: CALL SYSTEM('gams myfile') Be sure to close the file your Fortran code writes before calling GAMS. Here is a program which works with Lahey F77L (the old DOS compiler) under Windows NT:

```
        real test
        open(10,file='myfile.gms')
        write(10,'(a)') 'scalar one /1/;'
        write(10,'(a)') 'file kout /myfile.out/; put kout;'
        write(10,'(a)') 'putclose kout, one;'
        close(10)
        call system('gams myfile')
        open(11,file='myfile.out')
        read(11,*) test
        write(*,*) test
        end
```

1.43 Batch-processing on PC

I want to run multiple GAMS-sessions in sequence in the weekend on the PC. When I make a batch-file containing multiple GAMS-invokations, it halts after the first execution of GAMS. How can I overcome this?

I choose this construction because this is in my program the easiest way to handle runs on different data sets. Moreover, it should prevent stopping all jobs after the event of an execution error in one of them. On UNIX, this always worked fine by use of a shell script.

Answer from n.n:

I think your problem might be that if you call a BAT-file from within a BAT-fiel, DOS does not return from the second BAT-file unless it is called with the command CALL. Since the GAMS-command is really a BAT-file, making a BAT-file with:

```
GAMS <firstgamsfile>
GAMS <secondgamsfile>
```

will not work, but

```
CALL GAMS <firstgamsfile>
CALL GAMS <secondgamsfile>
```

and you should be OK when you give the command RUN late Friday when you head off for beers. The "call" keeps the batch file in DOS memory, so that when the first job is finished it knows what to do next.

1.44 Flexible $include-statements

I have a GAMS program that I would like to reuse for several of our Portfolios. I would like to be able to incorporate the data specific to the Portfolio I am running for at any given time (name and holdings amount) using an $include statement and a file. The file name is expected to be different for each Portfolio (e.g. curr_portfolio_segA.inc, curr_portfolio_segB.inc).

Answer from rutherford@colorado.edu:

Look at this example:

```
---------------------------- cut here for run.bat----------------------------
:==>run.bat     Accepts portfolio name as first argument, segment name as
second
@echo off
:       There must be a TAB following echo on the following two lines!
if a%1==a goto syntax
if a%2==a goto syntax
echo    $setglobal portfolio %1 >defines.gms
echo    $setglobal segment   %2 >>defines.gms
call gams model
goto end
:syntax
echo    Syntax:  run portfolio segment
:end
---------------------------- cut here for model.gms----------------------------
$title  Code fragment illustrating the use of setglobals to assign file
names.
$include defines
$include P%portfolio%_%segment%.dat
*... process the data
*       Write output to a file associated with the current portfolio:
file kout /P%portfolio%_%segment%.sol/; put kout;
*... put statements
*       Pass output to a spreadsheet of the appropriate name:
$libinclude ssexport item P%portfolio%_%segment%.wk1 solution
```

1.45 CPU time

Below are the CPU times reported in the GAMS output files of the solution of a given NLP problem are:

Using MINOS5 as solver:

```
        - COMPILATION TIME : 1.050 sec. (1)
        - GENERATION TIME  : 2.010 sec. (2)
        - EXECUTION TIME   : 2.430 sec. (3)
        - EXECUTION TIME   : 0.740 sec. (4)
```

1) Which of the two execution times [(3) and (4)] reported is the
    real CPU time expended in the resolution of the NLP problem?.
2) How all these times((1), (2), (3) and (4)) related each other?
3) If I want to report the CPU time require for the solution of a
    given problem using GAMS, in order to compare it with others procedures, which
    time I should write (3) or (4)?

CASE B: Using CONOPT as solver, the CPU times reported for the same NLPproblem are:

```
        - COMPILATION TIME : 0.960 sec.   (5)
       - GENERATION TIME  : 1.080 sec.   (6)
       - EXECUTION TIME   : 1.480 sec.   (7)
CONOPT time Total            1.930 seconds  (8)
```

2) The same questions as for MINOS. Which is the difference between time (7) and (8).
3) How I should understand all these times if I want to compare
    solution times using MINOS and CONOPT as alternative solvers?

CASE C:Below are the CPU times and the log file reported for a MINLP problem.

```
        - COMPILATION TIME : 1.940 sec.    (9)
        - GENERATION TIME  : 1.240 sec.    (10)
        - EXECUTION TIME   : 1.920 sec.    (11)
 ------------------------------------------------------------------
 DICOPT Log File
 ------------------------------------------------------------------
 Major Major     Objective    CPU time  Itera- Evaluation Solver
 Step  Iter       Function     (Sec)    tions   Errors
  NLP    1      8892.62853     10.76      144       0      minos5
  MIP    1        69.53081      9.06      152       0      osl
  NLP    2        *Infeas*      3.46       22       0      conopt
  MIP    2        69.53081      8.35      167       0      osl
  NLP    3        69.53081<     3.57       14       0      conopt
  MIP    3        63.04459      8.95      190       0      osl
  NLP    4        *Infeas*      4.12       19       0      conopt
  MIP    4        63.04459      9.06      190       0      osl
  NLP    5        *Infeas*      3.90       28       0      conopt
  MIP    5        26.60000     10.00      210       0      osl
  NLP    6        *Infeas*      4.50        4       0      conopt
  MIP    6        26.60000      7.31      196       0      osl
  NLP    7        26.60000      2.19        6       0      conopt
 ------------------------------------------------------------------
   Total solver times :  NLP =    32.50   MIP =    52.73
   Perc. of total     :  NLP =    38.13   MIP =    61.87
 ------------------------------------------------------------------
        - EXECUTION TIME   : 0.810 sec.    (12)
```

1) Again, which is the real CPU execution time (11) or (12)
2) What is the relation of the CPU times reported in the log file
   with both EXECUTION TIMES (11) avd (12) (all times seems to be given in
   seconds!!)

Answer from n.n:

As noted, the listing file contains a number of different times. On a UNIX system, GAMS times can be obtained via the command grep "SECONDS" trnsport.lst

1) COMPILATION TIME: GAMS uses a two-pass process to compile and run a

model. The compilation time is the time required to perform the first (compilation) pass, and is often dominated by the time required to read in large data files, especially if the data appears in column-major order.

2) EXECUTION TIME: time required for the second (execution) pass. This includes model generation time (processing a solve statement and writing the problem to disk). Other big factors can be heavy numerical calculations on existing data and the use of "long" loops. Parallel assignments should be used wherever possible. For example:

```
    set I / 1 * 100000 /;
    parameter u(I);
    * bad!
    loop { I,
        u(I) = uniform(0,2);
    };
    * good
    u(I) = uniform(0,2);
```

3) GENERATION TIME: time required to process a solve statement. GAMS constructs the problem at this time, and writes it to disk. A dominant part of execution time. There are two execution times because GAMS resumes execution after the solve to read in the solution and do any reporting necessary.

To get the time required to solve the model, you want to look at resource usage by the solver, for example:

```
  grep "RESOURCE USAGE" trnsport.lst
RESOURCE USAGE, LIMIT          0.190     1000.000
```

This appears in the solve summary, grep "S O L V E". Note that this time is not included in the GAMS execution time.

1.46 Precision problems

I have some problems with the machine precision of real numbers. I am trying to calibrate a demand system derived from a Symmetric Generalized McFadden Cost Function to a set of demand elasticities, quantities and prices. The first step, calibration to the base year situation, works pretty good, but in a second step I want to introduce time dependency by calibrating to a trend estimation for some target year t years from the base year. In principle, the following equation respresents a simple LP to find the unknown parameters C(i) (all symbols but C are predefined parameters):

```
 f(i) * g - gi(i) =E= (t - f(i) * t * p(i)) * C(i)
   - sum(j$(ord(j)<>ord(i)), f(i) * t * p(j) * C(j) );
```

with i and j being aliased sets over the products. However, due to the very unequal expenditure shares of the different products the value of function f(i) times the price p(i) may happen to have dimension 1.E-09, so that they are very small relative to unity. Consequently, in the term (t - f(i)*t*p(i)) the second part - f(i)*t*p(i) actually is ignored, leading to wrong results for the parameters C(i). Thus the question to more experienced GAMS users: Is there any possibility in GAMS to increase the precision of value representation (as there are the DOUBLE PRECISION or REAL*8 variables in FORTRAN)?

Answer from n.n:

Calculations in GAMS are done in double precision (64-bit) arithmetic, so there is not much room for improvement there. Using quad precision is possible on some architectures, but the slowdown would be considerable; I have seen reports of tenfold increases in time on applications that do pure number crunching. The default behaviour is to pass the problem to a solver in a binary format that causes no loss of precision, the solvers use double precision, and they pass the solution back in binary format, so there shouldn't be a problem there either. GAMS does not typically ignore values in the range 1e-9 (unless the rest of the equation looked like 1e9). Perhaps you need to increase the precision to which the variable C(i) is displayed.

Answer from n.n:

Regarding the constraint you mentioned,

```
f(i) * g - gi(i) =E= (t - f(i) * t * p(i)) * C(i)
  -  sum(j$(ord(j)<>ord(i)), f(i) * t * p(j) * C(j) );
```

Forgive me for asking but I can't help wondering why you have written it that way. (Fortran programmers should be good at introducing intermediate variables, moving things out of loops, and so on!) How about this:

```
   Variables  C(j), S                            (that's a comment)
              S  =E=  sum( j, p(j)*C(j) );       (new constraint)
   f(i)*g - gi(i)  =E=  t*C(i) - t*f(i)*S;       (is this equivalent?)
```

Although you've worried about f(i)*p(i) being small for some i, S probably won't be small. You shouldn't need to worry about precision (16 digits is enough for most models, and there's no choice anyway!). Just make sure that t, f(i), gi(i), p(i) are not huge numbers, and that a typical C(i) is not huge either.

1.47 Help with put facility

When we run GAMS models through a lop, GAMS includes a list of active stage in the loop run just before model statistics. This is very handy to keep verify the results of the run.

My question is can we get this info written into a put file? What suffix is used to indicate this in the put statement?

PS: Here is the loops I used and an example of the listing that I refer to:

```
set run /run1/;
set price /p1*p10/;
Set scenario scenario identifier     / scenario-1{*scenario-2} /;
Set millmax mill capacity identifier / mmax1{*mmax2} /;
loop(millmax,
     mmax(y,m) = mmax('1996',m)* 1.01;
loop(price,
    PPS = PPS + 10;
loop(run,
     Loop(scenario,
         PP(f) = iterm * PPS * (ccs(f)-4) + Kterm;
         lnterms(ber) = CF/(PP('hiinput')*PM(BER)*d(ber)*b(ber));

solve Herbert using nlp maximizing PVNB ;

$include 'putaares99d.txt';
$include 'putout2d.txt'
     );
     );
         );
     );

Extract from GAMS output:

LOOPS                           MILLMAX      MMAX1
                                PRICE        P1
                                RUN          RUN1
                                SCENARIO     SCENARIO-1
```

Answer from gideon.kruseman@alg.oe.wau.nl:

```
Actually it is fairly simple, use your loopcounter in the put statement

loop(cnt1,
    loop(cnt2,
        loop(cnt 3,
            solve model using nlp maximizing z;
            put 'active stage:   ', cnt1.tl, cnt2.tl, cnt3.tl /;
            put 'whatever you want to know'/;


            );
        );
    );
```

1.48 Include statement with wild cards

```
Is to possible to use include statement with wild cards in
include  multiple files?

For example

 $include "m*.inc"
  >
```

Will it include all the files starting with 'm' and having
extention 'inc'?

Answer from rutherford@colorado.edu:

Have a look at the example below:

```
*==>test.gms    Shows how to include all files matching a given
pattern.

$call 'if exist incfile del incfile'
$call 'for %%f in (m?.inc) do echo $include %%f >>incfile'
$include incfile display one, two;

*==>m1.inc scalar one /1/; *==>m2.inc scalar two /2/;
```

1.49 Counting the number of Equation for a specific constraint Content-Type

I wonder if there is a nice way to count the number of the
equation that is is related to a specific constraint. For example,
a gams expression that is

```
timing2(I,L,II,LL,J,K)$(Set_Li(I,L)*Set_Li(II,LL)*SET_Ji(I,J)*SET_Ji(II,J)
        *Set_Kj(J,K)*SET_Jl(L,J)*SET_Jl(LL,J)$(ORD(I) NE ORD(II)))..
```

How can I count the number of equation generated by
timing2? Is there any suffix or index I can refer to?

Answer from thorntop@hoffman.army.mil:

If you use the limrow = 1
option, then the output of your gams model will include the first
instance of every equation you have defined and include a message
indicating the number of skipped rows also generated.

1.50 Loops and calling external programs from GAMS

I'm using GAMS to do a series of period-by-period optimisations. At each step
I have to do some complex data processing to generate some of the data; this
is sufficiently involved to oblige me to do it outside GAMS. There's also a
set of data files, one of which gets $INCLUDEd in each period. For example,
DATA.1 is needed for period 1, DATA.2 for period 2 etc.

I'd like to use a FOR or LOOP to call the external program, set up the file
names and do the optimisation. I've therefore got two questions I'd appreciate
some help with:

1) Are the GAMS '$CALL' and 'EXECUTE' commands for executing
external programs (as used in SSDUMP.GMS) documented anywhere?

2) Is there a mechanism for constructing a string of the form

DATA.I (where I is the value of the FOR or LOOP index) that can be used in an $INCLUDE or PUT? I’ve tried things like

```
%
FOR ( I = 1 to N, ......... $setlocal infile DATA.I ;
$include %infile% ;
.......... ) ;
%
```

but this doesn’t work, as the variable infile always takes the value ’DATA.I’ not ’DATA.1’, ’DATA.2’, etc.

Answer from **gideon.kruseman@alg.oe.wau.nl**:

I have tried the same myself until I realized that it doesn’t work and will never work. Why is this so: GAMS compiles the complete model at the start of the "run" which means that all include statements are executed before actual execution of the optimisations. It just means you have to be creative to get all your calculations into GAMS or else restart GAMS after each external calculation.

Answer from **rutherford@colorado.edu**:

(i) It is correct that you can only output data at execution time. You can only input data at execution time through a SOLVE. This means that you might find it easier to do the data processing within your loop by figuring out how to do it with GAMS. (I admit that several years ago, I always moved stuff into Fortran for complex operations, but as time has passed, I increasingly move logically complex tasks in the other direction.) If you tell us what the complex task is that needs to be done off-line, someone on the list might have some tips on how to do it with GAMS.

(ii) Regarding your question about documentation, I understand from Alex that a set of documents describing GAMS command language is being generated. I have picked the syntax up by trial and error, with occasional visits to GAMS.

(iii) How to generate data files with a set index name? Here is an example which runs under NT.:

```
set     i /1*3/; parameter
a(i); file dat /temp.dat/; file bat /copyit.bat/;  bat.lw=0;
$libinclude gams2txt set iter /it1*it3/; loop(iter,
  a(i) = uniform(0,1);
  put dat;
$libinclude gams2txt a
  putclose;
  putclose bat, ’@echo off’/’copy temp.dat ’,iter.tl,’.dat >nul’/;
  execute ’copyit >nul’;
);
```

This generates files IT1.dat, IT2.dat, and IT3.dat.

Here I am writing numbers using a $libinclude routine, but you could just use PUT (See http://robles.Colorado.EDU/~tomruth/inclib/tools.htm)

Common programming error in this type of thing is to pick a DOS reserved name for the batch file -- note that if I had called the batch command file COPY.BAT rather than COPYIT.BAT, who knows what would happen.

Of course, I would be inclined to code up this approach in a LIBINCLUDE routine so that I would not need to remember all the syntax. For example, you could define gams2fil as a file output routine. Then the example shown above would be:

```
set     i /1*3/; parameter a(i);

*       Blank invocation required before calling
*       the routine from inside a loop:

$batinclude gams2fil

set iter /it1*it3/; loop(iter,
  a(i) = uniform(0,1);

*       First argument is the parameter to dump, and the
*       second argument is the name of the destination file
*       in PUT format.  Note that the double quotes are
*       stripped off by $batinclude.

$batinclude gams2fil  a    "iter.tl,'.dat'" );
```

Here is gams2fil.gms:

```
$if defined bat $goto start $libinclude gams2txt file bat
/copyit.bat/; bat.lw=0; file dat /temp.dat/;

$if "%1"=="" $exit

$label start put dat;
$libinclude gams2txt %1
putclose;
putclose bat, '@echo off'/'copy temp.dat ',%2,' >nul'/;
execute 'copyit >nul';
```

1.51 On bugs in ssimport

I have tired to use your utility SSEXPORT to write results into a spreadsheet. Unfortunately, it does not work, at least on my PC. I have also tried to run your example ssexamp.exe. Again I received an error message which I have attached to this message. Any help or suggestion?

Answer from rutherford@colorado.edu:

GAMS changed its install program during the past year. Beginning with version 2.25.091, the "execute" statement is a default part of the language. This was not the case with version 2.25.089.

If you are running SSLINK and you receive an error message like:

```
LIBINCLUDE C:\GAMSINST\INCLIB\sectionEXPORT.GMS

 400  execute 'echo  SSEXPORT CL ramsey.wk1 CL -M';

****        $140$36
**** LINE    102 IN FILE C:\GAMSINST\INCLIB\sectionEXPORT.GMS
****  36  '=' or '..' or ':=' or '$=' operator expected - rest of statement ign
**** 140  Unknown symbol
```

then you need to edit the GAMSPARM.TXT file in your GAMS system directory, replacing the statement:

```
g205 2
by
g205 0
```

Sorry for the inconvenience. If you upgrade to the latest version of GAMS this problem goes away.

1.52 How to get an equation listing without solving the model

I want to display the equations, while I'm writing them. I mean that I want to see them without solving the model, just to check if they are OK.

Answer from sdirkse@gams.com:

A quick and dirty way to see a equation or variable listing for some of the equations is to define a dummy objective $(min x | x = 0, say)$, change the equations you are interested in to =n= (nonbinding), put them into a dummy model, and solve this model.

For example,

```
set I / 1 * 3 /;

variable x(I), z;

equations
norm2,
dummyobj;

* this one is part of the "real" model, but now is =n=
norm2 .. sum {I, power(x(I),2)} =n= 4;

dummyobj .. z =e= 5;

x.l(I) = 3;

model foo / dummyobj, norm2 /;

solve foo using nlp minimizing z;
```

Now check out the equation and variable listing in the .lst file

1.53 Functions

I was wondering if gams supports the calculation of averages and medians. Is there a function for counting numbers of an index. To give you an example of what I mean: I have a multi region model and want to add up the results for all regions in my output file. This can be done using

```
sum(index1, ...
```

But I have some output parameters I don’t want to add up over the regions but rather calculate an average and median over all regions. So is there a function like

```
avg(index1, ... ?
med(index1, ... ?
```

If not I might have to do it by hand by summing all the values of the regions and divide them by the number of regions. In case I don’t run all regions at a time but only a selection I need to express the number of regions in a general fashion by counting it. And this is my third question: Is there a way to count numbers in GAMS like

```
count(index1, ... ?
```

Answer from adrud@arki.dk:

There is no mean or average function. You will have to compute them the usual way. The count is done using

```
sum(index[$condition], 1)
```

If you have no condition you may use card(index) that returns the number of elements in the set index (or in general: the number of elements in a multidimensional set or the number of nonzero record in a parameter, or the number of nondefault records in a variable or equation).

Answer from thorntop@hoffman.army.mil:

I don’t believe there is an "average", "median", or "count" function but you can use the almighty $ operator to set up the condition that will sum the indices you want or increment a counter for the indices you desire.

For example, if you want to average the variable x for each index i that was even then average could be calculated as

```
    average = sum(i$(mod(i,2)=0), x.l(i) ) / sum(i$(mod(i,2)=0), 1) ;
```

Or if you want the average of x over a subset of the i indices, let j be this subset and

```
    average = sum(j, x.l(j) ) / CARD(j) ;
```

1.54 The $abort statement

I am using checks with ABORT statements as warning flags. Unfortunately, once an ABORT statement has been encountered, it seems impossible to RESTART from there (possibly after making a few checks/changes): is there no way to neutralize the effetcs of the ABORT statement once it has been met? It seems so ridiculous to have to restart the whole solution procedure (which in my case is a multisolve iterative procedure) from zero!

Answer from glennwh@ix.netcom.com:

But why ABORT when you don't want to? That seems to me to be the ridiculous thing, not how GAMS reacts to your instruction. Just exit the program and the SAVE and RESTART options will be useable.

Answer from saddy@cba.ua.edu:

I personally believe that the main utility of ABORT is in model development, using checks. I think, however, that what you want to do is to use the dollar operator for exception handling (checks), make changes and then continue. If you really do need to exit the program then you might consider using the SAVE and RESTART facilities.

Answer from rutherford@colorado.edu:

When I am developing a large GAMS application I often use the $label, $goto and $exit statements during development work. The program development algorithm is:

```
1) Insert an $exit at some point down the program and run GAMS to that
    point with a save, i.e.  GAMS program s=s1
2)  Add a $label s1 immediately after the first $exit statement and add
    a $goto s1 statement at the top of the file.  Then insert a second $exit
    statement and GAMS program r=s1 s=s2.
3)  {Repeat step ii for s2, s3, ...}
```

If you are working on s5 and you discover that you made an error in s3, then you can restart from s2 without having to rerun sections 1 and 2. This approach is particularly helpful if you are working on a project which requires a large number of preliminary calculations, such as matrix balancing operations.

If I run into a problem figuring out what is going on at some point in the program, then I write a separate GAMS program file to generate debug output, and I process that file with a restart from one of the save files. This approach avoids littering your program with extraneous debug-related symbols.

Of course the alternative to this method is to split up the program into s1.gms, s2.gms and so on, but I dislike having to edit across multiple files. If you keep everything in one file then you can easily search for symbols backwards and forwards over the entire program.

1.55 Check for empty dynamic sets

I am working with dynamic sets in GAMS. How can I check whether a set is EMPTY?

Answer from boehringer@zew.de:

Just do a simple display on the set
and you will have the answer.

Answer from rutherford@colorado.edu:

```
if (CARD(dset) eq 0,
        display "The set is now empty";
);
```

1.56 Using the screen as the put file

```
I am looking for a way to bring messages to the screen while the
program is running. In the manual, I found the PUT statement to be
the appropriate way, but no advice for the syntax. I've tried
the following style:
   ...
   FILE CON;
   PUT CON 'screen message text'/;
   ...

hoping that the text 'screen message text' would be displayed to
the  screen, but it is not.
```

Answer from sdirkse@gams.com:

```
Here is a small example showing how this can be done, using some code to
take care of any OS dependencies. I prefer this style, as it makes the
name of the PUT file explicit. In the example above, GAMS uses an
implicit PUT file name when none is given: <handle\_name>.put (in the
above model, con.put).  This doesn't result in screen output for Martin,
nor does it on my Windows 95 box here.

$set console
$if %system.filesys% == UNIX  $set console /dev/tty
$if %system.filesys% == DOS $set console con
$if %system.filesys% == MS95  $set console con
$if %system.filesys% == MSNT  $set console con

$if "%console%." == "." abort "filesys not recognized";

file screen / '%console%' /;
put screen;
put /"output from put statement"/;
putclose;

Note also that on older GAMS systems, the %system.filesys% variable
will not be defined.  It was introduced with build 089 (September 96).
To check this, just

display "%system.filesys%";
```

1.57 Multiple solve

```
We use input files containing more than one solve and for each one the
initial values are very important. More precisely, the solves belong to
several series and, for each series, the environment must be the same as
for the first series, i.e. "the initial values for the second and
```

subsequent solves (of the series) are the final values returned from the previous one" (see the end of section 15.1 (a), p. 156, of Release 2.25 GAMS user's guide, 1992), while those for the first are (in general only partialy) specified. The problem is to do in such a way that the conditions of implicit initial values are the same for the second and subsequent series than for the first one. Does there exists a gams instruc- tion - to put at the beginning of each series - which put again the environment of solve which is implicitely put at the beginning of Gams execution ? Note : the option SOLVEOPT does not seem convenient for this purpose.

Answer from mccarl@masked.tamu.edu:

Before the first solve in the loop save xour initial values

```
parameter savexl(set1,set2) saved values of variables
          savexm(set1,set2) saved marginals for variables
          saveeqm(set1)     saved marginals for equations;
savexl(set1,set2)=x.l(set1,set2);
savexm(set1,set2)=x.m(set1,set2);
saveeqm(set1) =  eq.m(set1);
* this would be needed for each equation and variable in the model
then in your solve loop use
loop(cases,
x.l(set1,set2)= savexl(set1,set2);
x.m(set1,set2)= savexm(set1,set2);
eq.m(set1)= saveeqm(set1);

solve ...

);
```

this would always start with the same basis.

1.58 Loops over subsets

I would like to run a loop over two subsets of the same main set, say f(i) and l(i), and to do some assignments depending on wether or not the current element of f equals the current element of l. Any idea how to do this? An example:

```
SET I 'all products'
    /wheat, corn, beef, milk/;
SET F(I) 'feedingstuffs'
    /wheat, corn, milk/;
SET L(I) 'lifestock products'
    /beef, milk/;
... (data input and parameter definition)...
LOOP((F,L),
* and now, how to manage something like:
   IF(F=L,
      do some data assignment
   );
* i.e., do the assignment only for F=milk and L=milk
);
```

Answer from thorntop@hoffman.army.mil:

```
I believe this will work for you

SET I 'all products'
    /wheat, corn, beef, milk/;
SET F(I) 'feedingstuffs'
    /wheat, corn, milk/;
SET L(I) 'lifestock products'
    /beef, milk/;

PARAMETER FILL(I);

LOOP(I,
    FILL(I) = 1$(F(I) and L(I) );
   );
DISPLAY FILL;
end{verbatim}
```

Answer from mccarl@masked.tamu.edu:

```
the function sameas(i,j) can be used
it retuens a true if set element i = set element j
if returns a false otherwise

sum((i,j)$sameas(i,j), expression)
   only sums when the set element name for i = that
   for j

sum((i,j)$(not sameas(i,j)), expression)
   only sums when i and j are not equal

sum(i$(not sameas(i,"cleveland")), expression)
   only sums when i is not the set element cleveland

thus you can put specific values in for sets and can include
this syntax anywhere a conditional can be used
```

1.59 Summation Question

```
I have a question regarding controling index while performing
summation on gams.


Two sets I  = /1,2,3,4,5/
              J =  /1,2,3,4,5/
             Variable U(I,J)

declaring 25 equations, for each combination of I and J:
equation(I,J)..    SUM  U for all I such that index J goes from 1 to I      = 0;

for equation (1,1)   J=1
for equation(1,2)   J=1,2
```

```
etc..
```

Answer from adrud@arki.dk:

```
You cannot have an equation depending on both I and J and then Sum over
I. You will get a message about "Set already under control." I assume
that you want one equation for each I where you sum J from 1 to I.
Try:
equ(i) .. sum(j$(ord(j) le ord(i)), u(j) ) =e= 0;
```

Answer from mccarl@masked.tamu.edu:

```
Try:

set i /1*5/
alias(i,j);
z=sum(i,
   sum(j$(ord(j) le ord(i)), expression);
   ...

or

set i /1*5/
set j /1*5/
z=sum(i,
   sum(j$(ord(j) le ord(i)), expression);
   ...

other strategies can be used if the set is not ordered.
```

Answer from rutherford@colorado.edu:

```
You need to be more precise about the equation.  If you are including the
indices I,J in your equation declaration then you cannot refer to those
indices in the body of the equation.  This is ordinary rules for
mathematics, nothing idosyncratic about GAMS.  Adding set indices in the
declaration is equivalent to saying "for all (I,J)" in your mathematical
statement. I can guess that what you want to write is something like:

SET     I /1*5/;

PARAMETER  N(I);  N(I) = ORD(I);

EQU(I)..        SUM(J$(N(J) LE N(I)), U(J)) =E= X...;

If this is what you want to write, then J cannot appear in the
equation identifier because it is controlled by the summation.
```

1.60 Many to many mapping

```
I'm trying to master many to many mapping with SETS. I have the feeling that
this can be used to filter variables not used in a specific model within an
environment with many models (run in loops). Is it possible to "turn off"
variables with dynamic many to many mappings?
```

Answer from mccarl@masked.tamu.edu:

I think the answer is yes. If you have a model that covers crops and is defined over several sets

```
growcrop(region,landtype,croptype,fertiliz,irrigat)
```

and one puts togeather a set telling when these cases are viable such as

```
set yescrop(region,landtype,croptype,fertuse,irrigat);

yescrop(region,landtype,croptype,fertiliz,irrigat)$
    (landarea(region,landtype) gt 0
     cropmix(region,croptype) gt 0
     and irrigat(crop)$sameas(irrigat,"irrigated")
     etc
     )=yes;
```

than one can set up a model like the collowing and the sums will automatically only cover the relevant cases of the crop

```
objective.. profits=e=
  sum(yescrop(region,landtype,croptype,fertiliz,irrigat),
     netrev(...)* growcrop(region,landtype,croptype,fertiliz,irrigat))

balance(region,commodity)=
  -sum(yescrop(region,landtype,croptype,fertiliz,irrigat),
     budgetdata(commodity,croptype,...)*
     growcrop(region,landtype,croptype,fertiliz,irrigat))
  +use(crop(region,commodity) =l=0;
```

one must be carful with this if one ever redefines the data in the formation of yescrop as the calculation must be repeated to update it. ialso one would need to reset all the entries to no

There is one other very evil feature of this (I call it a gams bug but that is a debatable point according to Alex) Namely under a conditional like this if one is solving in a loop and the variables are sometimes present and sometimes absent then the values of the variables will remain in the growcrop.l data even if the conditional has removed the variable. this happens even under solveopt=replace. We need to have solveopt=destroy

For grins try the following model
Note when i eliminate a variable with the limit conditional it does not go away from report write calculations unless I zero it my self This has caused me big headaches at times

```
option limrow=0
option limcol=0;
$offsymxref offsymlist
set varname /x1,x2,x3/
Parameter limit(varname)  /x1 1,  x2 1, x3 1/
          limit2(varname) /x1 1 , x2 1, x3 1/
variable           z  obj var
```

```
positive variables variablval(varname) variable values
                   secondvar(varname)  other variables;
equations          obj objective function
                   bound2(varname) bounds on secondvar
                   bound(varname) bounds via equations;
option lp=bdmlp;
obj.. z=e=sum(varname,variablval(varname)$limit(varname)
                     +secondvar(varname)$limit2(varname));
bound(varname)$limit(varname).. variablval(varname)=l=limit(varname);
bound2(varname)$limit2(varname).. secondvar(varname)=l=limit2(varname);
model try /all/
*option solprint=off;
parameter sol(*,*,*);
solve try using lp maximizing z;
sol("z","z","trybefore")=z.l;
sol("variablval",varname,"trybefore")=variablval.l(varname);
sol("secondvar",varname,"trybefore")=secondvar.l(varname);
sol("margbound",varname,"trybefore")=bound.m(varname);
sol("margbound2",varname,"trybefore")=bound2.m(varname);
limit2(varname)=0;
limit('x1')=0;
*solve two x1 is suppressed
solve try using lp maximizing z;
sol("z","z","tryaft")=z.l;
sol("variablval",varname,"tryaft")=variablval.l(varname);
sol("secondvar",varname,"tryaft")=secondvar.l(varname);
sol("margbound",varname,"tryaft")=bound.m(varname);
sol("margbound2",varname,"tryaft")=bound2.m(varname);
*note the darn value of x1 is still here but we are merging
display variablval.l;
display secondvar.l;
*so now we replace
option solveopt=replace
solve try using lp maximizing z;
*note the darn value of x1 is still here
sol("z","z","tryaftrep")=z.l;
sol("variablval",varname,"tryaftrep")=variablval.l(varname);
sol("secondvar",varname,"tryaftrep")=secondvar.l(varname);
sol("margbound",varname,"tryaftrep")=bound.m(varname);
sol("margbound2",varname,"tryaftrep")=bound2.m(varname);
secondvar.l(varname)=0;
*despiration i get reid of it myself
solve try using lp maximizing z;
sol("z","z","tryaftrep2")=z.l;
sol("variablval",varname,"tryaftrep2")=variablval.l(varname);
sol("secondvar",varname,"tryaftrep2")=secondvar.l(varname);
sol("margbound",varname,"tryaftrep2")=bound.m(varname);
sol("margbound2",varname,"tryaftrep2")=bound2.m(varname);
display sol;
```

1.61 NLP with real-power constraint

I am trying to solve a simple nonlinear maximization problem. One of the constraints is as follows.

```
 TRMTCP .. 2*X1**0.8 - X2 =L= 10;
```

But GAMS/MINOS gives me an error message that I have an undefined real power in this constraint. I thought the message is related to power 0.8. So, when I change it to 1, I got the solution. My question is how I can include the real power in the constraint?

Answer from erwin@gams.com:

Add X1.LO = 0.001; to your model. x1**0.08 is
in fact evaluated as exp(0.8*log(x1)).
Apparently GAMS is extremely smart in recognizing a power of 1 as a
special case.

1.62 Parameter declaration

I'm having some problems reading data into a parameter and would appreciate any suggestions. I just may not be thinking of the problem the right way. I have three fishing vessel types Gillnet, Trawl and Hook which I've declared as a set

```
Set I /gillnet, trawl, hook/
```

There are also two other sets J and K. I want to be able to read in data from an external file set up by a program where the user has the choice of selecting all vessel types, so I've declared a subset of I as follows:

```
ALLG(I) /gillnet, trawl, hook/
```

So the subset contains all the members of the original set.I then declare a parameter tempcl(I,J,K) and try to read in the following data:

```
/ALLG.139.6          1
 Gillnet.132.3       1
 Trawl.132.3         1
 Hook.132.3          1
 Trawl.131.3         1
 Trawl.131.4         1
/;
```

When I run GAMS, I get an error message $170 right below the ALLG.139.6 in the parameter file. This is a domain violation error, but I've declared ALLG to be a subset of I, so I'm not sure why it's occurring.

Answer from mccarl@masked.tamu.edu:

This error will occur here since allg is not a member of the allg set
only the words gillnet,trawl and hook are allowed

Answer from rwigle@wlu.ca:

In your example, ALLG is a SET, whereas GAMS wants a set ELEMENT.

I am not quite sure if this is like what you are trying to do, but consider the following example:

```
SET II / TOTAL, gillnet, trawl, hook /;

Set I(II) /gillnet, trawl, hook/

SETS J / 1*5 /,
     K / A, B, C / ;

PARAMETER NUMBERS(II,J,K)
/
  TOTAL.1.A   100
  GILLNET.2.C  10
  GILLNET.3.B  20
  trawl.1.A    60
  hook.1.A     40
/;
```

Answer from rrosenthal@monterey.nps.navy.mil:

You could do what you attempted with the following:

```
parameter tempcl(I,J,K);
tempcl( I, "139", "6" )  = 1;
tempcl( "Gillnet", "132", "3" )  = 1;
```

etc. You don't need the extra set ALLG to do this.

1.63 Loop / recursive dynamic CGE

My intention is to construct a recursive dynamic CGE over several time periods. If I understand the principle right, a recursive dynamic model can be built by looping over the same static model a couple of times, saving after each loop the data and using it as the "new" data starting-point for the next loop.

Now, how to program a time-loop is clear to me. What I don't know is how to get GAMS to save the data results from the previous loop in order to use them for the next loop. (In my looping attempts I had the impression that GAMS uses the original benchmark data as starting point in each loop again and again).

Answer from ricardo@isr.uni-stuttgart.de:

Try something like the following:

```
SETS

   K1 "Reactors" / 1 , 2 , 3 /
    J "Phases" / ORG , AQ /
  K1D(k1) "Dynamic reactor set"
;
```

```
VARIABLES

 VG1(J,K1) "partial volume, phase j, reactor k1" VR1(K1) "total
 volume, reactor k1" etc.
;

* * EQUATIONS DEFINITION *

 cc1(k1d) .. vr1(k1d) =e= sum(j,vg1(j,k1d));

* * solve a cascade of reactors *

 k1d(k1) = no;
 loop(k1,
       k1d(k1) = yes;
       solve ABC minimizing DEF using NLP;
       vg1.l(j,k1+1) = vg1.l(j,k1);
       vr1.l(k1+1) = vr1.l(k1);
       k1d(k1) = no;
      );
```

1.64 Error message

Could someone please tell me what this error message is telling me and how to fix it? It occurs on any model I try to run.

Error: could not create process directory: too many scratch directories exist.

Answer from thorntop@hoffman.army.mil:

My guess would be the Temporary directory that GAMS wants to write to already has too many numbered scratch files. Try cleaning up this directory or folder and re-run

Answer from ghosh@sscl.uwo.ca:

Delete all the subdirectories "225a...225z" from the directory you are working at and run your model. Hope this helps.

Answer from rutherford@colorado.edu:

As someone has already mentioned, this problem arises due to more than 26 orphaned scratch directories. These are created every time to Ctrl-C a GAMS job. I find that they build up pretty quickly, so I have encountered this error myself on several occasions.

Here are a couple of batch programs I use to clean up a GAMS model directory:

```
clear.bat:
@echo off
if exist 225a\nul call deltree 225a
if exist 225b\nul call deltree 225b
if exist 225c\nul call deltree 225c
if exist 225d\nul call deltree 225d
if exist 225e\nul call deltree 225e
if exist 225f\nul call deltree 225f
if exist 225g\nul call deltree 225g
if exist 225h\nul call deltree 225h
if exist 225i\nul call deltree 225i
if exist 225j\nul call deltree 225j
if exist 225k\nul call deltree 225k
if exist 225l\nul call deltree 225l
if exist 225m\nul call deltree 225m
if exist 225n\nul call deltree 225n
if exist 225o\nul call deltree 225o
if exist 225p\nul call deltree 225p
if exist 225q\nul call deltree 225q
if exist 225r\nul call deltree 225r
if exist 225s\nul call deltree 225s
if exist 225t\nul call deltree 225t
if exist 225u\nul call deltree 225u
if exist 225v\nul call deltree 225v
if exist 225w\nul call deltree 225w
if exist 225x\nul call deltree 225x
if exist 225y\nul call deltree 225y
if exist 225z\nul call deltree 225z


deltree.bat (only needed for NT  -- deltree is not a system command!):
@echo off
if not exist %1\nul goto syntax
rmdir /s /q %1
goto end

:syntax
echo    Not a subdirectory: %1?

:end
```

1.65 Spot an error

I have a batch file (i.e. file.bat) with the following content:

```
gams filename.gms s=s1
gams sim1 r=s1 s=s2
gams sim2 r=s2 s=s3
gams sim3 r=s3 s=s4
gams sim4 r=s4 s=s5
gams sim5 r=s5 s=s6
gams sim6 r=s6 s=s7
```

I run the model and the simulations as follows

```
c:>file
```

When I check the results, it happens that there is a mistake.

Here is the question:

Do you know an automatic way to figure out where exactly (i.e at sim2, sim3, sim4...) the algorithm could not solve?

Answer from mccarl@masked.tamu.edu:

You should have files sim1.lst sim2.lst etc Go through them and look for s o l v or i believe the words solver status
In the future you could do the following parameter solveres(*) saved solver result flags; then in your code

```
solve mymod using lp maximizing it
solveres("sim1")=mymod.modelstat;
```

which in turn can be displayed at any time and has the values explained on pages 117-118 of red or blud manuals
(section 10.3 solve summary

Answer from leibypn@ornl.gov:

1) There are many ways to determine model success for multiple runs. The most straightforward is of course to look in the listing files, simn.lst, for n=1..6. Near the text string "S O L V E" it will report solver status. Bruce McCarl's solution is better, yet still simple to implement. It does require that the successive model solutions are "chained," each departing -from the save files of the previous solution. In this way the modelstatus values stored in parameter "solveres(*)" for each solve are saved and accumulated as runs are added.

2) A more automated (and more cumbersome) way to summarize model results for multiple solves could use the following BAT files and GAMS post-processing utility:

```
        SUMMRUNS.BAT  (repeatedly calls SUMM!RUN.BAT)
        SUMM1RUN.BAT  (calls GAMS on SUMM1RUN.bat)
        SUMM1RUN.GMS  (gets info from a gams run)
```

(this draws on ideas posted at GAMS web site and other exchanges on the GAMS-L list. Perhaps parts of it will be useful to others):

Note: I altered your bat file (file.bat) slightly so that the save files named "sn.g0*" correspond to the solution of "simn"):

```
* Snip HERE **************************

REM Beginning of file.BAT **************************
REM this executes 7 versions of the model "mymodel," chaining one solution
after another
  gams sim.gms s=s0
  gams sim1 r=s0 s=s1
  gams sim2 r=s1 s=s2
  gams sim3 r=s2 s=s3
  gams sim4 r=s3 s=s4
  gams sim5 r=s4 s=s5
  gams sim6 r=s5 s=s6
```

```
REM End of file.BAT **************************

* Snip HERE **************************

REM Beginning of SUMMRUNS.BAT **************************
REM SUMMRUNS reports a summary of multiple model runs listed below (s1 ...
s6).
REM Call SUMMRUNS from DOS command line, with 1 argument,
REM  that argument being the name of the summary file to create: e.g.
SUMMRUNS simrslts
REM  Note: extension for summary file name will be .txt
CALL SUMM1RUN s0 %1
CALL SUMM1RUN s1 %1
CALL SUMM1RUN s2 %1
CALL SUMM1RUN s3 %1
CALL SUMM1RUN s4 %1
CALL SUMM1RUN s5 %1
CALL SUMM1RUN s6 %1
REM End of SUMMRUNS.BAT **************************

* Snip HERE **************************

REM Beginning of SUMM1RUN.BAT  **************************
:==>summnote.bat    This routine executes a GAMS postprocessing routine
SUMM1RUN.GMS
@echo off
REM Call with 2 args:
REM    %1 name of run to be summarized (w/o extension),
REM    %2 name of outputfile (also w/o extension)
REM E.g.:  SUMM1RUN runname notefile
REM first check to see if input and output file names specified.
if a%1==a goto syntaxerr
if a%2==a goto syntaxerr

REM second check for existence of output file.  If not, then write header line
if exist %2.txt goto addinfo
echo Date           Time           Run#                Model Stat    Solve Count      Objective  > %2.txt
:addinfo
if not exist %1.g01 goto nofileerr
REM Get source "runname" from command line and put into temp.gms file
REM Get destination "notefile" from command line and put into temp.gms file
REM (temp.gms file is a way to pass strings to GAMS, and in this case the
strings are filenames)
echo $setglobal runname %1 > temp.gms
echo $setglobal notefile %2 >> temp.gms
call gams summ1run r=%1 s=summ1run
goto end

:syntaxerr
echo    Syntax:  summnote runname notefile
goto end
:nofileerr
echo    File not found: %1.g01
echo    File not found: %1.g01 >> %2.txt
```

```
:end
REM End of SUMM1RUN.BAT  **************************

* Snip HERE **************************

* Beginning of SUMM1RUN.GMS  **************************
*  This report post-processing file provides solution diagnostics a given model run.
*  It assumes:
*   1. there exists a file temp.gms which specifies the source model runglobal "runname"
*      and the destination file as global you have run a gams file whose
*   2. Exist a set of files runname.g01 .. runname.g06 which chronicle the
*      solution of a particular case for a model named "mymodel"
*   3. The objective of the model "mymodel" is a variable "obj"
*   4. somewhere in the model mymodel is the definition of a parameter called "solvecount".
*      One way to do this is to define and initialize solvecount=0 in the first version of
*      the model, sim.gms.  Subsequent versions (sim1.gms - sim6.gms) simply increment
*      solvecount _after_ the call to Solve.  I use solvecount to count the number of
*      times solve must be called to achieve convergence.  That is,
*      sometimes in my models that restart GAMS from the endpoint
*      of a prior solution, I have the following test:
*         IF (mymodel.MODELSTAT GT 2,
*           SOLVE mymodel MAXIMIZING OBJ USING NLP;
*           MODELSTATS(TLAST,"DYNAMIC") = mymodel.MODELSTAT;  { save model status }
*           SOLVECOUNT = SOLVECOUNT+1;
*           { Increment count of solve calls, to track solution success }
*         ); {if MODELSTAT GT 2}
*      You may find some other use for solvecount.
*

$OFFSYMLIST OFFSYMXREF
$OFFUPPER
* get global "runname" from command line and put into temp.gms file
$include temp.gms
FILE tempjunk /%notefile%.txt/;
tempjunk.ap = 1;
tempjunk.nw = 15;
tempjunk.tw = 15;
PUT tempjunk;
* write out solution date and time, modelname, modelstatus and solvecount
PUT  system.rdate, system.rtime ,;
PUT   '%runname%',;
PUT  mymodel.MODELSTAT ::0, SOLVECOUNT ::0, obj.L ::3;
* End of SUMM1RUN.GMS  **************************

* Snip HERE **************************
```

1.66 Endogeneous relational operations

I am looking for a way to express the following simplified mathematical relationship in a GAMS program:

```
y = a + b * x for x <= z and
y = c + d * x for x > z
```

where a, b, c, d are constants and
y, x, and z are variables.

As GAMS does not allow variables in dollar logical conditions etc.
I am somehow stuck with this problem.

Answer from mccarl@masked.tamu.edu:

You cannot do this with lp. You need to use a binary variable from
integer programming also the strict inequality gives problems

```
x -m*q2<= z
x -m*q1>= z+0.0001
y<= a+bx +m*q1
y>= a+bx -m*q1
y<= c+dx +m*q2
y>= c+dx -m*q2
q1+q2=1
q1,q2 binary integer variables
```

m is a large number ie 9999999 > expected value of z.

You could possibly just solve the model twice

1.67 Suppressing the output listing

I have a GAMS 'program' which solves potentially thousands of small LPs
(using OSL). My problem is that, even with the following options...

```
$offlisting
$offsymxref offsymlist
option  limcol = 0;
option  limrow = 0;
option  solprint=off;
```

I get an enormous .lst file. I also suspect that the execution time
is substantially increased by so many writes to disk. My main objective
is to reduce execution time so i don't want to fool around with deleting
the .lst file every so often.

Can anyone suggest how i can prevent the creation of the .lst file. All
i really want is for GAMS to shut-up and do its job, not tell what it is
doing ;)

Answer from tor@sam.sdu.dk:

Use the -o /dev/null, i.e gams <modname> -o /dev/null on a UNIX,LINUX etc.
or gams <modname> -o NUL on a PC platform

Answer from rutherford@colorado.edu:

I typically configure GAMS with suppress=1 which omits the source code
echo print. It is then necessary to invoke GAMS with

```
        gams model suppress=0
```

in order to generate a listing.

option solprint=off; suppresses the solver listings.

I previously believed that turning off output would speed a program which solved many cases, but when I actually checked this it did not save any significant time. It still might make sense to reduce output to the listing file in order to make it easier to load the file in your editor.

Under DOS I used a RAM disk to dramatically improve run times, but this is neither necessary nor helpful under W95/NT where disk caching is built into the OS.

1.68 Stopping the iteration process

I'm working with a model which is run with two set of loop iterations. One set of iterations is over a grid of parameter values; the other is to solve the model successively until convergence to an equilibrium is achieved. For the majority of parameter value combinations (1st loop), it turns out that the model needs only a few iterations(2nd loop) to coverge, but for some parameter combinations achieving convergence requires a far larger number of iterations.

My problem then is this: Say that I set the number of iterations in my second loop equal to 50. Since for the majority of parameter combinations the model needs no more than 10 iterations to converge, and the number of combinations I have runs in the thousands, I don't want Gams to do 50 iterations all the time. Can I tell Gams to stop once a specified covergence critirion has been met, and go back to the first loop? How?

Answer from vlampe@agp.uni-bonn.de:

You should use the WHILE statement or include a $-operator with your loop. Suppose your critical value to check is tval, and the critical level tval should be lower than is tvallim, iterset be your set to run the second loop on, and count represents a simple counter (you should use it with the while statement to avoid infinite loops in case of missing convergence), then write either

```
tval = 1.e+4;
count = 0
WHILE( (count<1000 AND tval>tvallim),
   count = count + 1;
   ... solve yourmodel using ...
   tval = whatever your criterion is;
);
```

or

```
tval = 1.e+4;
LOOP( iterset$(tval>tvallim),
   ... solve yourmodel using ...
   tval = whatever your criterion is;
);
```

Answer from rrosenthal@monterey.nps.navy.mil:

I am enclosing a GAMS implementation of Benders Decomposition Algorithm, which has the same property of requesting an unknowable number of solves-in-a-loop. Lisandro, I hope this is helpful. It may be more than you need, but I hope others will see some useful lessons. The most difficult and critical thing for me to learn was the need to use of a static set to declare the CUT equations and a dynamic set to define them. I have made several attempts at Benders implementations over the years. This is the first version that I think is worth sharing. (If you use this, be sure to add a smarter, context-specific starting solution than all zeroes as done here. Also add instrumentation to report progress at each iteration. I left the progress report out of this mailing because it takes more code than the mere 15 lines it took to implement Benders algorithm.) As usual, VARIABLES and EQUATIONS are in upper case, parameters in lower case. By the way, Benders decomposition is very popular among energy modelers. Do you economic modelers out there use it?

```
Sets
   i   origin bases
   j   destination bases
   k   commodities
   h   Benders iteration counter        / 0 * 999 /
   hnow(h)  cuts in current Benders master problem
   ;

...  Data entry skipped  ...

Positive Variables
   X(i,j,k)    tons of cargo k delivered from i to j
   NOGO(j,k)   tons of demand not delivered
   ;
Integer Variables
   Y(i,j)      number of aircraft flying from i to j
   ;
Free Variables
   LPOBJ       LP objective variable
   MIPOBJ      MIP objective variable
   ;

*Override default upper bound of 100 on general integers.
  Y.UP(i,j)  =  ceil( sum(k,dem(j,k)) / cap(i,j) ) ;

EQUATIONS
*Benders LP:
   LPOBJDEF
   SUPPLY(i,k)
   DEMAND(j,k)
   CAPACITY(i,j)
*Benders MIP:
   CUT(h)
   ;

* Integer variables are fixed to current Y.L in LP definition.
LPOBJDEF..
    sum( (i,j,k), c(i,j,k) * X(i,j,k) )
  + sum( (j,k), pen(j,k) * NOGO(j,k) )
  + sum( (i,j), f(i,j) * Y.L(i,j) )
  =E= LPOBJ ;

SUPPLY(i,k)..
  sum( j, X(i,j,k) )  =L=  sup(i,k) ;

DEMAND(j,k)..
  sum( i, X(i,j,k) ) + NOGO(j,k)  =G=  dem(j,k) ;

CAPACITY(i,j)..
  sum( k, X(i,j,k) )  =L=  cap(i,j) * Y.L(i,j) ;

* Cuts are declared with static set h and defined with dynamic set hnow.
* One new cut is added each Benders iteration.

CUT(hnow)..
  MIPOBJ
  =G=
    sum( (i,j), f(i,j) * Y(i,j) )
  + g(hnow)
  - sum( (i,j), pi(hnow,i,j) * cap(i,j) * Y(i,j) ) ;

model BENDERSLP / lpobjdef, supply, demand, capacity / ;
model BENDERSMIP / cut / ;

* Initialize Benders Decomposition Algorithm
Y.L(i,j) = 0 ;
ub = +inf ;
lb = -inf ;
tolerance = -inf ;

* Benders Iterations
loop( h $( ub-lb > tolerance),
   hnow(h) = yes ;
*  Solve LP Subproblem for possible UB improvement
   solve BENDERSLP using lp minimizing LPOBJ ;
   if( LPOBJ.L < ub,
      ub = LPOBJ.L ;
      ybest(i,j) = Y.L(i,j) ;
      tolerance = abs(tolpct*ub) ;
   ) ;
   if( ub-lb > tolerance,
*     Create coefficients for new Benders cut and solve MIP
      pi(h,i,j) =  - CAPACITY.M(i,j) ;
      g(h) =   sum( (i,k), sup(i,k) * SUPPLY.M(i,k) )
             + sum( (j,k), dem(j,k) * DEMAND.M(j,k) ) ;
      solve BENDERSMIP using mip minimzing MIPOBJ ;
      lb = MIPOBJ.L ;
   ) ;
) ;

* Final solve for optimal continuous variables if current Y not optimal.
Y.L(i,j) = ybest(i,j) ;
solve BENDERSLP using lp minimizing LPOBJ ;
```

Answer from adrud@arki.dk:

```
We have seen a few comments about loop or while for stopping an iterative
process. I would like to add a warning to the while approach based on the
following example. I have added a few lines to the well known Ramsey
model, and I have added an illegal options file for CONOPT2 (just one
line with "Junk = 5"):

 ramsey.optfile = 1;
 option nlp = conopt2;
 scalar count / 0 /;
 scalar continue / 1 /;
 while( continue,
   Solve ramsey maximizing utility using nlp;
   count = count + 1;
   continue = (ramsey.numinfes > 2 or count < 4);
   display ramsey.numinfes, ramsey.modelstat, ramsey.solvestat,
           count, continue;
  );

If the Solve happens to abort (which I have forced it to with the bad options
file), then ramsey.numinfes gets the value NA = not available. In the GAMS
universe, NA or-ed with an expression is still NA so the scalar "continue"
becomes NA. And NA is considered nonzero in a test so the while statement will
run forever (until your disk is full with a very long listing file!).

The loop method with a condition is safer since the number of elements in the
loop set limits the damage. With the following construct you will get a long
listing file, but GAMS will stop. After the error in the first SOLVE, GAMS
will not generate the model again, so the overall execution time is very
small:

 scalar count / 0 /;
 scalar continue / 1 /;
 set lset loop st / l1*l200 /;
 loop(lset $ continue,
   Solve ramsey maximizing utility using nlp;
   count = count + 1;
   continue = (ramsey.numinfes > 2 or count < 4);
   display ramsey.numinfes, ramsey.modelstat, ramsey.solvestat,
           count, continue;
  );
```

Chapter 2

Solver related Questions

2.1 General Questions

2.1.1 Changing Solvers

I would like to know how to change gams default solvers (e.g., use CONOPT instead of MINOS for NLPs).

Answer from n.n:

Changing the default solver can be done in two different ways:

1) Run gamsinst again and change the default solver there. This changes will be permanent.

2) Add a line to your model (before the current solve statement) to switch your solver:

```
     option nlp=conopt
```

This change will only be valid for this particular model and can be changed using another option. Switching back to the default solver (from the installation routine) can be done using option nlp = default. Please check also section D in the user' guide for more information.

2.1.2 Using different option files

I have been using GAMS for some time now, but for highly non-linear problems it is sometimes hard to find the optimum. Therefore, I would like to make an option file to adjust some of the default options. However, after trying all the different names mentioned in the reference guide and studying the different help files that were supplied with the version of GAMS I use, I could not get GAMS to recognize the option file. Has anybody on this list had this problem before, or does anybody know how to solve it. If somebody has used an option file could he then please send me an example of it.

Answer from n.n:

I agree with those who say it is easy to get things mixed up concerning solver option files. For example, you might end up using an option file unintentionally. Part of the problem is that the option file name has nothing to do with the model file name. My colleague Jerry Brown had a great idea for managing your solver option file: generate it on the fly as needed with put commands. I use this trick often now. If this method doesn't keep you out of trouble, I don't know what will. Here is an example from an agricultural problem:

```
< sets, parameters, variables, equations ... >
Model harvest /all/;
* Create solver options file:
harvest.optfile = 1 ;
file optfil /cplex.opt/;
put optfil ;
put "* CPLEX.OPT CREATED FOR HARVEST OPTIMIZATION MODEL";
put / "* Created on:", system.date, " ",
system.time ;
put / "presolve 0" ; {Do not perform presolve.}
put / "writesos" ; {Write SOS file.}
putclose ;
Solve harvest maximizing profit using mip;
```

2.1.3 Solution of infeasible subproblems

I am interested in applying a decomposition method for the solution of an inventory problem. The method involves solving an iterated sequence of master and sub-problems. At any iteration, a given subproblem may be infeasible (i.e. the dual of the subproblem is unbounded, i.e. there exists one or more extreme directions in the defining polytope). My question is this: Having solved a problem in GAMS and found it to be unbounded, is it possible to determine the extreme direction vector? Specifically for the problem

```
positive variables
mu1
mu2
mu3
mu4;
variables
pi1
pi2
tot;
equations
e1
e2
e3
e4
cost
cost1
cost2;
e1.. 0.025 - pi1 + mu1 =G= 0;
e2.. 2.5 + pi1 + mu2 =G= 0;
e3.. 0.025 - pi2 + mu3 =G= 0;
```

```
e4.. 2.5 + pi2 + mu4 =G= 0;
cost.. tot =e= -90*pi1 + 50*pi2 + 100*(mu1+mu3) + 40*(mu2+mu4);
model test /e1,e2,e3,e4,cost/;

solve test minimizing tot using lp;
```

which gives the (annotated) output:

```
S O L V E S U M M A R Y
MODEL TEST OBJECTIVE TOT
TYPE LP DIRECTION MINIMIZE
SOLVER BDMLP FROM LINE 38
**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 3 UNBOUNDED
**** OBJECTIVE VALUE -125.0000
EXIT -- PROBLEM IS UNBOUNDED.
LOWER LEVEL UPPER MARGINAL
---- EQU E1 -0.025 2.500 +INF .
---- EQU E2 -2.500 -2.500 +INF EPS
---- EQU E3 -0.025 12.500 +INF .
---- EQU E4 -2.500 -2.500 +INF EPS
---- EQU COST . . . EPS
LOWER LEVEL UPPER MARGINAL
---- VAR MU1 . . +INF EPS
---- VAR MU2 . . +INF EPS
---- VAR MU3 . . +INF EPS
---- VAR MU4 . 10.000 +INF .
---- VAR PI1 -INF -2.500 +INF .
---- VAR PI2 -INF -12.500 +INF .
---- VAR TOT -INF . +INF 1.000 UNBND
**** REPORT SUMMARY : 0 NONOPT
0 INFEASIBLE
1 UNBOUNDED (UNBND)
```

how would I determine the unbounded direction vector

```
x = (mu1,mu2,mu3,mu4,pi1,pi2) = (0,0,0,1,0.025,-1)
```

from the output i.e. any kx, k > 0 satisfies the equations, and k may get arbitrarily large but still satisfies the feasibility conditions?

Answer from adrud@arki.dk:

This describes a problem of identifying unbounded rays for use in a decomposition algorithm. Unfortunately, the information is not available. Most LP algorithms will identify the situation and the extreme ray is known internally, but the move is never made, and neither the "solution" nor the duals are very useful for further computations. All GAMS LP solvers (BDMLP, CPLEX, MINOS5, OSL,...) behave the same although both primal and dual solution may be different, so the problem is a generic problem. However, you may try to work on the dual problem. If the original problem was unbounded the dual will be infeasible, and the final point represents some "minimum sum of infeasibility", and the duals will be relative to this objective. This means that the dual variables of the dual problem are well defined, and they represent the extreme ray of the unbounded primal problem. The dual of the example is:

```
positive variables
    e1
    e2
    e3
    e4
    tot;
equations
    mu1
    mu2
    mu3
    mu4
    pi1
    pi2
    cost;
mu1 .. e1 =l= 100;
mu2 .. e2 =l= 40;
mu3 .. e3 =l= 100;
mu4 .. e4 =l= 40;
pi1 .. -e1 + e2 =e= -90;
pi2 .. -e3 + e4 =e= 50;
cost.. tot =e= -0.025 * e1 - 2.5 * e2 - 0.025 * e3 - 2.5 * e4;
model infeas / mu1, mu2, mu3, mu4, pi1, pi2, cost /
solve infeas maximizing tot using lp;
```

which results in the following output (extract):

```
S O L V E S U M M A R Y
MODEL INFEAS OBJECTIVE TOT
TYPE LP DIRECTION MAXIMIZE
SOLVER BDMLP FROM LINE 31
**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 4 INFEASIBLE
**** OBJECTIVE VALUE -127.5000
RESOURCE USAGE, LIMIT 0.050 1000.000
ITERATION COUNT, LIMIT 2 1000
BDM - LP VERSION 1.01
EXIT -- PROBLEM IS INFEASIBLE.

LOWER LEVEL UPPER MARGINAL
---- EQU MU1 -INF 100.000 100.000 EPS
---- EQU MU2 -INF 10.000 40.000 .
---- EQU MU3 -INF . 100.000 .
---- EQU MU4 -INF 40.000 40.000 1.000
---- EQU PI1 -90.000 -90.000 -90.000 EPS
---- EQU PI2 50.000 40.000 50.000 -1.000 INFES
---- EQU COST . . . EPS
LOWER LEVEL UPPER MARGINAL
---- VAR TOT -INF -127.500 +INF .
```

```
---- VAR E1 . 100.000 +INF .
---- VAR E2 . 10.000 +INF .
---- VAR E3 . . +INF 1.000
---- VAR E4 . 40.000 +INF .
**** REPORT SUMMARY : 0 NONOPT
1 INFEASIBLE (INFES)
SUM 10.000
MAX 10.0001111111
MEAN 10.000
0 UNBOUNDED
```

and the extreme ray is

```
(mu1.m,mu2.m,mu3.m,mu4.m,pi1.m,pi2.m) = ( eps , 0.0 , 0.0 , 1.0 , eps , -1.0)
```

2.1.4 Scaling Strategies

I am using OSL and might be having scaling problems. GAMS allows one to select scale factors for both row and column. As I understand scaling, I pick scaling factors for each row and variable, set model.scaleopt=1, and I'm done. Perhaps, I have to turn some scaling within OSL off, but that's it. I'm trying to get my hands dirty enough to understand intelligent manual strategies. Dividing through by the max in absolute value is one way but I thought there might be a better way. This turned out to be more difficult than I thought. If anyone has worked out some fairly general purpose ad hoc procedures and/or strategies, I'd be interested in finding out what they are. Let a(i,j) be the coefficients of the matrix. The goal is to find scale factors b(i) and c(j) so that the non-zero elements of the resulting matrix are close to each other. Using NLP for this is impractical (in spades), but I'm just experimenting right now. This first effort tries to get the matrix elements close to 1.0. One should probably bound the weights away from zero and look for a relative error from a variable that is also part of the optimization, as well as consider negative coefficients, blah, blah, blah, but I'm not going to worry about those considerations yet.I started out with a raw matrix of:

```
    C1 C2 C3
R1 1.0 2.0 3.0
R2 0.5 0.0 3.0
```

```
C1    C2    C3
R1 1.138 1.000 0.805
R2 0.805 1.138
```

which is pretty good; the objective value is 0.1144. This used initial guesses of 1.0 or 0.1. Starting with initial weights of 0.0 gives a terrible answer because of the vanishing derivative. I was just wondering if anyone as had any practical experience with scaling.

```
SETS I /R1*R5/, J /C1*C5/;
```

```
TABLE A(I,J)
C1 C2 C3
R1 1.0 2.0 3.0
R2 0.5 0.0 3.0
;
SETS IJ(I,J);
IJ(I,J) = YES$ (A(I,J) NE 0.0);
POSITIVE    VARIABLE
B(I),
C(J),
D(I,J);
VARIABLE    E;
EQUATION    SQ_(I,J),
R;
SQ_(IJ(I,J))..  D(IJ) =E= (B(I)*C(J)*A(IJ)-1.0) * (B(I)*C(J)*A(IJ)-1.0);
R..         E =E= SUM(IJ, D(IJ));
MODEL GOOD_SCALE /ALL/;
B.L(I) = 0.0$ (SUM(IJ(I,J), 1.0) GT 0.0);
C.L(J) = 0.1$ (SUM(IJ(I,J), 1.0) GT 0.0);
D.L(IJ(I,J)) = (B.L(I)*C.L(J)*A(IJ)-1.0) * (B.L(I)*C.L(J)*A(IJ)-1.0);
E.L = SUM(IJ, D.L(IJ));
OPTION NLP=CONOPT;
SOLVE GOOD_SCALE USING NLP MINIMIZING E;
DISPLAY IJ, A;
DISPLAY B.L, C.L, D.L, E.L;
PARAMETER AA(I,J);
AA(IJ(I,J)) = A(IJ)*B.L(I)*C.L(J);
DISPLAY AA;
```

Answer from n.n:

There is quite some literature on scaling, both in general and for LP models. One reference to get you started is Tomlin, J.A.: "On Scaling Linear Programming Problems", Mathematical Programming Study, vol 4 1975, p 146-166. It contains several algorithms plus some discussion. The NLP model in your note has a problem. It is perfectly possible for scale factors to become zero. One standard way to get around the problem is to use an objective like min sum(sqr(log(b*a*c))) which tries to move the scaled entries close to 1, but eviations are measured in a logarithmic scale so 0.5 and 2.0 are considered equally good (or bad). In practice you would not implement the objective directly with the logs but instead use the identity log(b*a*c) = log(b) + log(a) + log(c). Log(b) and log(c) are your new variables and log(a) is evaluated once. So your model becomes min sum(sqr(Lb + La + Lc)) where La are data and Lb and Lc are free variables. The scaled matrix is A scaled = A * exp(Lb) * exp(Lc).

2.1.5 Strategies for Restarting models

We are modeling a dynamic market equilibrium model as an NLP. (Most of the nonlinearities are in the objective function). It is a fairly large model, and at best solved in a hour or so on a fast (300 Mhz) Pentium II system. Posting it here would not be practical (or pleasant for you, our colleagues!). It has solved scores of times over the last couple years, but returns to intransigence each time we re-benchmark the base case conditions.

Our problem is that the model frequently fails to converge. We typically get a final message of the form:

```
  **** SOLVER STATUS     4 TERMINATED BY SOLVER
  **** MODEL STATUS      7 INTERMEDIATE NONOPTIMAL
  ...
  ** Feasible solution. The tolerances are minimal and
     there is no change in objective although the reduced
     gradient is greater than the tolerance.
```

We have adopted 4 strategies for dealing with this Problem:

1) Choosing the "best" starting point we can, i.e., the base case solution prior to solving a policy case. \item For Minos5, fooling with the number of minor iterations per major iteration (40 or 60). We use the options file (Minos5.opt):

```
        BEGIN GAMS/MINOS options
        MAJOR ITERATIONS 5000
        * MINOR  ITERATIONS 60
        SUPERBASICS LIMIT 4000
        END GAMS/MINOS options
```

2) Switching between the solvers: Minos5, Conopt, and Conopt2.

3) Automated restarts from the stopping point of the last stalled/incomplete solution, solving in a loop until MODELSTAT = 2 or 1, or until "maxsolves" tried.

Usually some combination of these strategies will work, but not always.

The manual suggests another option: Rescaling We have not had success with rescaling. The problem here seems to be in the automated choice of reasonable scaling values. A variable which might be small in one run (e.g. demand of fuel f = 0.001) could be large (10.0) in another, given a different policy.

Answer from m.saunders@auckland.ac.nz:

Sorry the solvers are not 100% reliable. They never will be, but we can try to increase the chances. With Minos5 and nonlinear constraints, I think infeasible subproblems are the worst difficulty (sometimes cured by putting in your own penalized slacks on certain judicious rows).

You might try raising the Penalty parameter to 10.0 (default 1.0).

I mostly wanted to comment on scaling:

>We have not had success with rescaling. The problem here seems to be in
>the automated choice of reasonable scaling values. A variable which might

>be small in one run (e.g. demand of fuel f = 0.001) could be large (10.0)
>in another, given a different policy.

Where is the automation? -- in your generation of the model or in Minos5? (The Minos default Scale option 1 scales linear rows and columns.)

Let's assume you are in charge of scaling. Any given variable like f could have varying values as you say. Typically such a variable is a member of a set of similar variables. I think the best aim is to scale the set as a WHOLE (with just one number!) so that "typical variables in the set" are in a reasonable range. It doesn't matter if some of the them have tiny values -- that's inevitable. You wouldn't want to scale such values individually.

The same would apply to a set of constraints -- choose a single number to scale the whole set.

Of course, Minos would choose different scales for every row and column because it no longer knows about the sets. If you can choose your own scales well enough, it would be best to turn scaling off (Scale option 0).

Sorry if f is a unique variable in your case (not one of a set!).

2.1.6 Funny results from the simplex methods

While I was playing around with GAMS I encountered the following "problem" in a simple LP problem (I attached the .gms-file with the model in which this problem occured).

I will specify what I mean by "problem", but first this: 1) In the model, the feasable region is a pyramid with vertices (-1,-1,-1), (1,1,-1), (-1,1,1), and (1,-1,1) (a tetrahedron?). 2) The linaer function defined on this region takes on his minima on the plane through the vertices (1,1,-1), (-1,1,1), and (1,-1,1). 3) I had CPLEX, OSL or whatever solver minimize this function, with the use of the simplex method (just LP, not MIP).

The "problem" is the following: The answer that GAMS returns, (1,0,0), is a minimum indeed, but is not a vertex! So "GAMS" does find the minimum value of the function, but I think it should find one of the vertices in the "minimum plane", for a simplex method is used. This simple case is probably not the only case in which this happens.

(It is really getting a problem if you need the solution to be an integer solution and the only ones that are integer are the vertix solutions.)

Now I don't know if it is because of GAMS or because of the solvers, but maybe you can give me a possible explanation for this, in my opinion, "wrong" solution.

Even if my starting point is an optimal vertex, the solver finds
(1,0,0) as optimal solution!

```
* S3

SETS M /1*4/ N /1*3/;

TABLE V(M,N)
   1  2  3
1 -1 -1 -1 2 -1 1 1 3 1 -1 1 4 1 1 -1
  ;

VARIABLES
 S
 X(N);

EQUATIONS SEQ SIMP;

SEQ.. S =E= SUM(N, V('1',N)*X(N) );

SIMP(M).. SUM(N, V(M,N)*X(N) ) =G= -1;


MODEL S3 /ALL/;

OPTION LP=OSL;

OPTION LIMROW=25;

SOLVE S3 USING LP MINIMIZING S;

DISPLAY X.L;
--------------------------------------------------------------------------------* S3
```

```
SETS M /1*4/ N /1*3/;

TABLE V(M,N)
   1  2  3
1 -1 -1 -1 2 -1 1 1 3 1 -1 1 4 1 1 -1
  ;

VARIABLES
 S
 X(N);

EQUATIONS SEQ SIMP;

SEQ.. S =E= SUM(N, V('1',N)*X(N) );

SIMP(M).. SUM(N, V(M,N)*X(N) ) =G= -1;


MODEL S3 /ALL/;
OPTION LP=OSL;
OPTION LIMROW=25;
SOLVE S3 USING LP MINIMIZING S;
DISPLAY X.L;
```

Answer from adrud@arki.dk:

Free variables (all you X-es are free) are often treated in a special way in a simplex algorithm. They are started at zero and if they always have a reduced cost of zero they may stay at zero. So a non-basic free variable is allowed to be zero in the optimal solution. Otherwise you would have to make all free variables basic, and there may be too many free variables (more than the number of constraints) or their columns may be linearly dependent.
In your case you can get the 'expected' solution by adding redundant lower bounds on all the free variables (x.lo(n) = -2;). The variables are no longer free and they cannot be nonbasic at the value zero.

2.1.7 Infeasibility

I have big trouble due to infeasibilities in my model. These occurs from many constraints. To escape infeasibilities, I used artificial variables such as YPLUS and YMINUS. I could get optimal solution with these artificial variables. However, it leads to unrealistic results. Do you have any suggestions how I can handle the infeasibilities without changing my constraints in the model?

Answer from adrud@arki.dk:

Adding artificial variables (with some penalty weight) can sometimes result in a more reliable model and shorter solution times, so your approach seems reasonable. However, if the artificial variables have a nonzero level in the optimal solution then you cannot use the solution. Is this what happens? If so, mayby your model is really infeasible, and you need to change the constraints. You mention that you get unrealistic solutions, but what does that mean?

2.2 MIP-solver

2.2.1 Special Ordered Sets

Can anyone tell me how to make GAMS see SOS1 variables as binary 0-1? If I have the BINARY declaration together with the SOS1 declaration, GAMS complains with an error. Is there a special trick I can use? Therefore the structure I would like to have in the model is:

```
SUM(VINDEX, SOSVAR(M,VINDEX)) =L= 1; for every M
```

Now, the reason I cannot have the above constraints as =E= is that for some M, all the SOSVAR may be zero. I have no way to know a priori which these M will be. The model will decide that during the solution process.

Answer from adrud@arki.dk:

What about the following combination of SOS1 and BINARY variables:

```
SOS1 variable sosvar(m,vindex) Binary variable bin(m);

equation sosconstr(m);

sosconstr(m) .. sum(vindex, sosvar(m,vindex) ) =E= Bin(m);
```

where Bin(m) is used in your other constraints to determine whether all the corresponding SOS1 variables are 0 or one of them is 1.

Answer from n.n:

Your answers to my SOS set and your suggestions, triggered changes in my model that had the following impact:

```
Before:  After 86,000 seconds of B&B time (24hrs), I had a
 -------  solution with a value X that was still proven 33%
          away from optimality.

  After:  Within 700 seconds of SOS branching I had the optimal
  ------  solution whose value turned out to be about 5% better
          than X.
```

That tells a lot about the power of SOS branching. It also tells a lot about the power of those who are not inhibited or afraid to share their knowledge and experience.

2.2.2 Marginal Values in MIP-Poblems

What is the meaning of marginal price in GAMS output for a MIP?

Answer from n.n:

Marginals (shadow prices/reduced costs) on MIPS are a fuzzy topic. In many cases the marginal values are useful. The duals (marginals) reported for MIPS are not always the same when switching MIP solvers (manly for some internal technical reasons of convenience). GAMS reports consistent marginals across any MIP code used by GAMS. The idea is very simple. Fix all the discrete variables at the integer values and solve the resulting LP. Now you have a clear definition and can use it for your analysis. If you pick a different integer solution, you will, of course, get a different dual solution.

2.2.3 What is the default upper bound for an integer variable?

I have a model with large integer variables but I dont manage to get a level greater than 100- why?

Answer from n.n:

For some historical reasons, GAMS sets an upper bound of 100 for integer variables. The user has to reset the upper bound, i.e.:

```
x.up(i) = 1000;
```

However, formulating variables, which can take such huge numbers, as integer variables is in general.not reasonable.

2.2.4 Non integer results in a integer model

I'm trying to minimize the number of aerial tankings for a model under development, and I'm getting results I don't expect. Though I define a variable to be integer , the output file shows values that they aren't (eg, 0.194, etc).

Answer from n.n:

You may want to check if the solution status. It must read something like:

```
**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 8 INTEGER SOLUTION
**** OBJECTIVE VALUE 2730.0992
```

If optimal the model status may say OPTIMAL. If this is not the case check the number of iterations or the resource usage (old fashioned term for CPU time):

```
RESOURCE USAGE, LIMIT 92.933 1000.000
ITERATION COUNT, LIMIT 4843 10000
```

If it hit one of these limits before finding an integer solution you will get fractional values.

2.2.5 Error message from GAMS/ZOOM: Node table is full

Answer from n.n:

This message is generated by ZOOM, an old MIP solver that is not sold any more. ZOOM has difficulties to find a global solution for one of the MIP subproblems. You should experiment with other MIP solvers. (all of them are available in demo mode).

2.2.6 A query about CPLEX

I am using GAMS/CPLEX solver for my MILP problem. I'm having some difficulties due to the size of my problem. I would like to discuss some of the MIP options to optimize CPLEX run. Please mail me if you are using CPLEX for MILP problems.

Answer from telken@sgi.com:

There are a large number of CPLEX parameters that can be varied to improve performance. If you can get a CPLEX manual, there is a good section "Improving MIP performance..." at about page 64 in the CPLEX manual.

Two options that can save RAM memory (if running out of memory is part of your problems) are: (I'm not that familiar with GAMS, yet, so I don't know how to communicate CPLEX parameters to GAMS, but I suppose you know how to do that.)

'set mip strategy variable 3' this is called strong branching. It will significantly increase your time per LP subproblem (node), but should reduce the number of nodes in the branch-and-bound (B\&B) tree and thereby save memory. It can also speed up overall solution time.

'set mip strategy file yes' Saves some parts of the branch-and-bound tree on disk. Slows things down, but saves memory.

To stop the process before running out of memory or disk space, you may need to: 'set mip limits treememory ___' 'set mip limits file ___'

If you don't need a completely optimal solution you may want to 'set mip tolerance mipgap 0.01' or some number larger than the default of 0.0001.

2.3 General NLP solver

2.3.1 Model becomes infeasible after removing constraints

I am trying to solve the following NLP using the MINOS5 solver.

```
SET K / 1*2 /;
PARAMETERS
R(K)    / 1 0.5 2 0.5 /
P(K)    / 1 0.1  2 0.1 /
DEL1(K) / 1 2.00 2 2.00/
DEL2(K) / 1 2.00 2 2.00/
TAU1(K) / 1 0.50 2 0.50/
TAU2(K) / 1 0.50 2 0.50 /
D(K) / 1 0.700 2 0.700 / ;
VARIABLES
V(K), MU1(K), MU2(K) ,ZB(K),
ZS(K), FS1(K), FS2(K), FB1(K), FB2(K),
IDD1(K), IDD2(K),Y;
POSITIVE VARIABLES
ZB, ZS, FS1, FB1, FS2, FB2,
MU1, MU2, V, IDD1, IDD2;
EQUATIONS
OBJ, STARV1, STARV2, BLOCK1,
BLOCK2, LIM11, LIM12, LIM21, LIM22,
CAP11, CAP12, CAP21, CAP22, STA11(K),
STA12(K), BLO11(K), BLO12(K), MUL(K);
OBJ..       Y =E= SUM(K,ZB(K)) + SUM(K,ZS(K));
STARV1..    V('2')*(1-FS1('2'))*ZB('1')/D('1')+FS1('2')-
(1-R('1')/(R('1')+P('1'))*MU1('1')) =E= 0;
```

```
STARV2..    V(’2’)*(1-FS2(’2’))*ZB(’2’)/D(’2’)+FS2(’2’)-
(1-R(’1’)/(R(’1’)+P(’1’))*MU2(’1’)) =E= 0;
BLOCK1..    V(’1’)*(1-FB1(’1’))*ZS(’1’)/D(’1’)+FB1(’1’)-
(1-R(’2’)/(R(’2’)+P(’2’))*MU1(’2’)) =E= 0;
BLOCK2..    V(’1’)*(1-FB2(’1’))*ZS(’2’)/D(’2’)+FB2(’1’)-
(1-R(’2’)/(R(’2’)+P(’2’))*MU2(’2’)) =E= 0;
LIM11..     FS1(’1’) + FB1(’1’) + D(’1’)*IDD1(’1’) =L= 1;
LIM21..     FS2(’1’) + FB2(’1’) + D(’2’)*IDD2(’1’) =L= 1;
LIM12..     FS1(’2’) + FB1(’2’) + D(’1’)*IDD1(’2’) =L= 1;
LIM22..     FS2(’2’) + FB2(’2’) + D(’2’)*IDD2(’2’) =L= 1;
STA11(K)..  FS1(K) =E= 0;
STA12(K)..  FS2(K) =E= 0;
BLO11(K)..  FB1(K) =E= 0;
BLO12(K)..  FB2(K) =E= 0;
CAP11..     (R(’1’)+P(’1’))/R(’1’)-IDD1(’1’)*MU1(’1’)/TAU1(’1’) =E=0;
CAP12..     (R(’1’)+P(’1’))/R(’1’)-IDD2(’1’)*MU2(’1’)/TAU2(’1’) =E=0;
CAP21..     (R(’2’)+P(’2’))/R(’2’)-IDD1(’2’)*MU1(’2’)/TAU1(’2’) =E=0;
CAP22..     (R(’2’)+P(’2’))/R(’2’)-IDD2(’2’)*MU2(’2’)/TAU2(’2’) =E=0;
MUL(K)..    MU1(K)+MU2(K)+(DEL1(K)+DEL2(K))*V(K)*R(K)/(R(K)+P(K)) =E= 1;
MODEL SET221 /ALL/;
SOLVE SET221 USING NLP MINIMIZING Y;
SOLVE SET221 USING NLP MINIMIZING Y;

DISPLAY ZS.L, ZB.L, FS1.L, FS2.L, FB1.L, FB2.L;
```

I am able to get the optimal solution to this Problem. Now if I remove the constraints STA11,STA12,BLO11,BLO12, I am not able to get the optimal solution, GAMS says that the solution is Infeasible, although I am relaxing the constraints in the problem.

Answer from n.n:

The problem with your model is that you have trilinear constraints. This makes for a very interesting - and highly nonlinear - problem, and any solution that MINOS finds is very dependent on the starting point that you give it. Ok, let’s try a few experiments:

1) Defne a new model without the equations STA11,STA12,BLO11,BLO12.

```
MODEL SET221    /   ALL/;
MODEL test  /OBJ, STARV1, STARV2, BLOCK1,
BLOCK2, LIM11, LIM12, LIM21, LIM22,
CAP11, CAP12, CAP21, CAP22, MUL/ ;
SOLVE test using nlp minimizing y;
```

As you mention, MINOS fails to find an optimal solution for model "test". However, a quick look at the solution output shows that the infeasibilities are mainly in equations CAP11, CAP12, CAP21 and CAP22. What you >>can<< do is to see if these equalities can be written as inequalities. For example, suppose you write these equations as >= inequalities:

```
CAP11.. (R('1')+P('1'))/R('1')-IDD1('1')*MU1('1')/TAU1('1') =g=0;
CAP12.. (R('1')+P('1'))/R('1')-IDD2('1')*MU2('1')/TAU2('1') =g=0;
CAP21.. (R('2')+P('2'))/R('2')-IDD1('2')*MU1('2')/TAU1('2') =g=0;
CAP22.. (R('2')+P('2'))/R('2')-IDD2('2')*MU2('2')/TAU2('2') =g=0;
```

Now solve test with MINOS. This time it finds a solution! So if this is valid (i.e. you can change the equality to an inequality) you have a way to solve the problem. A general rule of thumb with bilinear, trilinear or quadratic constraints is that they are much easier to handle if they are in the form of inequalities.

2) You can also try the following. Solve the models in the following sequence:

```
SOLVE SET221 USING NLP MINIMIZING Y;
SOLVE test USING NLP MINIMIZING Y;
```

You'll notice that MINOS finds the same solution with both models. The reason this works is because after the first "full" model has been solved, all the variables are left initialized to the optimal values from that solution, and these are used as the starting points for the "relaxed" model.

Answer from n.n:

The problems seems to be one of non-convexity and non-uniqueness of solutions, both the (locally) optimal solution, but also the solution the the feasibility problem, where the sum of infeasibilities is reduced. The best general method in cases like these is to use the 'good' solution you already have as an initial solution to the second model -- it is feasible already. You can just add a second model statement and a second solve statement to the existing model:

```
MODEL SET221    /ALL/;
MODEL ALT   /OBJ, STARV1, STARV2, BLOCK1,
BLOCK2, LIM11, LIM12, LIM21, LIM22,
CAP11, CAP12, CAP21, CAP22, MUL /;
SOLVE SET221 USING NLP MINIMIZING Y;
SOLVE ALT USING NLP MINIMIZING Y;
```

In this case is solves to optimality very fast. The method is not 100% safe. MINOS uses a linearization technique and it may loose feasibility during the optimization. CONOPT will not loose feasibility (except for cases with severe numerical difficulty) and should be safer for the second solve. However, in this case CONOPT cannot find a feasible solution the first model !!

2.3.2 Error: ** A derivative is too large (larger than 1.0E+05)

I have been running GAMS/CONOPT lately to solve some nonlinear problems. My objective function to minimize looks like this:

```
ELIKE.. ELIKE =E= (VAR)**(NRUN)*DETA;
```

VAR is usually between 10 to 40 and NRUN is an integer around 16 to 30. DETA is around 0 and 1, but usually very close to zero. It seems that when the expression above is differentiated the derivatives are large, and I get the following error message while executing CONOPT:

```
** A derivative is too large (larger than 1.0E+05).
Scale the variables and/or equations or add bounds.
The critical limit may be increased with the line:
SET RTMAXJ X.XXE+XX
in the CONOPT control program.
Function calls: 21117 Gradient calls: 4258
CONOPT Time: 56.780 Interpreter: 6.100
Work length = 849011 double words = 6.48 Mbytes
Estimate = 849011 double words = 6.48 Mbytes
Max used = 619673 double words = 4.73 Mbytes
**** ERRORS(S) IN EQUATION ELIKE
1 INSTANCE OF - Jacobian element too large (-1.3E+05)
```

I have tried several monotonic transformations of my function above without much success. The rest of my independent variables are scaled between 0 and 1. One thing that has work partially is taking the 1/NRUN power of the expression and limit DETA as follows:

```
DETA.LO=1.0E-05;
```

The solution is right on this limit. Should I modify the RTJMAX parameter in the CONOPT control program or try some other scaling or transformation?

Answer from n.n:

A few words about the reason for the RTMAXJ parameter in CONOPT: If a derivative in a model is very large it implies that an equation is very sensitive to the particular variable. A derivative of 1.e+6 means that if you change the variable by 1.e-6, a rather small amount comparable to the tolerance on the variable, then the error in the equation will be of the order of 1, a rather large amount compared to a feasibility tolerance of 1.e-5 or smaller. Other variables will have to be changed to maintain feasibility, and the change in these variables may become very large. The result of the large derivative will often be, that it becomes virtually impossible to change the variable and still maintain feasibility. To avoid this problem CONOPT requires derivatives to be limited to about 1.e5. This is considered a "safe" limit. If you have models with larger derivatives you have two options:

Try to reformulate or scale the model so the derivative becomes smaller. You may try the GAMS scaling option described in the CONOPT documentation. This is the recommended option. Relax the limit with the line "SET RTMAXJ 1.EXX" in the conopt.opt options file. It may work, but if it does not work, do not complain! If it works, the accuracy and speed of the solution will usually be reduced. Why does CONOPT not scale the model itself? We have experimented with automatic scaling of these types of models, but it does not work very well. The reason is the nonlinearity of the model. If a derivative is very large, then the second derivative is usually also very large. Think or LOG(X), 1/X, or EXP(X) with derivatives 1/X, -1/X**2, and EXP(X) and second derivatives of -1/X**2, 2/X**3, and EXP(X), respectively. After a small change in variables the well scaled model is again poorly scaled. The large derivatives are not only a problem for CONOPT. We have seen several models that MINOS could not solve. When we tried with CONOPT we got the message about large derivatives. And after scaling the model, both CONOPT and MINOS could solve it.

2.3.3 EXIT - THE CURRENT POINT CANNOT BE IMPROVED UPON

MINOS performs about five major iterations and exits with the message:

```
    EXIT - THE CURRENT POINT CANNOT BE IMPROVED UPON.
```

I have tried in vain for the past few days trying to fix this problem. Could you suggest what might be wrong?

\Email{n.n}
\begin{verbatim}
This is not always bad, it means that MINOS is stuck in a local point without being able to establish local optimality. This may very well be the point you want to add up. It is difficult to to say more without having a look at the model. Some times it helps to restart the model a few time (it changes certain dynamic tolerances).

```
solve ....... * just add two more solves and see what happened
solve ....... solve .......
```

MINOS will restart at the previous point, and possibly, be able to say more about it. Another way is to switch algorithm.

```
solve ...
option NLP=conopt;
solve...
```

2.3.4 EXIT – NUMERICAL ERROR. GENERAL CONSTRAINTS CANNOT BE SATISFIED ACCURATELY

I have recently been encountering the MINOS (5.3) message:

```
EXIT -- NUMERICAL ERROR. GENERAL CONSTRAINTS CANNOT BE SATISFIED ACCURATELY
```

I haven't been able to isolate the cause in a medium size (1000 x 2000) model. Sometimes there are hundreds of infeasibilities in the displayed solution, sometimes several, and sometimes no infeasibilities. I suspect the solution, coming after no iterating, may not contain any clues to the problem.The problem arose after several modifications to a model which we run pretty reliably with MINOS. All constraints are linear, many variables are nonlinear, but scaled. Thought I had isolated the problem, but I'm not sure. Sometimes I've eliminated the problem by changing a couple of column vectors in the model, but not always. In some cases I've tried CONOPT instead of MINOS and it has worked without complaint, although very slowly. Anyone have a notion of what this error msg. signifies?

Answer from n.n:

This is a nasty message. It merely tells you MINOS is in numerical problems but does not give any hint what to look for. Here is the explanation from the MINOS Manual:

"An LU factorization of the basis has just been obtained and used to recompute the basic variables x_B, given the present values of the superbasic and nonbasic variables. A single step of "iterative refinement" has also been applied to increase the accuracy of x_B. However, a row check has revealed that the resulting solution does not satisfy the current constraints Ax+s=0 sufficiently well. This probably means that the current basis is very ill-conditioned. Request the SCALE option if there are any linear constraints and variables. For certain highly structured basis matrices (notably with band structure), a systematic growth may occur in the factor U. Consult the description of UMAX, UMIN and GROWTH in section 6.2, and set the LU FACTOR TOLERANCE to 2.0 (or possibly even smaller, but not less than 1.0)."

I think the last remark about the band structure does not apply to your models. I am sometimes successful in these cases to restart MINOS at this point with full scaling (hopefully we are relative close to the optimum, and scaling of the non-linear terms makes sense):

```
OPTION NLP=MINOS5;
M.OPTFILE=0; { no MINOS options }
SOLVE M USING NLP MINIMIZING Z; {say this one fails with the above message}
M.OPTFILE=1; { next solve use option file }
FILE MOPT /minos5.opt/; { generate MINOS option file }
PUT MOPT;
PUT "scale all variables";
PUTCLOSE;
SOLVE M USING NLP MINIMIZING Z;
```

You could also try to restart with CONOPT:

```
OPTION NLP=MINOS5;
```

```
SOLVE M USING NLP MINIMIZING Z; { say this one fails with the above message}
OPTION LP=CONOPT;
SOLVE M USING NLP MINIMIZING Z;
```

This may give you the speed of MINOS and the reliability of CONOPT! You may also try a few options, just to force MINOS to take a different path.I.e. turns scaling off, play with the START ASSIGNED NONLINEARS option, may be even give MINOS a basis by setting some marginals.

2.3.5 CONOPT: Fatal Error: Insufficient Memory

The error message shown above was provoked by a GAMS program after increasing the maximum \# of Superbasics in CONOPT2 (Lfnsup) in the option file from 1.500 to 2.000. As I am working under Windows NT on a Pentium with 128 MByte RAM and a lot more free disk space, a physical "out of memory" is not very probable.

Assuming a double precision working space for the Hessian, I would need something in the range of 2.E3 * 2.E3 * 8 / 1.E6 =3D 32MByte plus some extra stuff to work with the 2.000 superbasics (Arne Drud will tell us the correct formula ...).

Answer from adrud@arki.dk:

For many reasons CONOPT makes its memory guess BEFORE it reads the options file and it cannot adjust to the high Lfnsup value. You must therefore allocate extra memory yourself. Add the line "<modelname>.workspace xxx;", where xxx is some reasonable number of MBytes, before the SOLVE statement.

Allocating memory after options have been read is on the agenda for future versions, but it requires substantial changes in design.

2.3.6 MINOS: TOO MANY ITERATIONS

I recently ran a small program with two non-linear constraints, but it terminated before it converged at 200 major iterations, even though i set options iterlim = 2000. The reason for termination was "major iterations terminated before convergence/ Exit -- too many iterations. The problem was infeasible when it exited.

Answer from adrud@arki.dk:

The message indicates that you have used MINOS. MINOS uses two iteration counts, Minor (or inner iterations) and Major (or outer iterations corresponding to linearizations of the constraints). The Option Iterlim limit is related to Minor Iterations. The limit on Major iterations can only be set in an options file; the default limit is 200.

If you need more than 200 major iterations for a model with two nonlinear constraints then it is likely that: (1) The model is VERY nonlinear or (2) you have very bad initial values or (3) the model is poorly scaled.

2.3.7 CONOPT: Default accuracy

Answer from jean.mercenier@umontreal.ca:

Hi folks: this message is primarily addressed to economists using GAMS/CONOPT to solve equilibrium (i.e. square) models.

I was mentioning to Arne Drud some concerns on the accuracy of CONOPT when using default parameters; it turns out that for the class of problems we are dealing with, the default parameters are indeed unsatisfactory, and have to be changed (which is trivial to do). Following are: (a) Arne Drud's reply (and fix) to (b) my question with comments. (b) is not interesting except to convince that the problem is to be taken seriously... Jean Mercenier

(a) Arne Drud's reply 4. Accuracy: For square sets of equations there is sometimes a problem. CONOPT uses two sets of feasibility tolerances: The errors must be less than Rtnwma which by default is as high as 1.e-3, AND the errors times their dual variables must be less than some objective tolerance that usually is very small. However, for square systems of equations solved as NLPs, the duals are often zero so the last check, that is supposed to be the binding one, drops out, and the tolerance is Rtnwma. You can set a lower tolerance with "Set Rtnwma 1.e-8" in an options file. (b) My initial question/concern: I forgot to mention another problem I have encountered with CONOPT: it's relative lack of accuracy (w.r. to Minos which is my benchmark! I use the default accuracy parameter with both solvers: could it be that there is a difference there? or is it due to the type of algo, in which case I would suggest incresing that default parameter).

Let me be slightly more specific (I do not have the example unfortunately): I was solving a Hamiltonian system of difference equations (from a discounted optimal growth problem). I know (from theory) the type of smooth time-path I should obtain (more specifically: an initial jump of optimal investment/co-state variable on impact, and then a smooth monotonous decline to initial steady state level); using Minos I do obtain exactly what I expected, with Conopt I had a sort of "blip" early on the time horizon which is not quantitatively very significant but is nevertheless embarassing (in cases for which theory does not give me a clear answer, how am I to have faith in the solution computed by Conopt?) I have observed other cases (always discounted intertemp. optim. problems, because in that case I have a good intuition of the time paths) where, CONOPT gives a solution with an unexpected "blip" occuring in the middle of the time horizon, while MINOS declares "infeasible problem"; it turned out that increasing the length of the time horizon (at which a finite horizon approx of the infinite horizon transversality conditions is imposed) unabled me to solve the problem with Minos, and the "blip" to disappear with Conopt. I concluded that Conopt, for one reason or another, is less accurate than Minos when using the default options, and I have since then been slightly less confident with Conopt (When I do not encounter "the scaling problems" I mentioned previously, I usually use Conopt, and then feed that solution into Minos to "improve" the (perceived!) reliability of the solution. Again this might suggest that the value of the default accuracy might have to be lowered for Conopt if it is to generate the same level of accuracy as Minos.

Answer from adrud@arki.dk:

Under the above heading Jean Mercenier copied part of a message I send to him.

I would like to add, that we are doing something about the problem. GAMS has defined a new model class called CNS = constrained nonlinear system to be used for square sets of equations, exactly the model class for which the default tolerances can be too loose. The model class can help CGE modelers and other modelers with square sets of nonlinear equation in several aspects. (1) You do not have to add an artificial objective function. (2) GAMS will check that the model is indeed square. (3) The CNS solvers (initially CONOPT2 and PATH) can take advantage of the model class. The initial basis is straight forward, the tolerances can automatically be made tighter, and the overhead in having an objective and an optimization step can be avoided.

2.3.8 MINOS: ITERLIMIT

I have a problem with ITERLIMIT which I would like to increase to more than 200 but don't know how to do so. I have tried to list it within my XXX.gms file using option Iterlim but with no success.

Also, can anyone explain to me why do I keep on having UNBOUNDED and BADLY SCALE statements when I try to solve longer duration of my analysis (say more than 12 periods) but do not have any problem with shorter analysis (say 12 periods or less).

Answer from adrud@arki.dk:

It seems that you are using MINOS and that you reach the Major Iteration limit of 200. You will need a MINOS5.OPT file in which you have the line

```
Major Iterations xxxx
```

and you must tell GAMS to use the file by adding the line

```
<modelname>.optfile = 1;
```

before the SOLVE statement. <modelname> is the name of your model.

There can be two reasons for the Badly scaled message:

1) Poor scaling is not very problematic for small models. Almost anything will work. However, when a model grows in size, scaling becomes more critical. Your model is probably not well scaled, but for small number of time periods MINOS can handle it anyway.

2) If you have some large growth factor or large discount factor then adding extra periods will automatically make your model less well scaled. There is nothing you can do about this, you must just make sure that your building block, the single period model, is well scaled.

2.3.9 Problem with solving a quadratic program

I try to solve a (pretty) simple quadratic optimization problem where I want to minimize the squared difference of only one variable to a constant factor. There are two linear constraints who guarantee, that the variable is between given levels. The most interesting thing with this problem is, that the program can be solved using Minos but not using Minos5. Are there any major difference concerning the implemented algorithms in these two solvers?

The GAMS code is as follows

```
---------- snip -----------------------------------------------------------------------

$ONEMPTY

SETS
   RFD                / 1*1 /
   I                  / 1*1 /

PARAMETER T_DELTA(RFD)
/
1 -64452.739441
/
;


PARAMETER TOLERANCE(RFD)
/
1 0.100000
/
;


TABLE DELTA(RFD,I) $ONDELIM RFD,1 1,-200.235084 $OFFDELIM ;


VARIABLES
   X(I)
   D_HDG
   TARGET

* -------- EQUATION SECTION -------------------------------------------

EQUATIONS
   E_DHEDGE
   E_HEDGE
   E_DTOLL (RFD)
   E_DTOLU (RFD)
;

E_DHEDGE      .. SUM(RFD,SQR(SUM(I,X(I)*DELTA(RFD,I)) - T_DELTA(RFD))) =E= D_HDG;
E_HEDGE       .. D_HDG =E= TARGET;
E_DTOLL (RFD) .. SUM(I,X(I)*DELTA(RFD,I)) =G=  T_DELTA(RFD)
                    - TOLERANCE (RFD) * ABS (T_DELTA(RFD));
E_DTOLU (RFD) .. SUM(I,X(I)*DELTA(RFD,I)) =L=  T_DELTA(RFD)
                    + TOLERANCE (RFD) * ABS (T_DELTA(RFD));

* -------- MODEL SECTION ------------------------------------------

OPTION DECIMALS = 8;


MODEL Hedge
/
   E_DHEDGE,
   E_HEDGE,
   E_DTOLL,
   E_DTOLU
/; Hedge.OPTFILE = 1;


option nlp = minos5; option optcr = 0.0; option optca = 0.0;

SOLVE Hedge MINIMIZING TARGET USING NLP;

---------- snip ---------------------------------------------------
```

Answer from adrud@arki.dk:

This is one of my favorite subjects: SCALING.
If you look at the equation listing for your model you
will see:

```
---- E_DHEDGE    =E=


E_DHEDGE.. - (2.581140E+7)*X(1) - D_HDG =E= 0 ;

      (LHS = 4.1541556E+9, INFES = 4.1541556E+9 ***)
```

Nonlinear models can be hard to solve if terms and derivatives
are very large. And 2.8e7 = 28 million is very large. 4.15e9 is
very very large.

You can scale the model using the .scale and .scaleopt feature in GAMS. The objective is to get derivatives that are not too much over one in absolute value. The equation scale is selected to be of the order of the largest derivative or term in the equation, for example

```
e_dhedge.scale = 1.e6;
```

The equation says d_hdg =e= ... and in order to preserve the coefficient 1 for d_hdg you must also have

```
d_hdg.scale = 1.e6;
```

Now d_hdg is scaled and it appears in

```
E_HEDGE .. D_HDG =E= TARGET;
```

so you will also need

```
e_hedge.scale = 1.e6; target.scale = 1.e6;
```

And finally you must tell GAMS/MINOS to use scaling with the statement

```
Hedge.scaleopt = 1;
```

And your model solves nicely.

Other things could be improved in the model. You have the same expression in two constraints and inside the SQR in the objective function. Define this term as an extra variable using one extra constraint, and your models looks much simpler.

Answer from **tmunson@cs.wisc.edu**:

The new version of minos includes some preprocessing which removes the D_HDG variable and E_HEDGE equation in your model by putting the quadratic constraint into the objective function where minos can more easily deal with it.

You can do the same in your code by changing the E_DHEDGE equation to:

```
   E_DHEDGE.. SUM(RFD,SQR(SUM(I,X(I)*DELTA(RFD,I)) - T_DELTA(RFD))) =E= TARGET;
```

and removing the appropriate variable (D_HDG) and constraint (E_HEDGE) from the model. With these modifications, minos5 solves it in one major iteration.

I believe that other changes were made to the minos implementation, but do not have information on those. For your model, preprocessing is the key.

2.4 MCP solver

2.4.1 Problems with the MILES Solver

I have a problems when i solve my CGE model. I try different solvers (MINOS, Conopt, Miles) on the same program and the MILES one doesn't work. I have done like the US CGE model. I've just removed the objective. If i run the model with no changes in exogenous variable, this solver as others gives correctly the initial basis. However, when i try to simulate one policy, i feel the solver doesn't start. It says that one equation is infeasible and this is the equation changed for simulation.

Answer from rutherford@colorado.edu:

The MILES solver is for mixed complementarity problems, and this model format requires that you be somewhat careful about the use of upper and lower bounds. If you are solving a nonlinear system of equations using one of the optimizers, it is easy to apply bounds but if any are binding at the solution, your "dummy" objective may influence the result. When you use an MCP formulation, it is essential to associate bounded variables with equations. When a variable hits a bound, it is then "releases" the associated equation, just as in a Kuhn-Tucker system. So far as your specific problem, I cannot really comment unless I see the code. From what you have said, it sounds as though MILES has detected a logical inconsistency which probably derives from a failure to associate bounded variables and equations. That's my best guess.

2.4.2 CUMULATIVE_PIVOT_LIMIT_EXCEEDED

When use my GE model with a simple SAM, it works perfectly. The simulation fails, when I use a "real" SAM. There is a "cumulative pivot limit (0)exceeded" message (2-norm 1.5E-6, inf-norm 1.1E-6). The consistency of the "real" SAM was obtained by a RAS, but, of course there is still some differenence between row sum and column sum (<1.0E-5). Is the EXIT due to this small inconsistency? How can I reduce the sensibility of the solver-consistency-check? - I searched in the literature, but without success

Answer from n.n:

The solver properly exits when it hits the pivot limit, even though it has not solved the model. You can use a (solver-specific) options file to change the tolerance of the solver, but there would be little point in doing so. From what I understand of GE modeling, you don't really want to solve the model, just check your benchmark equilibrium.

Answer from rutherford@colorado.edu:

When you have large-scale datasets, it is often the case that rounding errors in the data construction exceed the convergence tolerance. I typically continue to use model.iterlim=0; to set up a benchmark replication test, and rather than mess with the options file I prefer to simply report results from the test. model.OBJVAL returns a norm of the residual. Depending on scaling of the data, this should be on the order of 1.e-3 to 1.e-6 if you have a consistent benchmark. Note that scaling can be important here. If the raw units of the data are such that you have reference quantities of 1.e6, you may not be able to reliably achieve the default tolerance of 1.e-6. Best to scale the input data once at the start so that numbers entering the Q: fields are roughly unity. Here is some code for reporting benchmark replication:

```
BMKSTAT("DEVIATION") = model.OBJVAL;
BMKSTAT("MODELSTAT") = model.MODELSTAT;
BMKSTAT("SOLVESTAT") = model.SOLVESTAT;
DISPLAY BMKSTAT;
```

2.4.3 Iteration limit

Here's hopefully a very simple question: I am trying to solve an MCP problem using MILES and the solver continues to exit after 100 iterations, stating "MILES iteration limit exceeded" in spite of my setting the iterlim to 1000. Is there some separate "MILES" iteration limit that can be set as well?

Answer from rutherford@colorado.edu:

Look in the solver manual. Set <model>.optfile = 1; then include miles.opt file in the currently connected directory. 99% of the time, if MILES does not converge in 25 iteraitons, it will never do so. If the model is correct, PATH is typically more robust.

2.4.4 MILES vs. PATH

I have a MCP problem which is going to grow. I have both MILES and PATH,and since PATH costs extra, I am assuming it is in some sense better. I have not been able to find anything clear in the documentation about their relative merits. Running the small model I have now produces exactly the same solution, and in approximately the same time.The model is soon to be shared with other users, and I don't know if they should buy PATH. Any experience or pointers to relevant literature would be appreciated.

Answer from n.n:

Both are fine algorithms. For some altruistic reason Tom Rutherford is no longer charging for access to MILES if you have MPSGE, so I think it is bit gratuitous to conclude that "since PATH costs extra, I am assuming it is in some sense better." I think that Tom has just decided that his comparative advantage is in doing economics rather than developing the algorithmic stuff, and so we tend to default to PATH these days. If your model runs with MILES or PATH I would just say so and let the user decide what to do. If you find it does not solve with MILES then flag that, and build it into your code with an OPTION MCP=PATH statement that "alerts" the user if they do not have the required algorithm.

Answer from sdirkse@gams.com:

A comparison of several MCP solvers, including MILES and PATH, is available in S.C. Billups, S.P. Dirkse, and M.C. Ferris, "A Comparison of Large Scale Mixed Complementarity Problem Solvers", Computational Optimization and Applications, Vol. 7, pp 3-25, 1997.The original technical report for this paper is available by anonymous ftp at ftp.cs.wisc.edu/math-prog/tech-reports/95-16.ps.Z

Answer from rutherford@colorado.edu:

I concur with the analysis of Ferris, Billups and Dirkse. PATH is basically a more robust solver than MILES, and it has many more solution strategies which can be adjusted with the options file. It is possible to configure PATH to perform more or less exactly the same as MILES. The true test of a solver comes during the early stages of model development, when ideas have not been clearly developed and (in the words of an electrical engineer I once knew) there are "short circuits between the headphones". At that point, a robust solver can return a screwy solution which helps you to identify where you have made a logical error. A non-robust solver returns no solution, at which point you do not know whether there is a "good" solution which has not been found or if the model has no solution. Working with PATH during the early stages of model development, if I hit a scenario which does not solve, I typically try three or four alternative solver configurations. If those do not work, I conclude that there is a logical error in the model specification. 95% of the time, I find a bug. Working with MILES there are fewer configuration choices, and the solver can more often fail to find a solution when one exists. It is hard to measure how much better one solver is than the other, but I know that I have been much more efficient and have been solving far more complex models since shifting to PATH a couple years ago. So far as efficiency (cpu time) on operational models, I don't think that you will find much difference. They use the same factorization engine (LUSOL) and this is where the vast majority of work is done. My suggestion to MCP modelers is to use PATH for development work, but then verify that your final simulations can be performed with MILES. This way your model can be distributed to anyone who has GAMS. A final note regarding pricing. We decided to release MILES as a "free" solver with GAMS (like BDMLP) for two reasons: (i) I did not want to do any more development work on algorithms, and by giving the code away I felt less compulsion to work on every model that failed to solve (I still fix bugs as they arise, but I generally do not work on improving efficiency or robustness.), and (ii) We wanted to promote complementarity modeling in general.

2.4.5 Matrix balancing with PATH

Tom Rutherford wrote a note on matrix balancing in which he concluded that path is more efficient than MINOS or CONOPT for matrix balancing. I have included below the model as presented in the original mail (Tom, thanks for sending me your example program).

I do not have a license for path, so I could not run path for large matrices, but I found that, after scaling the problem and giving an initial point, minos5 outperforms the other solvers with respect to CPU time needed. However, the differences between the solvers are not that large. For large models the difference between MCP and NLP becomes perhaps significant (?). However, especially with respect to Conopt and Conopt2 the needed CPU time is reduced with about 75% after scaling the problem. Personally I still prefer NLP over MCP because it is difficult (at least for me) to add additional constraints, but perhaps it is a tradeoff between time spend in formulating the problem and thinking about scaling of the problem.

Answer from **rutherford@colorado.edu:**

Find the attached GAMS program. I added your scaling suggestions and alternative objective function. This problem is solved by PATH in 47 seconds on a 150 MHz machine. Suggest that you try this with MINOS, CONOPT or CONOPT2. On any machine...

```
$title evaluate performance of alternative solvers for the matrix
$setglobal dimension 150
$setglobal seed 1
$setglobal solver minos5

$if exist matbal.def $include matbal.def

$if setglobal seed option seed=%seed%;

parameter results;

scalar density /0.5/;

set     i /1*%dimension%/;
alias (i,j);


parameter a(i,j) random matrix
          scalef  scalefactor;

a(i,j)$(uniform(0,1) gt 1-density) = uniform(0,1);


variable x(i,j) estimated matrix
                sqrdev  least-squares objective;

equations objdef defines the least squares
                balance(i)      row-column balance condition;

*objdef.. sqrdev =e= sum((i,j), sqr(x(i,j)-a(i,j)));
```

```
*relative objective function
objdef.. sqrdev =e= sum((i,j)$a(i,j), sqr((x(i,j)-a(i,j))/a(i,j)));

balance(i).. sum(j, x(j,i)) =e= sum(j, x(i,j));

************************************************************
*setting starting
point x.l(i,j)=a(i,j);

*setting average scaling
*scalef=sum((i,j), a(i,j))/(sqr(%dimension%));
*x.scale(i,j)$(a(i,j))=scalef;

*setting level scaling x.scale(i,j)$(a(i,j))=a(i,j);

****************************************************************



x.fx(i,j)$(not a(i,j)) = 0; x.lo(i,j) = 0;

$if %solver%==path $goto mcp

option nlp=%solver%;

model matbal /all/;
matbal.scaleopt=1;
matbal.holdfixed=1;
matbal.workspace=2;
solve matbal using nlp minimizing sqrdev;

results("%dimension%","%seed%","mstat","_","%solver%") =
matbal.modelstat;
results("%dimension%","%seed%","sstat",  "_","%solver%") =
matbal.solvestat;
results("%dimension%","%seed%","cpu",  "_","%solver%") = matbal.resusd;
results("%dimension%","%seed%","iters","_","%solver%") = matbal.iterusd;
$goto log


$label mcp
option mcp=%solver%;

variable p(i) shadow prices on balance constraint;


equation kkt(i,j) kkt conditions for variable x;

* kkt(i,j)..      2 * (x(i,j)-a(i,j)) =g= p(j) - p(i);
kkt(i,j)$a(i,j).. 2 * (x(i,j)-a(i,j))/a(i,j) =g= p(j) - p(i);

model matbalmcp /balance.p, kkt.x/; *matbalmcp.scaleopt=1;

solve matbalmcp using mcp;
```

```
results("%dimension%","%seed%","mstat",  "_","%solver%") =
matbalmcp.modelstat;
results("%dimension%","%seed%","sstat",  "_","%solver%") =
matbalmcp.solvestat;
results("%dimension%","%seed%","cpu",  "_","%solver%") =
matbalmcp.resusd;
results("%dimension%","%seed%","iters","_","%solver%") =
matbalmcp.iterusd;


$label log
*results("%dimension%","%seed%",i,j,"%solver%") = x.l(i,j);
results("%dimension%","%seed%","object","_","%solver%") =
                       sum((i,j), sqr(x.l(i,j)-a(i,j)));
results("%dimension%","%seed%","object2","_","%solver%") =
                       sum((i,j)$a(i,j), sqr((x.l(i,j)-a(i,j))/a(i,j)));
display results;
```

Answer from erwin@gams.com:

A few remarks from an "outsider" about these balancing problem.

1) I guess in absolute terms MINOS should not do too poorly as it can exploit that all non-linearities are in the obj. So the fact that CONOPT and especially PATH is doing so good on these problems is even more pronounced.

2) On the other hand, these squared terms may lead to large numbers of superbasics, which may be the cause why MINOS is slowing down for larger problems.

3) If you are really after speed, one could try to use absolute values in the obj instead of quadratic terms (i.e. using a different norm). The problem can then be cacasted into an LP and a commercial LP solver can be used. The solution may be less satisfactory however as the penalty for relative large deviations is less. The solution may contain a few bigger deviations and many zero deviations (the pain is less evenly spread).

2.4.6 Queries about the PATH solver (using GAMS 2.50)

I use the 2.50 version of GAMS and have two questions related to the PATH solver:

1) Occasionally, when I formulate a model that presumably is infeasible, PATH, instead of declaring it as such, redefines an equation and reports that a solution has been found. Is there any way of turning off this mechanism? I would prefer to be told up front that there are problems with my model. (In fact, the first time it took some time before I located the information about redefinitions.)

2) As I solve a CGE model the screen displays: "EXIT Solution Found, Major iterations..... 3, Minor iterations..... 10."

However, in the list file I find:

```
*====================================

               S O L V E      S U M M A R Y

     MODEL   MOR94
     TYPE    MCP
     SOLVER  PATH                FROM LINE  6209

**** SOLVER STATUS 1 NORMAL COMPLETION **** MODEL STATUS 1 OPTIMAL

 RESOURCE USAGE, LIMIT          1.148   100000.000
 ITERATION COUNT, LIMIT         0         5000
 EVALUATION ERRORS              0            0

 Work space allocated           --    1.34 Mb

*========and toward the end:

**** REPORT SUMMARY : 0 NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
                             0  REDEFINED
                             0     ERRORS

*====================================
```

I.e., an iteration count of zero in spite of the fact that the solver clearly iterated and found a solution different from the base. When I do the same run with MILES as solver, the number of iterations is reported accurately:

```
 *====================================

               S O L V E      S U M M A R Y

     MODEL   MOR94
     TYPE    MCP
     SOLVER  MILES               FROM LINE  6209

**** SOLVER STATUS 1 NORMAL COMPLETION **** MODEL STATUS 1 OPTIMAL

 RESOURCE USAGE, LIMIT          2.309   100000.000
 ITERATION COUNT, LIMIT        25         5000
 EVALUATION ERRORS              0            0

*====================================
```

Answer from ferris@cs.wisc.edu:

REDEF is a facet of the MCP format in the GAMS language, not of the PATH solver. There is a description of the GAMS/MCP format and how it relates to PATH 4.0 in the paper:

```
http://www.cs.wisc.edu/math-prog/tech-reports/98-12.pdf (pdf format)
http://www.cs.wisc.edu/math-prog/tech-reports/98-12.ps  (ps format)
```

A description of REDEF can be found on pages 15 - 16.

We would welcome any comments on the above document. Note however, that a few of the options and some of the output is particular to version 4.0 that will be released during October 1998.

[Point 2 is a bug that is fixed in version 3.3; the iterations in the PATH log are correct, not those in the listing file]

2.4.7 PATH and convergence

The simple question is, how do I get PATH to accept a ‘loose’ convergence tolerance.
I am using PATH to solve an MPSGE model, and want to abort if I fail to benchmark.
The data starts with a deviation reported as 1.3E-6, but the following triggers ABORT with solver status 2 and model status 6 (even though the reported deviation is 1.3E-6 at this point. If I reset the ITERLIM to 2000, the model solves, reporting a deviation of 9.2E-13.

```
* GAMS CODE
MRT_C.ITERLIM = 0; $INCLUDE MRT_C.GEN SOLVE MRT_C USING MCP;
ABORT$(MRT_C.MODELSTAT NE 1) "Model does not benchmark!" ;

* PATH.OPT
tolerance_converg 30 ; * Yes, that’s a BIG number!
```

Answer from rutherford@colorado.edu:

Rather than messing with an options file, how about this:

```
   ABORT$(MRT_C.OBJVAL GT 2E-6) "Model does not benchmark!" ;
```

Answer from ferris@cs.wisc.edu:

The solution is to use the option "con_tol 1e2", not "tol_con ". The log file should inform you that the option was not recognised.

With such a large tolerance, the listing file might indicate that the problem has infeasible constraints at the "solution". ie the "solution" satifies termination criteria of PATH (which we have just modified), but does not satisfy the GAMS notion of what is a feasible point of an MCP.

The enclosed simple example shows the effect. path.opt just contains:

```
   con_tol 1e2;
```

```
positive variable x; equation f;

f.. x =g= 1;

x.l = 2;

model foo /f.x/; foo.optfile = 1; foo.iterlim = 0; solve foo using mcp;

display foo.modelstat;
```

2.4.8 Memory problems in PATH

```
I work on a big CGE model with Gams/Path but when I try to solve
the model I have the Message "Memory: allocate :allocation of
1138625212 bytes failed". I read on the gams site that i must
change the size of the virtual memory (I work under Win 95), so I
made it. But now the message is "Memory: allocate :allocation of
-20229148212 failed" with a minus ! I have tried many different
parameters but i have always the same problem . Have somebody a
solution for me (my config is Pentium II 233Mhz with 64 Mb RAM
and 4Gb of HD)
```

Answer from adrud@arki.dk:

```
The numbers you show are VERY large and close to integer
overflow. The first number indicates that PATH would like to
allocate over 1 GB Ram, the second that there has been some
overflow.

In an earlier version of PATH there was a problem like this for
models with more than 47000 equations because 47000*47000 gives
an overflow. You may try to turn the memory allocation off by
using the statement "<modelname>.workspace = xxx;" after the
declaration of the model <modelname> and before the SOLVE
statement. xxx is the amount of memory you will give to the
model. Try with 50 (the unit is Mb) and increase it if necessary.

Warning: Even if you get it into the solver, this model seems to
be very large and you may run into many other problems.
```

2.4.9 Solutions in Miles

```
I am running a large GAMS/CGE model using miles as the solver,
and I get a baseline optimal solution. However, when I run an
impact program from it using a restart option the model converges
at 4.8E-4, above the convergence tolerance of 1.0E-6. If I use a
miles option file and set "contol" above 4.8E-4 it solves
optimally. The simple solution would be to use the miles option
file, but I wonder whether it is the "right" solution. Any
thoughts?
```

Answer from rutherford@colorado.edu:

```
Most often this type of problem has to do with scaling. If
```

numbers in the function blocks are too large, it may be impossible to achieve a convergence tolerance of 1.e-6. Suggest that you look at this -- when default activity levels are unity, the function coefficients should be on the order of 1 (say between 0.05 and 100).

Answer from coupal@uwyo.edu:

Thanks for the response on solutions in MILES and scaling. However, I am uncertain about how to look for scaling problems in the gams list file. To look at the function coefficients one can go to the equation listing in the gams lst file. However, in that section we have parenthesized coefficients (which means it is a non-linear equation and the coeff. is a function of activity levels of other variables) and non parenthesized coefficients. Should the order of magnitude be close between all types of coefficients or just the non-parenthesized ones?

Example of large differences in parenthisized coeffs:

Equation:

```
EXPORT(i).. E(i) =E= XXD(i)*(pe(i)*ER/PD(i) * (1-gamma(i))/gamma(i))**tau(i);
```

Equation listing:

```
EXPORT(18).. E(18) - (1.5609)*XXD(18) + (67664.1595)*PD(18)

      - (67664.1595)*ER =E= 0 ; (LHS = 5.456968E-12 ***)
```

Answer from rutherford@colorado.edu:

Short of providing a crash course on numerical analysis, here is the best I can do: You need to pay attention to the magnitude of variables and parameters which enter into a model. You can think about the inputs to a general equilibrium model as though they were a matrix of base year transactions. Formally, it is irrelevant whether you measure these transactions in billions of dollars, millions of dollars or pennies; but for the sake of evaluating demand and supply functions it can matter a lot.

If you pick up an undergraduate text on numerical methods, you will probably find a chapter near the front of the book on numerical precision. Here you will find that some operations are safer than others. For example, it is likely that you will have a more precise result for A + B than for A**B, particularly when A or B is a large number. For economic equilibrium models, the rule of thumb is that values entering into transcendental operations (log(), exp(), **) should always be kept as close to unity (one) as possible. It is for this reason that MPSGE assumes a default value of unity for all prices. From a practical standpoint, you may want to:

1) Try changing the convergence tolerance for MILES. Not a good long-term approach, but it will permit you to continue with your analysis.
2) Rescale base year values to a smaller magnitude.
3) Look at all of the transcendental operations in your model and see if the arguments are close to unity.
4) Get a copy of PATH and see if it processes your model more

> effectively. (Ferris and his students have been getting better and better results during the past year -- I am now able to reliably solve a number of models which would never be processed by MILES.)

Finally, on interpreting the GAMS listing output. In simple terms this provides a report of the initial linearization -- the Jacobian matrix. See the GAMS manual for further information. If you can't make sense of this, then just generate some display statements at the point of solution to report values entering the transcendental functions.

2.5 MINLP solver

2.5.1 When to switch from a NLP to a MINLP-formulation

I am trying to solve a problem using NLP(MINOS). All the constraints are linear, but the obj function is nonlinear. GAMS is reporting a divide by zero error for each of the terms in the obj func where I try to divide by a variable. Should this be the case? Secondly, I would really like to have the variables be binary. I originally thought the problem was completely linear (except for the binary variables), but I later discovered that the obj func was nonlinear.

```
POSITIVE VARIABLE
X(I) 1 if i is used
Y(G,TD) 1 if g and td are used
W(MB,TD) 1 if mb and td are used;
X.UP(I) = 1;
Y.UP(G,TD) = 1;
W.UP(MB,TD) = 1;
*maximize
OBJ..   Z =E= SUM((G,TD), LFM(G,TD)*Y(G,TD)) / BASEA
+ SUM((G,TD), DFM(G,TD)*Y(G,TD)) / BASEB
+ SUM((MB,TD), MFM(MB,TD)*W(MB,TD)) / BASEC
+ SUM(I, CRFM(I)*X(I)) / (4*BASED);
* + SUM(I,WEIGHT(I)*X(I)) / (HC*SUM(I,ETA(I)*X(I))**1.5)
* + SUM(G,DET(G)*X(G)) / (SEARCHW*SUM(I,WEIGHT(I)*X(I)))
* + SUM(G,VUL(G)*X(G))
* + SUM(MB,HEAT(MB)*X(MB)) / (SUM(I,WL(I)*X(I))**2)
* + SUM(MB,SN(MB)*X(MB)) / (SUM(I,WL(I)*X(I))*SUM(I,ETA(I)*X(I)));
```

Notice that right now, I have the terms where I divide by a variable commented out b/c including any one of them in the obj func gives me a divide by zero error.

Answer from n.n:

GAMS complains during execution because the level values of all the variables are 0 to start of with - the division by 0 then causes an overflow. Let me offer one way of getting around the problems that you have. this involves converting your model into an MINLP and solving it using DICOPT. First of all, I would equate all the sums over x,y,w to additional variables. this doesn’t yet add to any nonlinearity and leaves the binary variables as is. So, the constraints become:

```
obj..   z =e= term1/basea + term2/baseb + term3/basec +
term4/(4*based) + term5/(hc*term6)**1.5 +
term7/(searchw*term5) + term8 + term9/term10**2 +
term11/(term10*term6) ;

where

term1 =e= sum((g,td),lfm(g,td)*y(g,td)) ;
term2 =e= sum((g,td),dfm(g,td)*y(g,td)) ;
term3 =e= sum((mb,td),mfm(mb,td)*w(mb,td));
term4 =e= sum(i,crfm(i)*x(i)) ;
term5 =e= sum(i,weight(i)*x(i));
term6 =e= sum(i,eta(i)*x(i));
term7 =e= sum(g,det(g)*x(g));
term8 =e= sum(g,vul(g)*x(g));
term9 =e= sum(mb,heat(mb)*x(mb));
term10 =e= sum(i,wl(i)*x(i)) ;
term11 =e= sum(mb,sn(mb)*x(mb)) ;
```

Here term1,..,term11 are continuous variables and x,y and w are binary variables. This is now in the form of an MINLP that DICOPT can handle and the remaining issue is purely of the NLP modeling. You will now need to keep term5,term6 and term10 away from 0 by providing lower or upper bounds as is the case. Another option is to remove the nonlinearity from out of the objective function and further define new variables term12,..,term15:

```
term12*(hc*term6)**1.5 =e= term5 ;
searchw*term5*term13 =e= term7 ;
term14*term10**2 =e= term9 ;
term15*term6*term10 =e= term11 ;
```

this makes your objective function:

```
obj..   z =e= term1/basea + term2/baseb + term3/basec
+term4/(4*based) + term12 + term13 +
term8 + term14 + term15 ;
```

provide tight upper and lower bounds to term1,..,term15 and make sure that all the terms are properly scaled.

2.5.2 Problems switching from NLP to MINLP

I have been solving statistical sampling problems as NLP using GAMS/CONOPT and GAMS/MINOS successfully for some time. Now I want to solve the equivalent MINLP problem and I need some help. My original problem is formulated as,

```
minimize:
determinant [A]
subject to:
aij=F(x)              (the entries of matrix [A] are functions of the entries of matrix x)
[A]=L'L               (Cholesky decomposition of matrix [A])
det[A]=prod(i,a(i,i)) (det[A] is the product of the main diagonal)
0<= x <= 1            (x is a continuos matrix variable with each entry
                       bounded between zero and one)
```

No problem so far (although is a very nonlinear problem). Now I want x to be discrete and take values 0 or 1 only, and then 0, 0.5 and 1.0, and then 0, 0.25, 0.5, 0.75 and 1.0 and so on. With these changes, the problem becomes a difficult MINLP. Variable x becomes binary or integer and appears non linearly in the objective function right from the start. Here is what I have tried:For the case where x is either 0 or 1, I have solved the problem using relaxed MINLP, with option RMINLP. This gives me a solution in which most of the entries of x are 0 or 1 with a few exemptions (why is this I don't know). From what I can read from the DICOPT++ manual this is only a stepping stone for the next step. I add a binary variable xb the constraint x=xb and solve the problem using DICOPT++. Now the binary variables do not appear nonlinearly in the objective or the constraints. This option has not worked, since CONOPT blows up when DICOPT++ calls it. The reason for this is not clear to me, it might be due to my problem formulation My questions are: Am I doing things wrong? If I can use trick No.2 above, then how can I specify discrete values for x, such as 0, 0.5 an 1.0?

Answer from n.n:

If I understand it correctly, the easiest way of adding the requirement of discrete values in a continuous problem for the subset of variables v is the following:

```
Assume min f(u,v) st g(u,v)<0
```

Say each component of vector v, vi is restricted to discrete values bij them add the following:

```
vi = sum(j, bij*yij)
sum(j, yij) = 1
```

where yij are binary variables If vi is only zero/one the only need vi=bi where vi is continuous, bi is binary. If program blows-up could it be you were using nonzero lower bound on vi in the continuous? The difficulty might be due to the presence of rational or bilinear terms.

2.5.3 MINLP output

While working with a Mixed Integer Problem, using MINLP, if the NLP gives integer solution, it appears the message:

"The relaxed NLP gave a solution where all the integer variables have integral values. There is no need to continue the search".

Sand the .lst file doesn't show the output. I know it is
preferable to run the relaxed NLP (using RMINLP) before. The
problem is the solution needs the MINLP solver for certain
parameters values and the RMINLP for others. Since I use a LOOP
for solving the model, I would like to obtain the output in all
cases in the .lst file or the .put file.

Answer from ricardo@isr.uni-stuttgart.de:

In order to control the output in the listing file (.lst), check
in the GAMS User's Manual the item "options controlling output
detail" (appendix D).

In your message you already have the answer. The MINLP-solver
first looks for a RMINLP (NLP) solution and if all the integer
variables have integer solutions there is no need to go further
(remember the warning message "The relaxed NLP gave a
solution..."). Just use the MINLP solver and in the listing file
you'll have the appropriate indication if the relaxed problem
already have an integer solution. If you prefer to use PUT
statements to output items instead, just use them *inside* the
loop. Furthermore, if you don't want to look for model and solver
statuses in the listing file, remember to write them down in your
output file (your_modelname.modelstat modelname.solvestat).

Chapter 3

General Modeling Examples and Tricks

During nonlinear programming, we need Langrangian muliplier for constraints. If the value of is required for another modeling, is is possible for me to get the value from gams modeling? How can I refer to the value of lagrangian multiplier? Any answer will be appreciated.

Answer from n.n:

The Lagrange multipliers, also known as marginals, can be accessed after a solve as follows. Say the constraint of initerest is called CON(j,k) and you want to set lamda(j,k) to the value of its marginal after the solve. Then anywhere after the solve statement, write: lamda(j,k) = CON.M(j,k) ;

3.1 An IP formulation question - modeling logical constraints

Here is what I want to represent in a model: Given a continuous variable X, assign a value to variable Y as follows:

```
        Y = 0   if  X < 0
        Y = K-X if 0 < X < K/2
        Y = X   if K/2 >= X
```

I seem to recall something about using binary variables and some number called M to implement this sort of thing, but my memory is hazy.

Answer from rrosenthal@monterey.nps.navy.mil:

This is a very hard example of using binary variables to enforce logical constraints. Before giving it a try, let me first do a simple example to show the basic principle.
Part 1:
Say X is a nonnegative variable, B is a binary variable, and you want to enforce the logical relationship: if X>0 then B=1. You can turn this logical relationship into a linear inequality constraint as follows: X <= B*xmax, where xmax is a prespecified upper bound on X (xmax} is the "Big-M" you were thinking of, but

solvability improves if you keep these bounds as small as possible.). The idea is that if X "wants" to be positive, then B must be "turned on" to 1, and if X "wants" to be zero, then the constraint is irrelevant.

Part 2:

We didn't specify what B has to be if X=0. Suppose we want B=0 if X=0, (equivalent to the converse of Part 1). A common situation in optimization modeling is that we get this relationship to hold automatically, i.e., we don't need to create an explicit constraint to impose the relationship. This nice situation would arise, for example, if B has a positive cost in the objective function (e.g., a fixed cost for starting up the activity represented by X) and there are no constraints other than the one above that would motivate B to be one. This type of reasoning is very important in formulating optimization models. Sometimes you don't need to add complicating constraints because the economics of the problem are forcing some constraints to hold automatically.

Part 3:

What if the nice situation of Part 2 does not apply in our problem? To explicitly enforce the relationship: X>0 if and only if B=1, use the constraint above plus: X >= B*xmin, where xmin is a prespecified minimum on X if it is nonzero.

Now, let's tackle your problem. I don't know the context, so I can't know if any of the simplifications of Part 2 above will apply. Therefore, I'll assume they don't and try to cover all the bases explicitly. You may be able to delete some of the converse constraints below. We are given X, an unrestriced continuous variable and Y a nonnegative continuous variable. Introduce two binary variables A and B, and the constraint A+B <=1. Now write the logical relationships that we want to convert to linear algebra this way:

```
    If X<0, then A=1 and Y=0.
    If X>k/2, then B=1 and Y=X.
    If 0<=X<=k/2, then A=B=0 and Y=k-X.
```

Introduce parameters:

```
    xmax = largest possible value of X (and also of Y)
    xmin  = smallest possible value of X (a negative number)
    eps  =  small positive number
```

Here's the formulation:

```
     Logical constraints      Algebraic constraints
1)                            A+B <= 1
2a)  If X<0 then A=1.         -X <= A*(-xmin)
2b)  If A=1 then X<0.         X <= (1-A)*xmax + A*eps
3)   If A=1 then Y=0.         Y <= (1-A)*xmax
4a)  If X-k/2 >0 then B=1.    X-k/2 <= B * (xmax - k/2)
4b)  If B=1 then X-k/2 >0.    -(X-k/2) <= (1-B) * (-xmin+k/2) - B*eps
5)   If B=1, then Y-X=0.      Y-X >= 0   [a property of Y in all cases]
                              Y-X <= (1-B) * k
6)   If A=B=0, then Y=k-X.
```

```
6a)  If A=0, then Y>=k-X.     Y >=  k-X -A*(k-xmin)
6b)  If B=0, then Y<=k-X.     Y <= k-X + B*(2*xmax-k)
```

Notes: The epsilon term is needed in (2b) because of the discontnuity of Y at X=0. The converse of (3) is not needed because (5) and (6) cover the A=0 case. The epsilon term in (4b) is optional since Y is continuous at X=k/2. I don't promise this will be easy to solve. (I am curious why there is the discontinuity at X=0.) But I'll be very interested to know what happens if you experiment with these ideas.

Answer from n.n:

This model can be formulated with the use of 0-1 variables and "big-M"parameters as discussed by Rick Rosenthal. Another way of formulating such conditional models is to use disjunctions.Besides, we proved that the disjunctive formulations produce much tighter relaxation gaps than the "big-M" formulation. A detailed description of the approach and proof on the relaxation gaps can be found in: Turkay, M., and I.E. Grossmann, "Disjunctive Programming Techniques for the Optimization of Process Systems with Discontinuous Investment Costs-Multiple Size Regions", Ind. Eng. Chem. Res., 35, 2611-2623 (1996).

The above model can be represented with disjunctions as: {Y = 0, X < 0} OR {Y = K-X, 0 < X < K/2} OR {Y = X, K/2 >= X} The above formulation has a discontinuity at X=0; therefore, a very small positive parameter eps has to be defined. Besides, in the last term (i.e.,{Y = X, K/2 >= X}) no upper bound is defined for X. Assume an upper bound Xmax for this term. The formulation then becomes: {Y = 0, X < eps} OR {Y = K-X, eps < X < K/2} OR {Y = X, K/2 >= X<=Xmax} The above model written as an MILP by introducing a binary variable Z for each term, and by disaggregating the continuous variables X and Y for each term of the disjunction. The MILP formulation is:

```
Y=Y1+Y2+Y3                (continuous Y is disaggregated to Y1, Y2, and Y3)
X=X1+X2+X3                (continuous Y is disaggregated to Y1, Y2, and Y3)
Y1=0                      (first term)
X1<eps*Z1                 (condition for the first term)
Y2+X2=k*Z2                (second term)
eps*Z2<X2<(k/2)*Z2        (condition for the second term)
Y3-X3=0                   (third term)
(k/2)*Z3<=X3<=Xmax*Z3     (condition for the third term)
Z1+Z2+Z3=1                (only one term is active at the solution)
Z1,Z2,Z3=0,1              (Z's are 0-1 variables)
```

We also provided a theoretical proof on the conditions that the above system can be solved as a Relaxed MIP in the following paper: Grossmann, I.E. and M. Turkay, "Solution of Algebraic Systems of Disjunctive Equations", Comput. Chem. Engng., 20, S339-S344 (1996).

Answer from n.n:

Here is an alternative approach which will do the trick as well,

```
1) In order to model the discontinuity between the
   first two regimes, and since the 'strictly less than'
   constraint cannot be properly enforced, we will
   rewrite your problem as:
        Y = 0   if  X <= -eps
        Y = K-X if eps <= X <= K/2
        Y = X   if X >= K/2
   where eps is a very small number (say 1.0e-6)
2) Let me assume that L,U are the lower and upper bounds on x.
3) The convex-hull formulation for the problem is then the following,
       scalar eps /1.0e-6/ ;
       variables x, y ;
       positive variables y, y1, y2, y3, x2, x3 ;
       negative variable x2 ;
       binary variables bin1, bin2, bin3 ;
       equations eq01, eq02, eq03, eq04, eq05
                 eq06, eq07, eq08, eq09, eq10
                 eq11 ;
       eq01..  y =e= y2 + y3 ;
       eq02..  x =e= x1 + x2 + x3 ;
       eq03..  x1 =l= -eps*bin1 ;
       eq04..  x1 =g= L*bin1 ;
       eq05..  y2 =e= k*bin2 - x2 ;
       eq06..  x2 =g= eps*bin2 ;
       eq07..  x2 =l= 0.5*k*bin2 ;
       eq08..  y3 =e= x3 ;
       eq09..  x3 =g= 0.5*k*bin3 ;
       eq10..  x3 =l= U*bin3 ;
       eq11..  bin1 + bin2 + bin3 =e= 1 ;
```

Note that

```
1) If bin1 = 1
      Then from eq11       => bin2=bin3=0
      from eq09 and eq10   => x3 = 0
      from eq08            => y3 = 0
      from eq06 and eq07   => x2 = 0
      from eq05            => y2 = 0
      from eq01            => y = 0                 <<<<<
      from eq02            => x = x1
      from eq03 and eq04   => L <= x <= -eps        <<<<<
      This models the first situation

2) If bin2 = 1
      Then from eq11       => bin1=bin3=0
      from eq09 and eq10   => x3 = 0
      from eq08            => y3 = 0
      from eq03 and eq04   => x1 = 0
      from eq01            => y = y2
      from eq02            => x = x2
      from eq06 and eq07   => eps <= x <= 0.5*k         <<<<<
      from eq05            => y = k - x                 <<<<<
      This models the second situation
3) If bin3 = 1
      Then from eq11       => bin1=bin2=0
      from eq03 and eq04   => x1 = 0
      from eq06 and eq07   => x2 = 0
      from eq05            => y2 = 0
      from eq02            => x = x3
      from eq01            => y = y3
      from eq09 and eq10   => 0.5*k <= x <= U         <<<<<
      from eq08            => y = x                   <<<<<
```

This models the third situation.

This formulation (based on the disaggregation approach proposed by Balas) is as tight as it gets. However, it does require more variables and equations. That trade-off needs to be kept in mind when deciding which of the two approaches to use when modeling disjunctions. The disaggregation method that I described above does not use bounds on functions, but on variables only. This makes it attractive when the functions involved are more complicated than (k-x) and x. Note that if these two functions were replaced by a+f(x) and b+g(x), only eq05 and eq07 would change to

```
   eq05..  y2 =e= a*bin2 + f(x2)
   eq07..  y3 =e= b*bin3 + g(x3)
```

where a and b are the constants in the functions. Note that no constraint of the form,

```
    f(x) <= M*bin
```

(a.k.a. the dreaded 'big-M') is used.

For those interested, the complete reference of the theoretical properties of the disaggregated convex hull formulation for disjunctions is Balas, E. (1985) Disjunctive Programming and a Hierarchy of Relaxations for discrete optimization problems. SIAM J. Alg. Disc. Meth., 6, 466-486

3.2 Nonlinear model solution problems

I have a problem running a particular model that we are developing. The model is nonlinear, based upon a dynamic simulation model representing the economic impact of environmental change. The original simulation model has about 60 time--dependent variables; we are trying to develop a similar optimization based model; discretizing the decision and state variables into several time periods and using forward differencing to represent the first order differential equations. Most of the constraint equations are nonlinear (ie individual variables with noninteger powers multiplied together eg of the type $B(t) = P(t)^0.8 *E(t)^0.2*N(t)$, as well as third and fourth order term equations eg of the type A(t) = B(t)*C(t)*D(t). We are using the MINOS solver running on a 386 PC., and using a time horizon of 10 periods (so there are 60 cts variables * 10 periods = 600 variables in the model). We have been able to get the model to converge to a local optimum, but this requires repeated solves, starting with a base formulation and solving; adding in more equations and resolving...; and doing this process a couple of times. HOWEVER; the model is (perhaps not unsurprisingly) quite touchy, and even small changes in parameters or initial conditions will cause the solver to either spit out infeasible/unbounded messages (usually mentioning that the basis appears to be singular), or getting quite close to convergence but terminating because of insufficient iterations (in this case, the screen documentation reports that NINF = 0, and this remains consistent; but the violation term cycles between about 5e0 and 2 e-2; getting down quite low then suddenly shooting up two orders of magnitude. Increasing the max number of iterations parameter does not help).So, my specific questions:

```
1) Any suggestions to make the model easier to solve?  Will replacing the equations
           B(t) = P(t)^0.8 *E(t)^0.2*N(t)
       by something like
    PP(t) = P(t)^0.8*E(t)^0.2
    B(t) = PP(t) * N(t)
       help the solver?
2) Are there any specific options that will be helpful to use in the MINOS options file?
3) Is there a FAQ regarding using GAMS to solve nonlinear models?
```

Answer from n.n:

Here is a dumb suggestion, but at sufficiently low-cost that it is still worth making: try the CONOPT solver. One of the joys of GAMS is that you can access several solvers for the same class of problems, for most classes, at least. MINOS and CONOPT have different strengths in NLP, and it is an easy switch (just add OPTION NLP=CONOPT in your code).

Answer from n.n:

I've been working on a model that has a similar structure and problems. A few suggestions:
1) Put upper and lower limits on variables based on outside knowledge.
2) Try using the alternative solver CONOPT.
3) Generate a "value" file from a run which you believe to be correct, and use this file to initialize every run.

Answer from n.n:

A couple of additional suggestions:
1) Include nonlinear constraints in the objective where possible;
2) Replace equalities by inequalities where possible especially
   if you obtain convexity (this may be possible depending on your

objective and may reduce oscillations between infeasibilities);

3) Do the variables P and E approach zero - if so does it make sense to bound them away from zero by an epsilon; and
4) Can you transform B to log(B)?

3.3 DEA example model

There is some very nice GAMS code written for DEA which is presented in a paper by Olesen and Petersen (A Presentation of GAMS for DEA) which appeared in Computers and Operations Research (1996, Vol 23 pp 323-339). This allows for either primal or dual approaches to be used and is easily adapted to fit any data. The code was at one time downloadable from the PARN (Productivity Analysis Research Network) web site (http://www.busieco.ou.dk/parn).

3.4 Different solutions with different GAMS versions

I noticed that different versions of GAMS give different results for the same input file. In my case I am running a quadratic model under MINOS 5. Without yet providing additional information on the versions and the input code I would like to ask you if this sounds familiar to anyone of you and how this problem can best be handled. If needed I will certainly give additional information.

Answer from adrud@arki.dk:

Do you know if your model can have more than one optimal solution (is it STRICTLY convex or has other nice properties)? Are the differences large or small (round-off errors)? Different versions of an NLP algorithms can give different results if the model has more than one solution. And if the optimum is very flat, then you may get very small differences in objective but fairly large differences in primal values. A very small change caused by new ordering methods or small round-off errors early in the computation can change the whole iteration path and lead to different solutions if there are multiple optima. Example: variable X1 and X2 have mathematically the same reduced cost in the initial point. In one version of the algorithm X1 will be selected to enter the basis because it is first. In another version the variables have been sorted according to some criteria and X2 appears to be first and it will be selected. From there on, everything is different.

3.5 Scaling the Hessian

I am working on a model with a non-linear objective function (2-level nested CES) and over 1600 non-linear variables and 160 linear constraints. I have been using GAMS/MINOS and CONOPT to solve it. All the variables are scaled to 1. But, both MINOS and

CONOPT algorithms still stop short of optimal solution due to slow convergence. The problem is, I think, despite the scaling, the Hessian is still ill-conditioned as evidenced by a condition number of around 38000. There is a discussion of how to scale the Hessian in a book "Practical Optimization" by Gill, et. al. My question is: Do any one of you have a practical experience scaling a Hessian, can it be done with GAMS/MINOS - CONOPT, or know of other algorithms that does it in a painless way? I feel comfortable using the base year values of the variables for the numerical calculations needed for the scaling.

Answer from m.saunders@auckland.ac.nz:

Models with linear constraints (LC models) should solve reliably if the objective is smooth and the constraint matrix is well scaled. Here are some things to think about:\\ It is good that you are concerned about scaling. Note that "All variables are around 1" should not be taken too literally. In general it means choosing reasonable units for everything, so that a TYPICAL variable of each type is around 1, and a TYPICAL coefficient in EACH ROW and EACH COLUMN of the CONSTRAINT MATRIX is around 1. Some variables want to be small or zero at a solution (that's fine). Similarly, some matrix coefficients really do have to be small. The main thing is that LARGE variables and matrix entries should be avoided. For LC models, Scale option 2 is probably best in MINOS if there is doubt about the scaling. (Among other things, it allows for big right-hand sides.) Scale option 0 is better if you KNOW that a model is well scaled.In general, try both. To check, turn on the MINOS print file and look at the output following the EXIT message. See if "norm x" and "norm pi" are much different with and without scaling. The size of the objective function can have an effect. MINOS allows for "large" objectives but not "tiny" ones. If all the shadow prices (the dual variables pi(i)) are small, scale the objective function up. In general you want a TYPICAL objective gradient to be around 1. Initialize nonlinear variables to realistic values. Does "Slow convergence" mean the linesearch keeps failing to find a better point? Non-smooth functions have that effect, so be sure that you bound the variables away from any singularities in the objective function. If there are any "ifs" and "buts" in the function (unlikely in GAMS), be sure there are no discontinuities. For NC models (nonlinear constraints), Scale option 2 uses the initial Jacobian (constraint gradients), so may not be safe unless you take great care with the initial values.Scale option 0 or 1 is safer in general It is hard to say more about your particular model without seeing the MINOS print file (with Print level 1 and Solution Yes). But try Scale option 2 without too much of your own scaling.

3.6 Fixed variables vs. params vs. equations ?

I use CONOPT2 with GAMS for solving a rather tough NLP problem (many nonlinearities in the constraints). I have a question about the recent mailing list discussion on PARAMETERS vs. fixed

VARIABLES.Arne Drud mentioned that CONOPT does not simplify the nonlinear equations (yet) when variables are fixed (even with HOLDFIXED set true). If so, does introducing equations of the type:

```
    EQ_XFIX('A-1').. X('A-1') =E= 100.0 ;
```

simplify my NLP? That is, does this replace the variable X('A-1') with the value 100.0 in my nonlinear equations and regard it as a parameter ? Does this formulation have any other drawbacks (except for the loss of marginals' output), for example on the performance ?

Answer from n.n:

The model attribute <modname>.holdfixed is relevant to the GAMS compiler only; if holdfixed is set on, then variables whose upper and lower bounds are equal are removed from the model before it is passed on to the solver (i.e. they are treated as parameters). Thus, in order to simplify the NLP (or any model, for any solver), you would want to set the variable bounds, for example:

```
X.fx('A-1') = 100.0;
```

In general, doing something like this is a good idea.The following example illustrates different ways of fixing variables.

```
* begin holdfixed.gms
SET J / 1 * 10 /;
PARAMETER a(J);
    a(J) = uniform(1,10);
VARIABLES
    x(J),
    z;
EQUATIONS
    f(J),
    objdef;
objdef .. z =e= sum{J, power(x(J)-4,2)};
f(J) .. x(J) =e= a(J);

model foo / objdef /;
model bar / objdef, f /;
option nlp = conopt;

* this model has many constraints, many free variables
* the solver has freedom in presolving/removing them (if it can) or not
* in the listing, the nonzero marginals are on the equation f(J)
solve bar using nlp minimizing z;

* this model has one constraint, many variables
* the solver has freedom in presolving/removing them (if it can) or not
* in the listing, the nonzero marginals are on the variables x(J)
x.fx(J) = a(J);
solve foo using nlp minimizing z;

* with holdfixed on, the solver sees only one constraint, one variable
* it doesn't even know the variable x exists
* in the listing, you won't see any marginals for x,
* since the solver never saw x
x.m(J) = 0;
foo.holdfixed = 1;
solve foo using nlp minimizing z;
* end holdfixed.gms
```

3.7 Solving a system of non-linear equations

```
How can I use GAMS to solve a system of non-linear equations,
like: fi(Xi) = 1   where: fi's are non-linear functions of xi's
and i = 1,..., n.
```

Answer from rutherford@colorado.edu:

```
There are three basic approaches are available for solving square
nonlinear systems under GAMS:
1) Formulate as a NLP (nonlinear program) with a irrelevant objective function:

        max anything
        s.t.            f_i(x) = 0     i=1,...,n

2) Formulate as an MCP (mixed compelementarity problem) without bounds:

                f_i(x) = 0    i=1,...,n
                -inf <= x_i <= +inf  i=1,...,n

3) Formulate as a CNS (constrained nonlinear system):

                f_i(x) = 0    i=1,...,n
                xlo_i <= x_i <= xup_i


Approaches (1) and (3) offer some advantages if the functions you
are using are undefined for some values of x.  You can then apply
upper and lower bounds which assure that the algorithm does not
wander off, but even with bounds you may not be assured of
finding solution if the functions are not nicely behaved
(monotone, P, etc.).

$TITLE  THREE METHODS FOR SOLVING NONLINEAR SYSTEMS WITH GAMS
*       VERY SIMPLE NONLINEAR SYSTEM:
SET     I       /I1*I10/;  ALIAS (I,J);
PARAMETER       SOLUTION        SOLUTIONS FROM ALTERNATIVE FORMULATIONS;
PARAMETER       A(I)    QUADRATIC PARAMETER
                C(I,J)  LINEAR PARAMETER
                B(I)    INTERCEPT  PARAMETER;
B(I)   = UNIFORM(0,1);
C(I,J) = UNIFORM(0,1);
A(I)   = UNIFORM(0,1);
VARIABLES       X(I)    UNKNOWN VECTOR X;
*       FUNCTION F DEFINES THE SYSTEM OF EQUATIONS WHICH
*       APPLY IN ALL  FORMULATIONS:
EQUATIONS       F(I)    CONSTRAINTS ON X;
F(I)..
      SUM(J, C(I,J) * X(J) + A(J) * X(J) * X(J)) - B(I) =E= 0;

*       (1) FORMULATION AS A CONSTRAINED NONLINEAR SYSTEM:
MODEL CNS_NLSYS /F/;
X.L(I) = 1;
SOLVE CNS_NLSYS USING CNS;
SOLUTION(I,"CNS") = X.L(I);

*       (2) FORMULATION AS AN MIXED COMPLEMENTARITY PROBLEM:
MODEL MCP_NLSYS /F.X/;
X.L(I) = 1;
OPTION MCP=MILES;
SOLVE MCP_NLSYS USING MCP;
SOLUTION(I,"MCP") = X.L(I);

*       (3) FORMULATION AS A NONLINEAR PROGRAM:
VARIABLE        OBJ     DUMMY OBJECTIVE;
EQUATION        OBJDEF  DEFINES THE DUMMY OBJECTIVE;
OBJDEF..        OBJ =E= 1;
X.L(I) = 1;
MODEL NLP_NLSYS /OBJDEF, F/;
SOLVE NLP_NLSYS USING NLP MAXIMIZING OBJ;
SOLUTION(I,"NLP") = X.L(I);
*       PRINT OUT A COMPARISON:
OPTION SOLUTION:8;
DISPLAY SOLUTION;
$ontext
The program should produce:
            CNS         MCP         NLP
I1  -0.82541120 -0.82541120 -0.82541263
I2   0.42506470  0.42506470  0.42506311
I3  -0.29439840 -0.29439840 -0.29439632
I4  -0.46446261 -0.46446261 -0.46446288
I5  -0.58034820 -0.58034820 -0.58035020
I6   0.16156815  0.16156815  0.16157303
I7  -1.00001459 -1.00001459 -1.00000972
I8   0.11500797  0.11500797  0.11500841
I9   0.44654506  0.44654506  0.44654524
I10  0.64782840  0.64782840  0.64783009
$offtext
```

3.8 GAMS code for GAMMA sampling

Answer from rutherford@colorado.edu:

```
I received a question about how to generate variates from
non-uniform, non-Normal distributions in GAMS.  Here is an
example of how to program a BATINCLUDE file for this purpose.
This example produces samples from a GAMMA distribution for
parameter values in the range of 0 to 1.
```

```
* --------------- cut here for GAMMAT.GMS ---------------
$title  Generate some GAMMA distributions and plot the density functions
set     sample  Define sample size                   /0*10000/,
        alpval  Alternative values of gamma parameter /"0.5","0.8"/,
        values  Range over which to plot sample density (0.0 to 3.0)
/0*30/;
*       First invocation must be outside a loop.  No
*       argument is provided here -- this is only to
*       get local gamma.gms parameters declared:
$BATINCLUDE gamma
parameter
        x(sample,*)             Pseudo-ransom samples,
        f(values,alpval)        Sample density,
        v(values)               Numeric values in sample range;
*       Set the seed if you want to make the sequence
*       reproducible:
option seed=1001;
*       Generate some sample distributions:
loop(sample,
$BATINCLUDE gamma 0.5
        x(sample,"0.5") = gamma_rgs
$BATINCLUDE gamma 0.8
        x(sample,"0.8") = gamma_rgs
);
*       Sort out the resulting distribution and plot it:
v(values) = 0.1*(ord(values)-1);
f(values,alpval) = 0;
loop(sample,
  f(values,alpval) = f(values,alpval) +
     (1/card(sample))$( (x(sample,alpval) gt v(values)    ) and
                        (x(sample,alpval) le v(values)+0.1) );
);
display f;
*       Let's have a look at the generated distributions:
$libinclude gnuplot f


* --------------- cut here for GAMMA.GMS ---------------
$stitle  GAMS code to generate a random variate from a GAMMA
distribution
*       This version only valid for 0 <= alp < 1.
*       See cited text for other generation methods.
$ontext
Adapted from Fortran function RGS (page 313) in Paul Bratley, Bennett
L. Fox and Linus E. Schrage: "A Guide to Simulation",
Springer-Verlag, 1983. (See this text for lots of other algorithms
coded in Fortran.)
Instructions on Using this Routine:
Initialize in a context where SCALAR parameters may be
declared, e.g. at the top of your program.  No argument may
be provided for initialization:         $batinclude gamma
Then make sampling calls:         $batinclude gamma alp
where alp is the disbution parameter, less than unity.
```

```
References for this algorithm (from Bratley et al.):
J.H. Ahrens and U. Dieter (1972), Computer Methods for Sampling from
GAMMA, BETA, POISSON and BINOMIAL Distributions, Computing, Vol. 12,
pp 223-246.
P.R. Tadikamalla and M.E. Johnson (1981), A Complete Guide to GAMMA
Variate Generation, Amer. J. of Math.  and Man. Sci., Vol. 1, pp
213-236.
$offtext
$if not declared gamma_u1 scalar gamma_u1, gamma_u2, gamma_u3, gamma_b,
gamma_p, gamma_x, gamma_rgs, gamma_alp, gamma_exit;
$if a%1==a $exit
abort$(%1 gt 1) "Invalid GAMMA distribution parameter:  %1";
gamma_alp = %1;
gamma_exit = 0;
while( not gamma_exit,
  gamma_u1 = uniform(0,1);
  gamma_b = (2.718281828 + gamma_alp) / 2.718281828;
  gamma_p = gamma_u1 * gamma_b;
  if (gamma_p le 1,
    gamma_x = exp(log(gamma_p) / gamma_alp);
    gamma_u2 = uniform(0,1);
    if (gamma_u2 le exp(-gamma_x),
      gamma_rgs = gamma_x;
      gamma_exit = 1;
    );
  else
    gamma_x = - log(( gamma_b - gamma_p ) / gamma_alp);
    gamma_u3 = uniform(0,1);
    if (not (log(gamma_u3) gt (gamma_alp-1)*log(gamma_x)),
      gamma_rgs = gamma_x;
      gamma_exit = 1;
    );
  );
);
```

3.9 Obtaining the Hessian

I would be grateful if someone could tell me of a method toobtain estimates of the Hessian matrix from Gams-Minos. It is clear from the"H condition" that the Cholesky decomposition is calculated, but is there a way to get the Hessian or preferably the diagonal Cholesky matrix printed out ?

Answer from m.saunders@auckland.ac.nz:

The "Hessian" that MINOS works with is just an approximation to the so-called "reduced Hessian", which we always write as Z'HZ for some strange rectangular matrix Z that we never compute explicitly. We maintain a dense triangular matrix R such that Z'HZ = R'R approximately. That's in the manual, but it doesn't answer your question. The quick answer is that you cannot print R. A quick question is -- what is it that you really want to know? R may not be anything like (the Cholesky factor of) the exact reduced Hessian, so I would never use it for anything other than getting the next search direction. Remember, if the problem has linear constraints, H would be the Hessian of the objective, but otherwise it's the Hessian of the Lagrangian. Which case do you have? It would be possible to add code to MINOS to estimate the true Z'HZ at the final point. Let's know what you would use it for. It is important to note that the internal information in the solver is not the Hessian itself (of the objective or the Lagrangian or an estimate thereof) but the REDUCED Hessian. REDUCED means that All nonbasic variables are absent, and all basic variables are absent. The rows and columns of the Reduced Hessian correspond to the SuperBasic variables. If we make the usual heroic (read: unrealistic) assumption that the model is not degenerate then the nonbasic variables are unique and they have nonzero reduced costs and not having Hessian information is probably OK. However, the set of superbasic variables is not unique. The solver can to a large extent exchange basic and superbasic variables, and the resulting Reduced Hessian changes accordingly. About ill conditioning etc: The Reduced Hessian can be ill conditioned because the Hessian of the Lagrangian is ill conditioned, but it is more likely that it is ill conditioned because the basis used to perform the reduction is ill conditioned. But inside the algorithm we usually do not know (MINOS has some smart routines that tries to select a good set of basic variables so the basis is conditioned). If we start to provide the reduced Hessian, should we also provide the Basis (or its inverse)? And/or the Real Hessian of the Lagrangian? And how do we return it? Do you want a printed 500 by 500 matrix or do you want it returned in a GAMS parameter, and in the latter case, what are the indices for the rows and columns, how are they related to the many symbolic variables in GAMS etc... If all this information is useful (I doubt it), then it may be more appropriate to add a diagnistic tool (a special "Solver") to GAMS that generates the appropriate information (may be selectable through an options file) in the current point.

Answer from rutherford@colorado.edu:

I believe that Michael was right when he suggested that a dump of the Jacobian would not really be that useful to you. As he pointed out, there is a serious problem with indexing -- you ask for the full matrix, but what names do you want to use for the rows and columns? Setting OPTION LIMROW=1000, LIMCOL=1000; is probably your best bet if you need to check a few entries. If you are truly keen on getting access to the Jacobian values in a format that you can do numerical analysis, you may wish to have a look at a paper and a "phantom solver" which Michael Ferris and I developed a couple of years ago. To use this, you need to formulate the model as a complementarity problem. This provides a means of accessing the function and Jacobian values generated by a GAMS model from within a MATLAB program. Of course all of the structural details of the model are suppressed, with rows and columns indexed as a single domain. See: Accessing Realistic Mixed Complementarity Problems within MATLAB Proceedings of Nonlinear Optimization and Applications Workshop, Erice June

1995, Plenum Press.

3.10 Modeling absolute Values

How do I formulate a model with absolute values?

Answer from rutherford@colorado.edu:

You cannot use the ABS function with a variable argument in a linear program. Although this is piecewise linear, it is not differentiable at zero and can therefore not be properly evaluated. There is, however, a well-known "trick" for representing ABS in an LP. This method is described in most MA-level texts on LP formulation. I can provide a brief introduction using a standard application

```
Given:
       XO(i,j) Reported basse year trade flow  from region i to j
        EO(i)   Actual base year exports from region i
        MO(j)   Actual base year imports into region j
        W(i,j)  Weights assigned for flow i to j
Objective:
     Minimize sum((i,j), W(i,j) * ABS(X(i,j)-XO(i,j)))
subject to:     SUM(j, X(i,j)) = EO(i)  Export Balance
                SUM(i, X(i,j)) = MO(j)  Import Balance
```

To do this using "standard" LP syntax, you need to introduce two additional non-negative variables and two constraints:

```
    POSITIVE VARIABLES      DP(i,j)         Positive deviation
                            DN(i,j)         Negative deviation;
    DPDEF(i,j)..    DP(i,j) =G= X(i,j) - XO(i,j);
    DNDEF(i,j)..    DN(i,j) =G= XO(i,j) - X(i,j);
```

You can then observe that at an optimum, we will have:

```
    DP(i,j) + DN(i,j) = ABS(X(i,j)-XO(i,j))
```

So, you can use an objective function:

```
    min SUM((i,j), W(i,j) * (DP(i,j)+DN(i,j)))
```

and the linear constraints as in the first formulation. The representation of the ABS function through a pair of non-negative variable is very widely used in solving practical problems. For example, you can use the ABS function to approximate quadratic terms in the objecitve, thereby making it possible to estimate solutions for quadratic programs which would otherwise be much larger than you could solve with NLP methods.

Answer from rrosenthal@monterey.nps.navy.mil:

One word of warning concerning the conversion of mathematical programs with absolute values into LP's. You must have a convex program to begin with. E.g., using Tom Rutherford's example, the weights w(i,j) must be nonnegative (which I'm sure they are in his case). If you had a negative w and you tried the trick, the LP would terminate unbounded. The trick falls apart in that case because the solver sees an objective function incentive to making both the positive and negative deviation variables positive at the same time. Similarly, it is easy to convert math programs that have constraints with absolute value terms to LP's, when the absolute value term has a positive coeficient and appears in a less-than-or-equal constraint. If you change the sign of the coefficient or sense of the inequality, the problem is nonconvex and unsolvable by LP. I am not sure that these warnings are as widely published as the tricks are.

3.11 Writing to Files Named by Set Text Labels

Answer from rutherford@colorado.edu:

Example -- GAMS Code Illustrating How to Produce Files with Names Indexed by the Text Labels of a Set. The following is from a GAMS program which runs under Windows 95/NT or DOS. It generates a bunch of files to be mailed to different students and a Unix script to mail the files.

```
*----------------------------------------------------------------------------------
*       Set S identifies students by 9-digit student number:
SET S  STUDENTS /
585986377       Norman
446686312       Patrick
219068570       John
...

*       Set address(s) is used to associate an email address with
*       each student:
SET ADDRESS(S)  Student Email addresses /
585986377       Joe@mars.edu
446686312       Pat@themoon.EDU,
219068570       Bryant@earth.EDU,
...
*----------------------------------------------------------------------------------
*...    I omit all the grading stuff -- because of the syntax for
*...    exceptions, I find GAMS much easier to use than a spreadsheet
*..     for computing marks.
*----------------------------------------------------------------------------------
*       Define the temporary file to spool messages:
file kout  / tomail.txt/;  kout.lw = 0;
*       Define a temporary batch file which is executed on the fly
*       to copy the temporary file to a file with a name defined by
*       the text elements for set s:
file kbat  / copym.bat/;
*       This is the mail command file to be appended with variable-width
*       set identifiers:
file kmail / mail\3070mail.cmd /;
```

```
kmail.lw=0;
kmail.ap = 1;

*       Loop over students (set s):
loop(s,
*       Write a message for student s into file "tomail.txt":
        put kout ’Economics 3070-004: Multiple Choice Problems for
Chapter %c%’/;
        put s.te(s),’ (’,s.tl,’)’/;
        put ’Your Score: ’,mark(s)/;
        ...
        putclose;
*       Use a DOS batch file to copy the message to a file named
*       with the student number:
        put kbat ’@echo off’/;
        putclose kbat ’copy tomail.txt mail\’,s.tl,’.txt >nul’/;
*       The following is an execution-time system call, as distinct
*       from $execute which is processed at compile time.  The
*       execution-time statement is only in later versions of GAMS
*       (2.25.088 or higher, I believe):
        execute ’copym ’;
*       Add the student to the Unix script file for posting:
        put kmail ’mail ’,address.te(s),’ <’,s.tl,’.txt’/;
);
```

3.12 Help with linearizing function

I am interested in linearizing the following function:

```
    x^2 + y^2 + z^2 >= a.
```

The following approximation is fine:

```
    |x| + |y| + |z| >= a’
```

However, implementing it (in the obvious way) requires three binary variables. I have tried doing this and the MIP runs for a day and does not produce a solution. Other sorts of approximations are also fine.

Answer from n.n:

You have a nasty (a.k.a., nonconvex) problem. (If the sense of the inequality were reversed, it would be easy.) Your feasible region has a hole in it, The hole is spherical if you use the quadratic constraint, diamond-shaped with the absoulute-value constraint. You can’t really avoid MIP when trying to navigate around these holes. (This sort of nonconvexity is very likely to become a trap for any linearization or other convex approximation, which you requested.) If the rest of your problem is linear, you may have a chance, but I would be careful about using the "obvious MIP," whatever that is. If it is nonlinear and you have just a few absolute-value terms, try the new nonlinear

MIP stuff. If it's a nonlinear MIP with a lot of integer variables

Answer from n.n:

Your problem with the sum of squares required to be greater than or equal to a value is a well studied problem. In the literature it is called a "reverse-convex programming problem." For obvious reasons. Many references are available but the main and first references belong to R. R. Meyer. J. B. Rosen has also studied this problem. It has solutions. However, what you sent us is really incomplete to make a proper recommendation as to what to do. I make an assumption that what you really want is to have the sum of the positive parts of the variables to be constrained to be greater than some value. I make this based upon your statement that you could just as well substitute absolute value for square. In that case your problem has a trivial solution. Replace each variable by its positive and negative parts: x = xp -xn where xp and xn are constrained to be non-negative.Similarly for y, z, etc. Then the constraint becomes: xp + xn + yp + yn + ... >= a You can verify that (because of linear dependence) only one of xp or xn will be nonzero and everything will come out correctly. Unfortunately, this trick won't work when you have any of the following:

1) Minimization objective cp * xp + cn * xn + ...
   where cp + cn < 0, (or equivalent max problem)
2) Greater-than-or-equal constraint with positive coeffs on
   variables(or equiv =L= constraint).

The linear independence of the xp and xn columns does not preclude both variables from being positive in an unbounded solution. The original non-convex model may have a finite opt, but our convex approx of it may be unbounded or its optimal solution may map back to an infeasible point in the original problem's space. In other words, you can't fool Mother Nature on a non-convex problem. (Sometimes, we might get lucky, but we can't always count on it.)

Answer from n.n:

Here a few of my thoughts. Please feel free to reply if they are wrong or stupid.These x^2 +y^2+z^2 >= c constraints are nasty. I would say forget about these. The formulation with the binary variables should work for small instances but the model I saw (with about 1000 binaries) could not find an integer solution after hours of CPU time. The first thing I would do is to scale back the model to a manageable size so one can play with it. Also the model I saw consisted of one long list of declaration and definitions of scalar variables and scalar equations. I would say: make first a small toy model (say with 50 to 100 binaries -- if possible easy to scale up), and make it really nice (you know, something that really makes you proud, a pleasure to read).Then I would drop these nasty equations and see what the model is doing. Now you get something like a "relaxed" solution. The one could try to use the heuristic: if x is positive, keep it positive if x is negative, keep it negative and then reintroduce |x|+|y|+|z|>c. This now becomes a simple linear contraint. This should solve easily. If the objective is close to the "relaxed" objective: be happy, and stop. If this is not the case, we could try again a MIP but now we have an incumbent solution. May be if we pass this on to the MIP it can generate quickly a few better integer solutions. If this is the case be happy, and stop. If this approach also fails, well we could try some kind of hybrid approach. Instead of forcing all variables to be positive or negative, use the heuristic:

```
if abs(x) > tolerance
if X < 0 keep x negative
if X > 0 keep x positive
else
use binary variable to model |x|
```

The smaller the constant "tolerance" the easier the model. Play with this tolerance and see what happens. Compare objectives, and see if you can scale up the model. Also play with OPTCR.

3.13 How do I model either or or conditional constraints?

I need to include the following constraint in an existing programming model:

y=x if x < 1, otherwise y=1

where x and Y are positive, continuous variables. Without the above constraint the existing large-scale model consists of mostly smooth nonlinear constraints and objective function and a few linear constraints. So far none of the variables is integer and the NLP model solves fine with MINOS. Including the above constraint as given and trying to solve as DNLP model with MINOS doesn't produce useful results. What is the best way to formulate the above constraint? Which solver is best for the resulting type of model?

Answer from n.n:

There is a dirty way to do this. Consider the following function:

```
    f1(x) = 2*x^30 + x
```

where x^30 is x raised to 30 !! This function is almost perfectly the same as the function f1(x) = x for 0 < x < 1, and then it shoots up to infinity at x = 1 almost instantaneously. In other words, f1(x) is what you want with the axes changed, so for this case you would put the constraint

```
x = 2*y^30 + y
```

to get what you want.There will be a small gap between this function and the actual discontinuous function around x = y = 1, but it can be decreased by increasing the power 30 or the coefficient 2 to a higher value. I'm sure MINOS or any other solver would have some trouble with this kind of high degree equality polynomial constraint, but it should be worth a try.

Answer from n.n:

The best formulation depend on the underlying reality. The simple attempt is to have the constraints

```
    c1 .. y =l= x;
    c2 .. y =l= 1; ( or y.up = 1 which is more efficient ).
```

If the objective will try to increase y then you will get the behaviour you want. If the objective tries to decrease y you will need a mixed-integer formulation with a binary variable that determines which of the two segments is active:binary variable z 0 if x less than 1 and 1 if x greater than 1;

```
    c1 .. y =l= x;
    c2 .. y =l= 1;
    c3 .. z =l= x;
    c4 .. z =l= y;
    c5 .. x - (xmax-1)*z =l= y;
```

(xmax is an upper bound on x, preferably a tight bound).To check the constraints:

```
- if z = 0 then (c5) x =l= y =l= x (c1), i.e. x =e= y;
 - if z = 1 then (c4) 1 =l= y =l= 1 (c2), i.e. y =e= 1,
   and x  -(xmax-1) =l= y =l= x which is not binding.
```

Answer from n.n:

This is in other words:

```
    y =e= min(x, 1);
```

One could model this with with binary variables, but then one would need a MINLP solver (like DICOPT++). In some cases you may be able to model it via::

```
    y =l= x;
    y.up = 1.0;
```

(for instance when maximizing y).

Answer from n.n:

There is another somewhat dirty way to do this using an exact penalty function: If the original problem is:

```
min f(x,y) subject to ...
```

Use

```
min f(x,y) + epsilon*s subject to ... and y = x - s y <= 1 s >= 0
```

where epsilon is a small positive number. How small? Again this depends on the nature of your problem. It should be between zero and the absolute value of the directional derivative of the optimal value function with respect to x. If the nature of your problem is such that this holds, then this is a very convenient formulation (for example if f(x,y) is strictly positive definite in x). Then you can verify that everything will come out OK and you will not have introduced any integer variables or complex functions. Of course, the marginal values won't be correct, but once you have a solution you can re run eliminating s and get the correct marginals.

3.14 A Team Scheduling problem

I have the following problem.I am trying to organize a schedule for a set of team games. Here are the rules:
- there are 7 teams: A,B,C,D,E,F,G
- a game requires 3 teams
- each team must play each other in all combinations exactly once

Obviously this requires 35 games. So.
- 5 games will be played each week, requiring 7 weeks each week,
- one team will play 3 times
- and the other 6 teams will each play two games

The team that plays 3 times must play each of the other teams the team that plays 3 times changes each week I can iterate all of the required games, but I cannot come up with an algorithm that will solve this problem. Any takers?

Answer from n.n:

Look at this example:

```
set t teams /a,b,c,d,e,f,g/;
set n game number /game-1*game-5/;
set w week number /week-1*week-7/;

alias (t,t1,t2,t3,s);
set g(t,t1,t2) games ;
g(t,t1,t2)$((ord(t) lt ord(t1)) and
(ord(t1) lt ord(t2))) = yes;
* g now contains only tuples (t,t1,t2) such that
* there are no duplicates (i.e. a.b.c and c.b.a)
* and no repetations (i.e. a.b.a). You can check
* that the cardinality of this set (card(g)) is
* indeed 35.

set isg(t, t1,t2,t3);
```

```
isg(t, t ,t2,t3)$g(t ,t2,t3) = yes;
isg(t, t1,t ,t3)$g(t1,t ,t3) = yes;
isg(t, t1,t2,t )$g(t1,t2,t ) = yes;

* isg(t, t1,t2,t3) provides a mapping
* between team t and all the games
* (t1,t2,t3) it plays.
* For instance if one would like to
* know which games are played by team d,
* use something like:
*
* set gd(t1,t2,t3);
* gd(t1,t2,t3)$isg('d', t1,t2,t3) = yes;
* display gd;

set map(t,w);
map(t,w)$(ord(t) eq ord(w)) = yes;

* we let team a play 3 times in week 1
* etc. This means we have a direct mapping
* between weeks and teams. This mapping
* map takes care of that.

variables
x(t1,t2,t3,w) schedule
dummy
;

binary variables x;

* x is declared over (t1,t2,t3, w) but we will
* use it only over (g, w). I.e. there are
* card(g)*card(w) binary variables.

equations
obj just a dummy objective
vert(t1,t2,t3) assignment constraints
hori(w) assignment constraints
count(t,w) two or three games each week
forbid(t,w,s)
;
obj.. dummy =e= 1;

vert(g).. sum(w, x(g,w)) =e= 1;
hori(w).. sum(g, x(g,w)) =e= 5;

* these are normal assignment constraints: assign
* games to weeks.

count(t,w).. sum(g$isg(t,g),x(g,w) ) =e= 2 + 1$map(t,w);

* 3 games for the team selected for this week (via map)
* and 2 games for all the other teams
```

```
forbid(t,w,s)$((not map(t,w))*map(s,w))..
        sum(g$(isg(s,g)*isg(t,g)), x(g,w))=e=1 ;

* the team that plays 3 times in week w is team s. Its opponents are t. We
* have to make sure that s sees all t during this week.

model m /all/;

option iterlim=10000;
solve m using mip minimizing dummy;

* display results in nicer format
parameter r(w, t1, t2, t3);
r(w,g) = x.l(g,w);
option r:1:0:1;
display r;
```

3.15 RASing a matrix

A recent request to the GAMS group asked for code samples for RASing a matrix. Below is a standalone program that (in this case) RASes a consumption flows matrix. With appropriate modifications, it can be used for square matrices (IO tables or SAMs) as well. I use it frequently for matrices up to 30x30 without any difficulty at all.

```
$TITLE  PROGRAM TO CARRY OUT RAS OF CONSUMPTION SHARES
$OFFSYMLIST OFFSYMXREF OFFUPPER
* Program requires specification of row and column indices
* If square matrix, only one set need be provided,
*  and ALIAS statements changed
 SETS
 I   SECTORS                / FarmFood    Farm Food Crops
                              Nonfood     Farm Nonfood Crops
                              Livestck    Livestock
                              Forestry    Forestry
                              Fishery     Fishery
                              OilGas      Coal Gas & Oil
                              OthMines    Other Mining
                              Foodproc    Food  Bev  & Tobacco
                              Textiles    Textiles & Leather
                              WoodFurn    Wood & Furniture
                              PapPrint    Paper & Printing
                              ChemRef     Chemicals & Fertilizers
                              RefinLNG    Refining LNG &  Coal
                              Nometmin    Nonmetallic Mineral
                              BasicMet    Basic Metals
                              Machines    Metal Prod & Machines
                              OthIndus    Other Industry
                              Utility     Elect Gas & Water
                              Construc    Construction
                              TradStor    Trade & Storage
                              RestHot     Restaurants & Hotels
```

```
                               LandTran    Rail & Road Transpor
                               OthTrans    Sea Air Trans & Comm
                               FinServ     Financial Serv & Insur
                               BusServ     Real Estate
                               PubAdmin    Public Admin
                               OthServ     Social & Other Serv/

 HH  HOUSEHOLD TYPE          / rurwork     Ag worker households
                               rurcap      Ag owners households
                               urbwork     Low wage households
                               urbcap      High wage households/

 ;

*  Program requires three different data inputs:
*    CONFLOW: the matrix of original flows
*    C0:      column containing row sum controls
*    CON:     column containing column sum controls

 TABLE CONFLOW(i,hh) INITIAL PRIVATE CONSUMPTION FLOWS

            rurwork  rurcap    urbwork   urbcap

FarmFood    165.56   5356.08   1331.07   2757.31
Nonfood     149.77    410.60    154.18    357.37
Livestck    141.53    905.99    501.23   1904.72
Forestry     56.70    141.65     55.69     56.52
Fishery     170.94    849.68    394.32    932.19
OilGas         .00       .00       .00       .00
OthMines       .22       .86       .29       .56
Foodproc   1169.48   5245.54   2777.09   6066.52
Textiles     82.61    529.25    279.48    806.17
Woodfurn     18.70    125.78     63.12    249.40
PapPrint      5.10     54.76     33.10    151.39
ChemRef      47.13    432.05    299.22    844.25
RefinLNG     56.26    515.80    357.23   1007.92
Nometmin      2.33     21.38     14.80     41.75
BasicMet       .00       .00       .00       .00
Machines     44.89    481.86    291.23   1332.10
OthIndus      3.67     39.38     23.80    108.85
Utility       5.63     51.97    119.31    413.36
Construc       .00       .00       .00       .00
TradStor      3.20     18.64     39.30    113.54
RestHot     108.44   1045.05   1061.97   2564.35
LandTran     42.85    313.51    600.14   1737.71
OthTrans     13.25    147.42    277.10    994.59
FinServ      23.17    300.03    136.31    589.15
BusServ      87.98    779.65    700.31   1979.36
PubAdmin       .00       .00       .00       .00
OthServ     116.65    853.46    987.11   4353.09
   ;

PARAMETER c0(i)  Control vector ;
```

```
c0("FarmFood") = 6394.42 ;
c0("Nonfood")  =  990.61 ;
c0("Livestck") = 2661.24 ;
c0("Forestry") =  259.31 ;
c0("Fishery")  = 1569.93 ;
c0("OilGas")   =     .00 ;
c0("OthMines") =     .21 ;
c0("Foodproc") =13911.94 ;
c0("Textiles") = 1367.44 ;
c0("Woodfurn") =  329.88 ;
c0("PapPrint") =  161.59 ;
c0("ChemRef")  = 1160.43 ;
c0("RefinLNG") = 1385.44 ;
c0("Nometmin") =   57.41 ;
c0("BasicMet") =     .00 ;
c0("Machines") = 1419.86 ;
c0("OthIndus")=   116.05 ;
c0("Utility") =   523.35 ;
c0("Construc")=      .00 ;
c0("TradStor")=  5737.68 ;
c0("RestHot") =  4553.91 ;
c0("LandTran")=  3552.82 ;
c0("OthTrans")=  1607.77 ;
c0("FinServ") =   883.65 ;
c0("BusServ") =  3287.49 ;
c0("PubAdmin")=      .00 ;
c0("OthServ") =  9063.46 ;


PARAMETER CON(hh)  AGGREGATE CONSUMPTION LEVELS ;

  CON("rurwork") = 2516.05  ;
  CON("rurcap")  = 18620.37 ;
  CON("urbwork") = 10497.40 ;
  CON("urbcap")  = 29362.18 ;

 ALIAS(I,RR) ;
 ALIAS(HH,CC) ;

PARAMETER  a0(rr,cc)      Initial coefficients matrix to RAS
           a1(rr,cc)      Final coefficients matrix after RAS
           rasmat0(rr,cc) Initial flows matrix to RAS
           ct(cc)         RAS column control totals
           rt(rr)         RAS row control totals
           ratio          Adjustment parameter on control totals
           checkcol       Check sum of column control totals
           checkrow       Check sum of row control totals
           sumccc         Original column sums of RAS matrix
           sumrrr         Original row sums of RAS matrix
    ;

VARIABLES
           DEV            Deviations
           RASMAT(rr,cc)  RASed matrix
```

```
          R1(rr)          Rho of RAS matrix
          S1(cc)          Sigma of RAS matrix
          LOSS            Objective (loss) function value
    ;

* Parameter initialization

   sumccc(cc)     = SUM(rr, conflow(rr,cc) ) ;
   sumrrr(rr)     = SUM(cc, conflow(rr,cc) ) ;
   a0(rr,cc)      = conflow(rr,cc) / sumccc(cc) ;
   rasmat0(rr,cc) = a0(rr,cc) * CON(cc) ;
   ct(cc)         = CON(cc) ;
   rt(rr)         = c0(rr) ;
   ratio          = SUM(rr, rt(rr)) / SUM(cc, ct(cc)) ;
   ct(cc)         = ct(cc) * ratio ;
   checkcol       = SUM(cc, ct(cc) );
   checkrow       = SUM(rr, rt(rr) );

   display ratio, checkcol, checkrow ;

   display conflow, a0 ;

   display con, ct ;
   display c0, rt ;

* Variable initialization

     DEV.L           =  0.0 ;
     R1.L(rr)        =   1 ;
     S1.L(cc)        =   1 ;
     RASMAT.L(rr,cc) =  a0(rr,cc) * ct(cc) ;

     CON(cc)  =  ct(cc)  ;

EQUATIONS

          BIPROP(rr,cc)   Bi-proportionality for RAS matrix
          DEVSQ           Definition of squared deviations
          OBJ             Objective function
          RCONST(rr)      Row constraint
          CCONST(cc)      Column constraint
   ;

    BIPROP(rr,cc).. RASMAT(rr,cc) =E= R1(rr)*S1(cc)*rasmat0(rr,cc) ;

    CCONST(cc)..    ct(cc) =E= SUM(rr, RASMAT(rr,cc)) ;

    RCONST(rr)..    rt(rr) =E= SUM(cc, RASMAT(rr,cc)) ;

    DEVSQ..         DEV =E= SUM( (rr,cc)$rasmat0(rr,cc),
                    SQR( (RASMAT(rr,cc) - rasmat0(rr,cc)) / rasmat0(rr,cc)) ) ;

    OBJ..           LOSS =E= SUM(rr, R1(rr)**2 + (1/R1(rr))**2 )
                           + SUM(cc, S1(cc)**2 + (1/S1(cc))**2 )  ;
```

```
* Variable bounds

    RASMAT.LO(rr,cc) = 0.0 ;
    R1.LO(rr)        = 0.01 ;
    S1.LO(cc)        = 0.01 ;

MODEL  CONSUMERAS   / BIPROP
                      CCONST
                      RCONST
*                     DEVSQ
                      OBJ /  ;
*DEVSQ is commented out
OPTIONS ITERLIM=10000,LIMROW=0,LIMCOL=0,SOLPRINT=OFF ;
SOLVE CONSUMERAS USING NLP MINIMIZING LOSS ;
display rasmat.l, r1.l, s1.l ;
a1(rr,cc) = rasmat.l(rr,cc) / ct(cc) ;
display a0, a1 ;
```

3.16 Chip Design Problem

I am a new user in GAMS software am solving a problem to fold rows and columns of a chip to reach minimal possible area (~ 25%) if I can successfully fold columns and rows. I have some results in column folding, but applying rows is then more difficult.

```
               example                   approx. solution
        A B C D E F G H  F1              A B C D  F1
     1  x     x                       1  x     x
     2    x x x           x           2    x x x  x
     3      x     x       x           3  x   x    x
     4        x x x       x           4  x x   x  x
     5          x     x               5    x x
     6            x x                 6  x     x
                                         F E H G
```

x-is connecting. Probably you have ideas,which could be helpfull to me.

Answer from n.n:

This turned out to be a nasty model. First I thought I could do row and column placement simultaneously. For this I introduced the (binary) parameter P(I,J). The row, and column placements are R(i,ii) and C(j,jj). So if I understand the problem, the model would become (I think):

```
SET I  rows   / 1,2,3,4,5,6 /;
SET J  colums / A,B,C,D,E,F,G,H,F1/;
TABLE P(I,J) initial configuration
            A B C D E F G H  F1
         1  1     1
         2    1 1 1           1
         3      1     1       1
         4        1 1 1       1
         5          1     1
         6            1 1
;

ALIAS (I,II);
ALIAS (J,JJ);
VARIABLES
   Q(I,J)   final configuration
   R(I,II)  row placements
   C(J,JJ)  column placements
   Z        approximation for area
;
BINARY VARIABLES R,C;
EQUATIONS
   ROW(I)
   COL(J)
   QDEF(I,J)
   OBJ;

ROW(I).. SUM(II, R(I,II)) =E= 1;
COL(J).. SUM(JJ, C(J,JJ)) =E= 1;

QDEF(I,J).. Q(I,J) =E= SUM((II,JJ), R(II,I)*C(JJ,J)*P(II,JJ));
* I would like the real area here, but for now I have just
* a simple increasing penalty for large i's and j's
* make sure there is a term for each q(i,j), otherwise
* yy is not correct
OBJ.. Z =E= SUM((I,J), (ORD(I)*ORD(J))*Q(I,J));
* not more than 1 item on each place
Q.UP(I,J) = 1;
```

Q would give us the final configuration. QDEF is nonlinear. Replacing R(II,I)*C(JJ,J) by a variable which is 1 if r and c is 1 is possible but this would give you a variable of the form DELTA(II,I,JJ,J). In fact I tried this, and I came up with 8800 constraints. A little bit large for such a small toy problem. In the end I came up with:

```
SET I  rows   / 1,2,3,4,5,6 /;
SET J  colums / A,B,C,D,E,F,G,H,F1/;
TABLE P(I,J) initial configuration
            A B C D E F G H  F1
         1  1     1
         2    1 1 1           1
         3      1     1       1
         4        1 1 1       1
         5          1     1
         6            1 1
;
ALIAS (I,II);
ALIAS (J,JJ);
```

```
VARIABLES
Q(I,J)   final configuration
R(I,II)  row placements
C(J,JJ)  column placements
DELTA(II,I,JJ,J)  do we really need these
Z        approximation for area
;
BINARY VARIABLES R,C;
EQUATIONS
   ROW(I)
   COL(J)
   QDEF(I,J)
   OBJ;
ROW(I).. SUM(II, R(I,II)) =E= 1;
COL(J).. SUM(JJ, C(J,JJ)) =E= 1;
* this what I want, but this is non-linear
*QDEF(I,J).. Q(I,J) =E= SUM((II,JJ), R(II,I)*C(JJ,J)*P(II,JJ));

* this is a stupid way of linearizing
* we end up with 8818 equations and 3088 variables (117 integer)
* delta is automatically integer
*VARIABLES DELTA(II,I,JJ,J);
* DELTA.UP(II,I,J) = 1;
* DELTA.LO(II,I,J) = 0;
*EQUATIONS
*   DELTADEF1(II,I,JJ,J)
*   DELTADEF2(II,I,JJ,J)
*   DELTADEF3(II,I,JJ,J);
*QDEF(I,J).. Q(I,J) =E= SUM((II,JJ), DELTA(II,I,JJ,J)*P(II,JJ));
*DELTADEF1(II,I,JJ,J).. DELTA(II,I,JJ,J) =L= R(II,I);
*DELTADEF2(II,I,JJ,J).. DELTA(II,I,JJ,J) =L= C(JJ,J);
*DELTADEF3(II,I,JJ,J).. DELTA(II,I,JJ,J) =G= R(II,I)+C(JJ,J)-1;

* slightly more clever way of doing this
VARIABLES YY(II,I,J), Y(II,J);
EQUATIONS YYLO(II,I,J), YDEF(II,J);
QDEF(I,J).. Q(I,J) =E= SUM(II, YY(II,I,J));
* now I want:
* YYDEF(II,I,J).. YY(II,I,J) =E= R(II,I)*Y(II,J);
SCALAR M max of y;
M = CARD(JJ);
YYLO(II,I,J) .. YY(II,I,J) =G= Y(II,J) - M*(1-R(II,I));
YY.LO(II,I,J)=0;
YDEF(II,J).. Y(II,J) =E= SUM(JJ, C(JJ,J)*P(II,JJ));
* I would like the real area here, but for now I have just
* a simple increasing penalty for large i’s and j’s
* make sure there is a term for each q(i,j), otherwise
* yy is not correct
OBJ.. Z =E= SUM((I,J), (ORD(I)*ORD(J))*Q(I,J));
* not more than 1 item on each place
Q.UP(I,J) = 1;
MODEL MOD /ALL/;
OPTION ITERLIM=100000;
MOD.WORKSPACE=2;
```

```
SOLVE MOD USING MIP MINIMIZING Z;
DISPLAY Q.L;
```

3.17 Obtaining Eigenvalues

Has any one out there ever solved for the eigen values of a matrix using GAMS/MINOS?

Answer from n.n:

I assume that you are interested in the eigenvalues (and possibly
the eigen vectors) of a SYMMETRIC matrix. You may write all the
equations that define the eigenvectors, but most likely MINOS (or
any other NLP solver) will not be able to solve the model because
it is highly non linear and non-convex. However, you may compute
the eigen values one by one by maximizing a norm. The following
little GAMS program should show you the principle:

```
set i dimension of the symmetric matrix / i1 * i6 /;
alias (i,ip,ipp);
set io(i) subset with already defined eigenvectors;
parameter c(i,i)  matrix for which eigenvalues are computed
          ev(i,i) eigenvectors
          la(i)   eigenvalues;
*  Generate random positive semidefinite symmetric matrix, c.
*  The rank will be less than min( card(i), card(o) ):
*
set o / o1 * o6 /;
parameter d(i,o) data observations;
d(i,o)  = normal(0,1);
c(i,ip) = sum(o, d(i,o) * d(ip,o) );
display c;
*
*  Make sure io and ev are defined before we enter the loop
*
io(i)    = no;
ev(i,io) = 0;
variable x(i)      eigenvector
         lambda    eigenvalue;
equation edef      definition of max eigenvalue
         norm      eigenvector has norm 1
         ortho(i)  eigen vector is orthonormal to already defined vectors;
edef      .. sum( (i,ip), x(i) * c(i,ip) * x(ip) ) =e= lambda;
norm      .. sum( i, sqr( x(i) ) ) =e= 1;
ortho(io) .. sum( i, ev(i,io) * x(i) ) =e= 0;
model eigen / all /;
option nlp = conopt;
option limrow = 0, limcol = 0, solprint = off;
*  Now generate the eigen vectors one by one starting from the largest
*  eigenvalue. Each new eigen vector must be orthogonal to the previous
*  ones and maximize the weighted norm.
*  If maximizing is replaced by minimizing we get them in the reverse
*  order.
loop( ipp,
* Initialize with random vector and normalize it so the only nonlinear
* constraint is satiesfied:
x.l(i) = normal(0,1);
x.l(i) = x.l(i) / sum( ip, sqr( x.l(ip) ) );
*   solve eigen using nlp maximizing lambda;
*  Store the current eigen vector and eigen value and increment the
*  set of vectors that future vectors must be orthogonal to:
   ev(i,ipp) = x.l(i);
   io(ipp) = yes;
   la(ipp) = lambda.l;
);
display ev, la;
*
*  Test the accuracy of the solution:
*  If dtest below is too large you may reduce the optimality tolerance.
*  In CONOPT this is done with the line "set rtredg 1.d-10" in a conopt
*  options file, conopt.opt.
*
parameter test(i,i)  computed tr(ev) * diag(la) * ev -- should be equal to c
          dtest(i,i) error in test;
test(i,ip) = sum( ipp, ev(i,ipp) * la(ipp) * ev(ip,ipp) );
dtest(i,ip) = test(i,ip) - c(i,ip);
display test, dtest;
```

3.18 Stochastic optimization - an example

Do you have a simple example how to use GAMS to do stochastic optimzation?

Answer from rrosenthal@monterey.nps.navy.mil:

At the risk of boring many of the list members, I will attach a very simple stochastic program implemented in gams. It is Linus Schrage's snow removal example, which I have found to be a nice teaching example. The exact reference to Schrage's book is included.

```
$TITLE STOCHASTIC PROGRAMMING: SCHRAGE'S SNOW REMOVAL PROBLEM
  $offupper offsymxref offsymlist onlisting inlinecom { }
  $ontext
    Two-Stage Stochastic Programming Problem with Recourse
    Source:  Linus Schrage, LINDO, 3rd ed., p.140-145
    By:  Richard E. Rosenthal (Oct 91)
  $offtext
  SETS
      R  consumable resources  / FUEL, SALT /
      A  activities            / PLOWING, SALTING /
      S  states of nature      / WARM, COLD /
      E  durable resources     / TRUCK-DAYS /
      ;
  PARAMETER  C1(R)  cost of buying resources in Stage 1
      /
        FUEL    70
        SALT    20
      / ;
```

```
TABLE  C2(S,R)  cost of buying resources in Stage 2
                 FUEL    SALT
      WARM         73      30
      COLD         73      32
      ;
PARAMETER  SALV(R)  salvage value of resources after Stage 2
    /
      FUEL    65
      SALT    15
    / ;
TABLE  OPCOST(S,A)  operating cost in Stage 2
              PLOWING   SALTING
      WARM        110       110
      COLD        120       120
      ;
TABLE  EFF(S,A)  efficiency factor for snow removal activity
              PLOWING   SALTING
      WARM        1.0       1.2
      COLD        1.0       1.1
      ;
TABLE  CON(R,A)  consumption rates for snow removal activity
              PLOWING   SALTING
      FUEL        1.0       1.0
      SALT        0.0       1.0
      ;
PARAMETER  DEM(S)  demand for snow removal
    /
      WARM    3500
      COLD    5100
    / ;
PARAMETER  PROB(S)  probability of Stage 2 states of nature
    /
      WARM    0.4
      COLD    0.6
    / ;
PARAMETER  SUP(E)  equipment supply
    /
      TRUCK-DAYS    5000
    / ;
POSITIVE VARIABLES
     X1(R)      Stage 1 resource purchases
     X2(S,R)    Stage 2 resource purchases under SON s
     Y2(S,A)    Stage 2 activities under SON s
     W2(S,R)    Stage 2 salvages under SON s
     ;
FREE VARIABLE
     Z          total expected cost ;
EQUATIONS
     BAL(S,R)     Stage 2 resource balance under SON s
     EQUIP(S,E)   Stage 2 equipment supply under SON s
     DEMAND(S)    Stage 2 snow removal demand under SON s
     OBJDEF
     ;
BAL(S,R)..
```

```
  *     Purchases      =   Salvage  +   Consumption
     X1(R) + X2(S,R)  =E=  W2(S,R)  +  SUM(A, CON(R,A) * Y2(S,A)) ;
  EQUIP(S,E)..
     SUM(A, Y2(S,A) )  =L=  SUP(E) ;
  DEMAND(S)..
     SUM(A, EFF(S,A) * Y2(S,A) )  =G=  DEM(S) ;
  OBJDEF..
       SUM( R, C1(R) * X1(R) )
    +  SUM( (S,R), PROB(S) * C2(S,R) * X2(S,R) )
    +  SUM( (S,A), PROB(S) * OPCOST(S,A) * Y2(S,A) )
    -  SUM( (S,R), PROB(S) * SALV(R) * W2(S,R) )
       =E=   Z ;
   OPTION LIMROW=10, LIMCOL=0, SOLPRINT=ON
   MODEL SNOW  / ALL / ;
   SOLVE SNOW USING LP MINIMIZING Z;
```

3.19 Data aggregation with GAMS

Answer from n.n:

At the recent Waterloo Conference on General Equilibrium Analysis, I spoke with someone who was a novice GAMS programmer. He felt that GAMS was fine for models but not very helpful for working with data. One difficulty he noted was aggregation -- this seemed to him something for which FORTRAN was the natural approach. I disagree. To illustrate, I offer a short example of how to do data aggregation with GAMS. Assume that we have a two-dimensional parameter of disaggregate data, defined over sets I and J. For concreteness, I will use a reproducible pseudo-random dataset:

```
 OPTION SEED=10001;
* Declare some sets:
SET     I       Row index in disaggregate data set /I1*I10/
        J       Col index in disaggregate data set /J1*J10/;
PARAMETER A(I,J)  Disaggregate data (randomly generated);
* Generate a random matrix with 50% sparsity:
 LOOP((I,J)$(UNIFORM(0,1) GT 0.5),  A(I,J) = UNIFORM(0,1); );
* In any aggregation excercise, it is necessary to specify how
* the underlying sets will be aggregated.  I always do this using
* two-dimensional subsets ("tuples"), one for each dimension to be
* aggregated.  For the example, I need two such mappings:
SET     K       ROW INDEX IN AGGREGATE DATA SET /A,B,C,D/,
       L       COL INDEX IN AGGREGATE DATA SET /E,F,G/,
       RMAP(I,K)       ROW MAP ASSOCIATES ELEMENTS OF I WITH ELEMENTS FROM K
        /       (I1,I4*I7,I10).A,
               (I2,I3).B,
               I8.C,
               I9.D /
        CMAP(J,L)       COL MAP ASSOCIATES ELEMENTS OF J WITH ELEMENTS FROM L
        /       (J1*J8).E,
               J9.F,
               J10.G/;
* Check for two types of errors in the mapping:
```

```
* (i) An element of the disaggregate set is not assigned or assigned
* more than once.
* (ii) An element of an aggregate set has no assigned members.
LOOP(I,  ABORT$(SUM(K$RMAP(I,K),1) NE 1) " Type (i) error in RMAP.",  RMAP;);
LOOP(K,  ABORT$(SUM(I$RMAP(I,K),1) EQ 0) " Type (ii) error in RMAP.", RMAP;);
LOOP(J,  ABORT$(SUM(L$CMAP(J,L),1) NE 1) " Type (i) error in CMAP.",  CMAP;);
LOOP(L,  ABORT$(SUM(J$CMAP(J,L),1) EQ 0) " Type (ii) error in CMAP.", CMAP;);
* Declare the matrix to hold the aggregated data:
PARAMETER B(K,L)  Aggregated matrix;
* Do the aggregation (all the aggregation logic is here):
B(K,L) = SUM((I,J)$(RMAP(I,K)*CMAP(J,L)), A(I,J));
DISPLAY B;
* Tony Brooke programmed the pseudo-random number facility to
* produce consistent sequences on all platforms.  The following
* output should be generated on any computer running GAMS:
* ----     80 PARAMETER B             Aggregated matrix
* E           F           G
* A      10.609       1.986       0.670
* B       4.553                   0.709
* C       1.270
* D       3.035
* It is often helpful to save the aggregated data for use in another
* GAMS program.  The clumsy way to do this is to edit the listing file
* (ugh!).  A much easier method is to use the GAMS PUT facility in
* order to output the data in GAMS-readable form.  For this purpose I
* have written a $BATINCLUDE "subroutine" named wrtprm.inc.  Here is
* how it is used:
* Declare a file to which the data will be saved:
FILE KOUT /aggdata.gms/;
* Set the "append" flag so that parameter B is added to this file
* (If you omit this statement, then aggdata.gms will be overwritten.):
KOUT.AP = 1;
* Use the "standard" subroutine to output the data in a GAMS-readable format:
$BATINCLUDE wrtprm.inc KOUT B K L
* After executing this program, aggdata.gms should contain:
* PARAMETER B /
* A.E  1.06089500E+01
* A.F  1.98581249E+00
* A.G  6.69861822E-01
* B.E  4.55318796E+00
* B.G  7.09486162E-01
* C.E  1.26962683E+00
* D.E  3.03485903E+00
* /;
```

```
*--------------------- wrtprm.inc (cut here) -----------------------
$OFFLISTING
*       WRITE A SCALAR OR ARRAY IN GAMS-READABLE FORMAT
*       THIS "SUBROUTINE" HAS THE FOLLOWING ARGUMENT LIST:
*              %1       FILE IDENTIFIER
*              %2       PARAMETER TO BE WRITTEN
*              %3-%9    UP TO 7 SET ARGUMENTS (I.E. %2(%3,...,%9))
```

```
*
*       SET UP THE FILE FOR SCIENTIFIC NUMERIC FORMAT:
%1.LW = 0; %1.NR = 2; %1.NW = 16; %1.ND = 8;
*       IF %3 IS MISSING, WE OUTPUT A SCALAR:
$IF NOT A%3==A $GOTO BEGIN
PUT %1 'SCALAR %2 /'%2'/;'/;
$GOTO EXIT
*       OUTPUT A VECTOR BY FIRST WRITING THE PARAMETER STATEMENT
*       (WITHOUT EXPLICIT DOMAIN FOR SIMPLICITY) AND THEN LOOP THROUGH
*       ALL THE NONZEROS:
*
$LABEL BEGIN
SCALAR STARTED; STARTED = 0;
PUT %1 'PARAMETER %2 ';
$IF A%9==A $GOTO LESS7
*       A 7-DIMENSIONAL PARAMETER:
LOOP((%3,%4,%5,%6,%7,%8,%9)$%2(%3,%4,%5,%6,%7,%8,%9),
IF ((NOT STARTED),
  PUT '/'/;
  STARTED = 1;
 );
PUT %3.TL,'.',%4.TL, '.',%5.TL,'.',%6.TL,'.',%7.TL,'.',%8.TL,'.',%9.TL,
        %2(%3,%4,%5,%6,%7,%8,%9)/;
);
IF (STARTED,
  PUT '/;'/;
ELSE
  PUT '; %2(%3,%4,%5,%6,%7,%8,%9) = 0;'/;
);
$GOTO EXIT
$LABEL LESS7
$IF A%8==A $GOTO LESS6
*       A 6-DIMENSIONAL PARAMETER:
LOOP((%3,%4,%5,%6,%7,%8)$%2(%3,%4,%5,%6,%7,%8),
 IF ((NOT STARTED),
   PUT '/'/;
   STARTED = 1;
 );
PUT %3.TL,'.',%4.TL, '.',%5.TL,'.',%6.TL,'.',%7.TL,'.',%8.TL,
                %2(%3,%4,%5,%6,%7,%8)/;
);
IF (STARTED,
  PUT '/;'/;
ELSE
  PUT '; %2(%3,%4,%5,%6,%7,%8) = 0;'/;
);
$GOTO EXIT
$LABEL LESS6
$IF A%7==A $GOTO LESS5
*       A 5-DIMENSIONAL PARAMETER:
LOOP((%3,%4,%5,%6,%7)$%2(%3,%4,%5,%6,%7),
IF ((NOT STARTED),
    PUT '/'/;
    STARTED = 1;
  );
PUT %3.TL,’.’,%4.TL, ’.’,%5.TL,’.’,%6.TL,’.’,%7.TL,%2(%3,%4,%5,%6,%7)/;
);
IF (STARTED,
  PUT ’/;’/;
ELSE
  PUT ’; %2(%3,%4,%5,%6,%7) = 0;’/;
);
$GOTO EXIT
$LABEL LESS5
$IF A%6==A $GOTO LESS4
*       A 4-DIMENSIONAL PARAMETER:
LOOP((%3,%4,%5,%6)$%2(%3,%4,%5,%6),
 IF ((NOT STARTED),
   PUT ’/’/;
   STARTED = 1;
  );
 PUT %3.TL,’.’,%4.TL, ’.’,%5.TL,’.’,%6.TL,%2(%3,%4,%5,%6)/;
);
IF (STARTED,
  PUT ’/;’/;
ELSE
  PUT ’; %2(%3,%4,%5,%6) = 0;’/;
);
$GOTO EXIT
$LABEL LESS4
$IF A%5==A $GOTO LESS3
*       A 3-DIMENSIONAL PARAMETER:
LOOP((%3,%4,%5)$%2(%3,%4,%5),
 IF ((NOT STARTED),
    PUT ’/’/;
    STARTED = 1;
  );
 PUT %3.TL,’.’,%4.TL, ’.’,%5.TL,%2(%3,%4,%5)/;
);
IF (STARTED,
  PUT ’/;’/;
ELSE
  PUT ’; %2(%3,%4,%5) = 0;’/;
);
$GOTO EXIT
$LABEL LESS3
*       A 2-DIMENSIONAL PARAMETER:
$IF A%4==A $GOTO LESS2
LOOP((%3,%4)$%2(%3,%4),
  IF ((NOT STARTED),
    PUT ’/’/;
    STARTED = 1;
  );
  PUT %3.TL,’.’,%4.TL, %2(%3,%4)/;
);
IF (STARTED,
  PUT ’/;’/;
ELSE
```

```
  PUT '; %2(%3,%4) = 0;'/;
);
$GOTO EXIT
$LABEL LESS2
*      A 1-DIMENSIONAL PARAMETER:
LOOP(%3$%2(%3),
 IF ((NOT STARTED),
    PUT '/'/;
    STARTED = 1;
  );
  PUT %3.TL, %2(%3)/;
);
IF (STARTED,
  PUT '/;'/;
ELSE
  PUT '; %2(%3) = 0;'/;
);
$LABEL EXIT
```

3.20 A TSP Gams Code

Answer from n.n:

```
I was asked to solve a small Traveling Scientist Problem and I
thought that I would simply "cook-up" a GAMS model. It was harder
than I thought, but I learned a bit about "Sets as Sequences". I
offer this to the list for comment - I still consider myself a
novice gams user.  If you have any suggestions, I (and I assume
the rest of the list) would like to hear them. My example is
definately *NOT* something you want to use for any TSPs that are
larger than 50 nodes (unless you like to torture solvers).  It is
simply something to put in the toolbox for quick and dirty use
(assuming that all is well with the code).
```

```
*---------------------------------------------------------------------
*
*   A GAMS version of the Traveling Scientist Problem
*   using H.P. Williams' formulation in "Modeling Building in
*   Mathematical Programming"
*
*   Philipp A. Djang
*    4 May 1995
*
*  A few Comments:
*
*  a. this is an example of how to use Sets as Sequences
*     see Chpt 12 in GAMS: A Users Guide.
*
*  b. This formulation generates a very large number of contraints
*     to prevent subtours.  Its O.K. for small TSPs (<50).
*     This is something for the toolchest, but for bigger problems,
*     you need something more efficient than a B&B solver.
*
*---------------------------------------------------------------------
```

```
* -- the option is required for proper set element ordering.
*    see p.134 & p.192-193 in GAMS:AUG
$ONUELLIST
Sets
c locations of interest
/
a1*a10
/
;
scalar
  NC number of cities - 1 / 9 /
  M diagonal value / 999 /
;
alias (c, cp);
table tdist(c, cp) square distance matrix
     a1   a2   a3   a4   a5   a6   a7   a8   a9  a10
a1  999    0    1    6    9   10    2    5    4    3
a2    0  999    7    1    6    6    8    6    5    9
a3    1    7  999   10    7   10    3    4    9    7
a4    6    1   10  999    9   10    2    9    9    3
a5    9    6    7    9  999    9    4    3    5    4
a6   10    6   10   10    9  999    8    7    5    6
a7    2    8    3    2    4    8  999    1    6    4
a8    5    6    4    9    3    7    1  999    6    8
a9    4    5    9    9    5    5    6    6  999    5
a10   3    9    7    3    4    6    4    8    5  999
;
variables
z   cost
binary variables
X(c,cp)  locations
positive variables
U(c) subtour detection
;
equations
cost      objective function
from(c)   for all cities
to(cp)     to  all cities
fseq(c,cp)   prevent forward subtours
bseq(c,cp)   prevent backward subtours
;
cost..  z =e= sum ( (c,cp)$(tdist(c,cp) ne M), tdist(c,cp)*X(c,cp));
from(c)..  sum( cp$(tdist(c,cp) ne M), X(c,cp)) =e= 1;
to(cp ).. sum( c$(tdist(c,cp) ne M), X(c,cp)) =e= 1;
*------------------------------------------------------------------
* the fseq and bseq equations require the set element ordering.
* ORD - used to denote the relative position of a set element
*       only for 1-dim, static, ordered sets
*
* Note the use of the lead operator in the equation defination
* e.g. fseq(c+1,cp+1)
*
*------------------------------------------------------------------
fseq(c+1,cp+1)$(ord(c) lt ord(cp))..
```

```
     u(c+1) - u(cp+1) + NC*X(c+1,cp+1) =l= NC-1;
bseq(c+1,cp+1)$(ord(c) gt ord(cp))..
     u(c+1) - u(cp+1) + NC*X(c+1,cp+1) =l= NC-1;

model tsp /all/;
option iterlim = 200000;
option reslim  = 200000;
option optcr = 0.1;
option optca = 0.1;
option sysout=off;
option limrow = 100;
option limcol = 100;
solve tsp minimizing z using mip;
display Z.L, X.L, U.L;
parameter Mdist(c,cp) route distance for each arc of the tour ;
Mdist(c,cp) = tdist(c,cp)*X.L(c,cp);
display Mdist;
```

3.21 A Jacobian construction Problem

Answer from n.n:

```
$TITLE  A suggestion for V. Shyamsundar's Jacobian construction  problem
$ontext Here is a description of V. Shyamsundar's problem, and my
suggestion for how to approach it.  The basic issue is how to use
GAMS to solve a nonlinear program via repeated linearization.  In
this application, formulating the linear submodel involves
calculation of a Jacobian matrix with one complicated nonlinear
equation. Mr. Shyamsundar proposed to access the Jacobian across
a subroutine call to the GAMS IO library.  I disagree with this
idea. I believe that it is MUCH easier to simply write the GAMS
code to evaluate the Jacobian and solve the linear programming
subproblem with a "generic" SOLVE. What I have included in this
message this is not an operational piece of code, but it
illustrates in detail how to do the Jacobian construction for
this model at a given point, set up and solve the linear
programming subproblem.
*-----------------------------------------------------------------------
Statement of the problem from various communications with
V.Shyamsundar goes as follows:
"I wish to use a recursive linear programming to solve a nonlinear
program. The algorithm involves linear programming subproblems of the
form:
            minimise  sum(j,p(j)+n(j)) +omega*(tau1 +tau2)
subject to
                   h(j) + del(h(j))*d =p(j)-n(j), (j=1 to(2*i+3))
                                       p(j),n(j) >= 0
                                       0< =x(i) <=1 (i= 1 to 5)
                                       0 <=y(i) <=1 (i=1 to  5)
                                       0<= tau1 <=1
                                       0 <= tau2 < =1
                                       0 <=L <=F
                                       0 <=V <=F
Here, the equations
```

```
                 L+V=F;
                 X(I)*L +Y(I)*V = Z(I)*F;   (I=1 TO 5)
                 SUM(I,X(I)) + TAU1 =0;
                 SUM(I,Y(I)) +TAU2 =0;
                 Y(I) = PHE2(I)*X(I)* PHE3(I)/(P*PHE1(I)); (i=1 to 5)
are denoted by the vector h(w) =0 and del(h(j)) is the Jacobian
matrix. Here w consists of the variables x(i),y(i),L V, tau1 and
tau2.  Also d is the search direction and omega is the scaling factor
for convergence.  p(j) and n(j) are artificial variables.  Hence I
need to calculate the Jacobian matrix del(h(j))at various levels of
variables in an iterative algorithm.
In these equations
(1) phe1(i) is a function of Y:
        phe1(i) = exp( ( (1/2)* bs(i) +bij(i,i)) * p/(rg*t) );
where:
        bs(i) = sum(jp,sum(s,(y(jp)/sumy)*(y(s)/sumy)*
                       (2.0*delik(jp,i)-delis(jp,s) ) ) ) ;
and
        sumy = sum(i,y(i));

(2) phe2(i) is a function of X:
        phe2(i) =  exp( log(prop(i,'ruq')/sumr) +luq(i)
                          + zcoord/2*prop(i,'quq')*
                          log( prop(j,'quq')*sumr
                          /(prop(j,'ruq')*sumq) )
                          -prop(j,'qpuq')*log(sumtptau(i))
                      -prop(j,'qpuq')*(sum(jp,prop(jp,'qpuq')
                       * (x(jp)/sumx)*exp(-cona(i,jp)/t)/
                       (sumqp*sumtptau(jp) ) ) ) ;
where
        sumx = sum(i,x(i));
        sumr = sum(i,prop(i,'ruq')*(x(i)/sumx) );
        sumq = sum(j,prop(j,'quq')*(x(j)/sumx));
        sumqp = sum(j,prop(j,'qpuq')*(x(j)/sumx));
        suml  = sum(j,luq(j)*(x(j)/sumx));
        sumtptau(j) = sum(jp,prop(jp,'qpuq')/sumqp
                        * (x(jp)/sumx)*exp(-cona(jp,j)/t)) ;

(3) phe3(j) does not depend on x(j) and y(j). It is a parameter:
        phe3(i) = exp( (p/(rg*t))* bij(i,i)); "
$offtext
*-----------------------------------------------------------------------
*       Begin with definitions of sets and other model data:
set         I         "component code ---name     :      formula"
                       /MTHNOL    methanol       CH3OH
                        ACETON    acetone        CH3COCH3
                        MTHACT    methyl acetate CH3COOCH3
                        BENZEN    benzene        C6H6
                        CLRFRM    chloroform     CHCL3/;
alias (I,J);
parameter z(j)   mole fractions in feed stream
                     /MTHNOL     0.15
                      ACETON     0.40
                      MTHACT     0.05
                      BENZEN     0.20
                      CLRFRM     0.20/  ;
scalar  omega     scaling factor for equilibrium eqn  /1.0E-03/
.... Other data tables need to be inserted here.
*-----------------------------------------------------------------------
*       Read current values for these parameters.  These could come
*       from an include file during debugging.  Later we can move the
*       LP code from here into a complete GAMS program which executes
*       the outer algorithm.
scalar
        L       liquid flow rate
        V       vapor flow rate
        tau1    tau value for liquid
        tau2    tau value for vapor      ;
parameter
        y(j)      mole fraction of jth component in vapor
        x(j)      mole fraction of jth component in liquid ;
$INCLUDE levels.gms
*-----------------------------------------------------------------------
*       Evaluate h(w) at the current point.
*       Declare intermediate variables we will need for constructing
*       the LP subproblem:
parameter
        sumx            Sum of current X’s
        sumy            Sum of current Y’s
        sumr            Function of current X’s
        sumq            Function of current X’s
        sumqp           Function of current X’s
        suml            Function of current X’s
        sumtptau(j)     Function of current X’s
        bs(j)           Sum of bij*y(j)
        phe1(j)         Vapor fugacity coefficient
        phe2(j)         Liquid activity coefficient
        phe3(j)         Pure component fugacity coefficient
        H_F             Current level value for fdef
        H_TAU1          Current level value for tau1def
        H_TAU2          Current level value for tau2def
        H_Z(I)          Current level value for zdef(i)
        H_Y(I)          Current level value for ydef(i)
h_f = L + V - F;
h_tau1 = SUM(I, Y(I)) - TAU2;
h_tau2 = SUM(I, X(I)) - TAU1;
h_z(I) = X(I) * L + Y(I) * V - Z(I) * F;
sumx = sum(j,x(j));
sumy = sum(j,y(j));
sumr = sum(i,prop(i,’ruq’)*(x(i)/sumx) );
sumq = sum(j,prop(j,’quq’)*(x(j)/sumx));
sumqp = sum(j,prop(j,’qpuq’)*(x(j)/sumx));
suml  = sum(j,luq(j)*(x(j)/sumx));
sumtptau(j) = sum(jp,prop(jp,’qpuq’)/sumqp
                                   * (x(jp)/sumx)*exp(-cona(jp,j)/t)) ;
bs(j) =e= (1/(sumy*sumy))
        * sum(jp,y(jp)*sum(s,y(s)*(2.0*delik(jp,j)-delis(jp,s))));
phe1(j) =e= exp(((1/2)*bs(j)+bij(j,j))*p/(rg*t));
```

```
phe2(j) = exp(log(prop(j,'ruq')/sumr)+luq(j)
                            +zcoord/2*prop(j,'quq')
                            *log(prop(j,'quq')*sumr
                            /(prop(j,'ruq')*sumq))
                            -(prop(j,'ruq')/sumr)*suml
                          -prop(j,'qpuq')*log(sumtptau(j))
                -prop(j,'qpuq')*(sum(jp,prop(jp,'qpuq')*(x(jp)/sumx)*
                            exp(-cona(j,jp)/t)/(sumqp*sumtptau(jp))))
                            +prop(j,'qpuq')) ;
phe3(j) =e= exp((p/(rg*t))*bij(j,j));
h_y(I) = Y(I) - PHE2(I) * X(I) * PHE3(I) / (P * PHE1(I));
*----------------------------------------------------------------------
*       Evaluate the Jacobian del( h(w) ) at the current point:
        D_BS_DY(J,I)    Gradient of BS(J) wrt Y(I)
        D_PHE1_DY(J,I)  Gradient of PHE1(J) wrt Y(I)
        D_SUMR_DX(I)    Gradient of SUMR wrt X(I)
        D_SUMQ_DX(I)    Gradient of SUMQ wrt X(I)
        D_SUMQP_DX(I)   Gradient of SUMQP wrt X(I)
        D_SUML_DX(I)    Gradient of SUML wrt X(I)
        D_SUMTPTAU(I,J) Gradient of SUMTPTAU(I) wrt X(J)
        D_PHE2_DX(J,I)  Gradient of PHE2(J) wrt X(I)
*       Defining  FXY(I) = PHE2(I)*X(I)*PHE3(I)/(P*PHE1(I)):
        D_DY(I,J)       Gradient of FXY(I) wrt Y(J)
        D_DX(I,J)       Gradient of FXY(I) wrt X(J);

*       Need to do some calculus here differentiating h_y(J)
*       respect to x(I).  It helps to do the derivatives of the
*       intermediate terms first.
d_bs_dy(j,i) = ...
d_phe1_dy(j,i) = ...
d_sumr_dx(i)    = ..
d_sumq_dx(i)    = ..
d_sumqp_dx(i)   = ..
d_suml_dx(i)    = ..
d_sumtptau(i,j) = ..
d_phe2_dx(j,i)  = ..
d_dy(i,j) = (PHE2(I)*X(I)*PHE3(I)) / (P*PHE1(I)*PHE1(I))
*               D_PHE1_DY(I,J);
d_dx(i,j) = (X(I) * PHE3(I) / (P * PHE1(I))) * D_PHE2_DX(I,J);
*---------------------------------------------------------------------
*       Declare and solve the linear programming subproblem:
positive variables
                P_F,N_F         Artificial variables
                P_TAU1,N_TAU1
                P_TAU2,N_TAU2
                P_Z(I),N_Z(I)
                P_Y(I),N_Y(I)
variables
                LPOBJ           LP objective
                D_L             Direction for L
                D_V             Direction for V
                D_X(I)          Direction for X(I)
                D_Y(I)          Direction for Y(I);
```

```
*       Declare the objective function and linearized constraints
*       based on the Jacobian of h.
objdef..        lpobj =e= P_F + P_TAU1 + P_TAU2 + SUM(I, P_Z(I) +
P_Y(I))
                        - N_F + N_TAU1 + N_TAU2 + SUM(I, N_Z(I) +
N_Y(I))
                        + omega * (D_TAU1 + D_TAU2);
fdef..          h_f    + D_L + D_V               =E= P_F - N_F;
tau1def..       h_tau1 + sum(i, D_X(I)) + D_TAU1 =E= P_TAU1 - N_TAU1;
tau2def..       h_tau2 + sum(i, D_Y(I)) + D_TAU2 =E= P_TAU2 - N_TAU2;
zdef(i)..       h_z(I) + L * D_X(I) + X(I) * D_L +
                         V * D_Y(I) + Y(I) * D_V =E= P_Z(I) - N_Z(I);
ydef(i)..       h_y(I) + D_Y(I) + SUM(J, d_dy(I,J) * D_Y(J))
                                + SUM(J, d_dx(I,J) * D_X(J)) =E=
P_Y(I)-N_Y(I);
model subproblem / LPOBJ, FDEF, TAU1DEF, TAU2DEF, ZDEF, YDEF/;
*       Install bounds on the direction vector to assure that the
*       bounds on the resulting central variables (X, Y etc.) are
*       satisified:
dx.up = 1-x;
dx.lo = -x;
dy.up = 1-y;
dy.lo = -y;
d_tau1.up = 1-tau1;
d_tau1.lo = -tau1;
d_tau2.up = 1-tau2;
d_tau2.lo = -tau2;
d_L.up = F - L;
d_L.lo = -L;
d_V.up = F - V;
d_V.lo = -V;
solve subproblem using lp minimizing lpobj;
```

3.22 Formulating a piece wise linear function

I am programming a nlp problem in agricultural economics using a quadratic yield function which contains the amount of nitrogen application as independent variable (y=f(n)). After I managed to model this yield function I am facing serious trouble in formulating a series of equations that again depend on the variables y and n which are both calculated endogenously (by maximizing profit). For example, I need to set up an equation on the demand of a fertilizer spreader which depends on the amount of n required to reach the optimal yield level. Since the optimal yield level is not pretermined but calculated endogenously the demand for the fertilizer spreader is not a single value but a function. However, this function is discontinuous reaching from 0 applications to 4, i.e. demand for the spreader needs to be represented by an integer variable. The ranges are as follows:

kg nitrogen	number of applications per year
0	0

0<N<=70	1
70<N<=120	2
120<N<=160	3
N>160	4

I tried to set up the following equation:

```
A(c)              =E= 4 $ ((N(c) GT 160))
                     +3 $ ((N(c) GT 120) AND (N(c) LE 160)
                     +2 $ ((N(c) GT 70)  AND (N(c) LE 120)
                     +1 $ ((N(c) GT 0)   AND (N(c) LE  70);
```

A means the number of applications per year, and c are the crops. N is the amount of nitrogen applied. This equation results in an error message saying that endogenous operands are not allowed. I need to specify endogenous conditions (i.e. conditions that that contain endogenous variables). I read in the manual that this is not possible in combination with nlp. How can I avoid this and does GAMS have the possiblity to express condition like the one above (which to me seems logical but does not work for some reason).

Answer from n.n:

Discrete decisions such as the "number of applications per year" need to be modeled with GAMS "Integer" or "Binary" variables. Since your model is also nonlinear, you will then have a mixed integer nonlinear model, GAMS problem type MINLP. To solve such a model you would need to use the GAMS/DICOPT solver. DICOPT requires that you have a MIP solver available in addition to an NLP-solver. If you would like to try an MINLP approach but don't have the required solvers, you can contact me "off list" to arrange for an evaluation. Here's a rough sketch of one way to use binary variables to do what you ask for.

```
        Set nApps /1*4/;
        Binary Variable app(nApps);
        Equation HowMany;
        HowMany.. Sum(nApps, app(nApps)) =L= 1;
        Set bounds /low, high/;
        Table nBounds(nApps, bounds)
                        low     high
                1       0       70
                2       70      120
                3       120     160
                4       160     Inf
                ;
        Equations nLow, nHigh;
        nLow..  n =G= Sum(nApps, nBounds(nApps, "low")  * app(nApps));
        nHigh.. n =L= Sum(nApps, nBounds(nApps, "high") * app(nApps));
```

Now at most one of the binary variables will equal one at the solution. Limits on n will be adjusted accordingly.

3.23 A PSD-Problem

Recently we got a problem to solve, that we are not able to manage. We hope, that you can give us some hints, how we can apply GAMS to our problem. In general we have an objective function, that has to be minimised with some non-linear constraints, that must be calculated iteratively. We give you a short example, that shows the principal structure of the problem: Given a matrix A0 with elements A0(i,j) defined as parameter and a matrix A with elements A(i,j) defined as variables. The objective function is defined by

```
        DEV .. delta =E= SUM((i,j), SQR(A(i,j) - A0(i,j)));
```

The constraints are: All eigenvalues of A must be positive

```
Eigenvalues(i) .. EV(i) > 0
```

We want to use some iteratively working routine like Jacobi transformation to calculate the eigenvalues of the (symmetric) matrix A. We think that we have to define something like a subroutine, that can be called by the solver each time it is necessary for the optimisation process. This call must be performed each time the solver checks the constraint EV(i) > 0. Another problem arises from the fact, that the solver checks the constraints in sequence. So it can make no use of the fact, that after checking the first eigenvalue the other eigenvalues are yet known and have only to be referenced for the remaining n-1 checks. Our question to you is: Is it possible to provide user defined subroutines (FORTRAN or C -Syntax) like FUNOBJ and FUNCON by using the GAMS interface or must we use MINOS directly ?"

Answer from rutherford@colorado.edu:

Let's begin with a fundamental rule: if you can program in GAMS, don't write Fortran or C code. In this case, I would resort to writing my own procedure only if GAMS were completely incapable of solving the problem. An example shown below shows one approach, based on Result 6C from Strang: "A is positive definite if and only if it satisfies the condition: There exists a nonsingular matrix W such that A = W'W." I list a sample program here. In this program (straight GAMS, nothing fancy here) I solve for W and its inverse (V) to minimize the norm distance of W'W from A0. Applying a system of constraints to define V in terms of W: W V = I, and installing upper and lower bounds on the values of V assure that W is nonsingular. CONOPT solves this from the stated starting point, W = V = I, but it fails if you set both equal to zero. There is a slight difference in the results from MINOS and CONOPT for the first example. This is because the target matrix is completely random, so the optimizer drives the calibration toward a singular W. I have (rather arbitrarily) set upper and lower bounds on elements of V equal to +/-100.

```
----    130 PARAMETER A0            Target matrix
            1           2           3           4
1      -0.970       0.726       0.188       0.403
2      -0.006      -0.944       0.953       0.877
3      -0.199      -0.126       0.432       0.779
4       0.933      -0.777       0.381       0.689

Answer from MINOS(non-PSD target matrix):

----    187 VARIABLE  OBJ.L = 5.368 objective function
----    187 PARAMETER A             PSD estimate of the matrix a0
            1           2           3           4
1       0.091       0.045       0.157       0.274
2       0.045       0.038       0.140       0.149
3       0.157       0.140       0.514       0.524
4       0.274       0.149       0.524       0.836

Answer from CONOPT(non-PSD target matrix):

----    173 VARIABLE  OBJ.L = 5.389 objective function
----    173 PARAMETER A             PSD estimate of the matrix a0
            1           2           3           4
1       0.081       0.052       0.186       0.255
2       0.052       0.034       0.119       0.163
3       0.186       0.119       0.427       0.582
4       0.255       0.163       0.582       0.796
----    133 PARAMETER A             PSD estimate of the matrix a0
            1           2           3           4
1       0.081       0.052       0.186       0.255
2       0.052       0.034       0.119       0.163
3       0.186       0.119       0.427       0.582
4       0.255       0.163       0.582       0.796

Answer from CONOPT (PSD target matrix):

----    197 VARIABLE  OBJ.L =  1.507813E-6 objective function
----    197 PARAMETER A0            Target matrix
            1           2           3           4
1       1.851      -1.398       0.081       0.092
2      -1.398       2.039      -1.114      -1.169
3       0.081      -1.114       1.274       1.510
4       0.092      -1.169       1.510       2.013

----    197 PARAMETER A             PSD estimate of the matrix a0
            1           2           3           4
1       1.850      -1.398       0.081       0.092
2      -1.398       2.039      -1.114      -1.169
3       0.081      -1.114       1.274       1.510
4       0.092      -1.169       1.510       2.013
```

```
scalar  psdtarget       Switch for a PSD target matrix /0/;
set     i  row and column indices /1*4/;
alias (i,j,k);
```

```
parameter       a0(i,j)         Target matrix
                w0(i,j)         Generator for the target matrix
                ident(i,j)      The identity matrix;
option seed=101;
if (psdtarget,
        w0(i,j) = uniform(-1,1);
        a0(i,j) = sum(k, w0(k,i)*w0(k,j));
else
        a0(i,j) = uniform(-1,1);
);
ident(i,j) = 1$(ord(i) eq ord(j));
variables       obj     objective function (square norm)
                w(i,j)  symmetric matrix which determines a
                v(i,j)  the inverse of w;
equations       objdef  defines the deviation
                vdef    defines the matrix v;
objdef..        obj =e= sum((i,j), sqr( a0(i,j) - sum(k,
w(k,i)*w(k,j))));
vdef(i,j)..     sum(k, w(i,k)*v(k,j)) =e= ident(i,j)

model approx /all/;
*       Place bounds on V to assure that W is nonsingular:
v.up(i,j) = 100;
v.lo(i,j) = -100;
*       Initial values help!
v.l(i,j) = ident(i,j);
w.l(i,j) = ident(i,j);
*       Solve the problem:
approx.iterlim = 8000;
solve approx using nlp minimizing obj;
*       Report some results:
parameter    a(i,j)  PSD estimate of the matrix a0;
a(i,j) = sum(k, w.l(k,i)*w.l(k,j));
display obj.l, a0, a;
```

Answer from adrud@arki.dk:

Tom Rutherford had a nice note on symmetric positive definite (spd) matrices. You can sometime do it a little simpler. If A is spd the it can be written as L*L' where L is a lower triangular matrix (Cholesky factorization) and if A is strictly positive definite then you can select the diagonal elements of L to be strictly positive. Since A is symmetric you only need to work on half of it (both when you count variables and equations) and you can get some variables and constraints like these:

```
set i;
alias (i,j,k)
variables a(i,j), l(i,j);
equations adef(i,j) ;
adef(i,j)$(ord(i) ge ord(j)) ..
   a(i,j) =e= sum(k$(ord(k) le ord(j)), l(i,k)*l(j,k) );
* ensure A will be strictly positive definite:
l.lo(i,i) = 1.e-5;
```

```
* Initial values are important, and a consistens set is even better:
l(i,i) = 1; a(i,i) = 1;

Note that the model is very nonconvex and there is no guarantie
for a global optimum. The model is also rather dense for large I
which means that the model can be expensive to solve. And
finally, this one of the classes of models for which CONOPT seems
to be better than MINOS! The construct shown here is also useful
for other purposes. It is fairly easy to compute the inverse of L
(also in GAMS) and from this you can compute the inverse of A,
which is needed for some statistical estimation models.
```

Answer from **rutherford@colorado.edu:**

```
$title  Least squares approximation of a given matrix with a psd matrix
$ontext
This file contains an improved version of yesterday's posting on
least squares estimation of a positive definite matrix.  I have
incorporated suggestions from Arne Drud and Alex Meeraus.  The
original question, my initial reply, and the following
suggestions from Arne and Alex are listed.  The revised program
appears below. The key difference with the earlier program is
that I have incorporated Arne's suggestion to use an Cholesky
factorization of the PSD matrix so that invertibility can be
easily imposed as a lower bound on the diagonal elements of W.
In addition this approach assures that the resulting matrix is
symmetric. This significantly improves performance (the sample
program solves a 20x20 problem in 16 seconds on my PP200!). - Tom
------------------------------------------------------------------------
Query:
"...recently we got a problem to solve, that we are not able to
manage.  We hope, that you can give us some hints, how we can apply
GAMS to our problem.
In general we have an objective function, that has to be minimised
with some non-linear constraints, that must be calculated
iteratively. We give you a short example, that shows the principal
structure of the problem:
Given a matrix A0 with elements A0(i,j) defined as parameter and a
matrix A with elements A(i,j) defined as variables.  The objective
function is defined by
        DEV .. delta =E= SUM((i,j), SQR(A(i,j) - A0(i,j)));
The constraints are: All eigenvalues of A must be positive
Eigenvalues(i) .. EV(i) > 0
We want to use some iteratively working routine like Jacobi
transformation to calculate the eigenvalues of the (symmetric) matrix
A. We think that we have to define something like a subroutine, that
can be called by the solver each time it is necessary for the
optimisation process. This call must be performed each time the
solver checks the constraint EV(i) > 0.  Another problem arises from
the fact, that the solver checks the constraints in sequence. So it
can make no use of the fact, that after checking the first eigenvalue
the other eigenvalues are yet known and have only to be referenced
for the remaining n-1 checks.
Our question to you is: Is it possible to provide user defined
subroutines (FORTRAN or C -Syntax) like FUNOBJ and FUNCON by using
```

```
the GAMS interface or must we use MINOS directly ?"
------------------------------------------------------------------------
My initial reply:
Let's begin with a fundamental rule: if you can program in GAMS,
don't write Fortran or C code.  In this case, I would resort to
writing my own procedure only if GAMS were completely incapable of
solving the problem.  An example shown below shows one approach,
based on Result 6C from Strang: "A is positive definite if and only
if it satisfies the condition: There exists a nonsingular matrix W
such that A = W'W."
I list a sample program here.  In this program (straight GAMS,
nothing fancy here) I solve for W and its inverse (V) to minimize the
norm distance of W'W from A0.  Applying a system of constraints to
define V in terms of W: W V = I, and installing upper and lower
bounds on the values of V assure that W is nonsingular.  CONOPT
solves this from the stated starting point, W = V = I, but it fails
if you set both equal to zero.
There is a slight difference in the results from MINOS and CONOPT for
the first example.  This is because the target matrix is completely
random, so the optimizer drives the calibration toward a singular W.
I have (rather arbitrarily) set upper and lower bounds on elements of
V equal to +/-100.
------------------------------------------------------------------------
Alex Meeraus replied:
1. New predefined set and parameter
      sameas(i,j)    this is a set and compares labels
      diag(i,j)      this is a parameter with 0/1 entries
   all fully optimized, is equivalent to
       alias(*,u); parameter diag(u,u) = 1;    etc
2. You claim that the matrix W has to be symetric. Just adding a
   constraint like
       wsym(i,j)$(ord(i) > ord(j)).. w(i,j) =e= w(j,i);
    makes the solution much faster.
3. symetric w -> symetric V -> symetrix A ?????
   I played around with only putting a triangle in but this
   created more headaches because of linear dependecies that
   are introduced...
How important are symetric matrices in the kind of modeling you do?
------------------------------------------------------------------------
Arne Drud replied:
You can sometime do it a little simpler. If A is spd the it can be
written as L*L' where L is a lower triangular matrix (Cholesky
factorization) and if A is strictly positive definite then you can
select the diagonal elements of L to be strictly positive. Since A is
symmetric you only need to work on half of it (both when you count
variables and equations) and you can get some variables and constraints
like these:
set i;
alias (i,j,k)
variables a(i,j), l(i,j);
equations adef(i,j) ;
adef(i,j)$(ord(i) ge ord(j)) ..
   a(i,j) =e= sum(k$(ord(k) le ord(j)), l(i,k)*l(j,k) );
* ensure A will be strictly positive definite:
```

```
l.lo(i,i) = 1.e-5;
* Initial values are important, and a consistens set is even better:
l(i,i) = 1; a(i,i) = 1;
Note that the model is very nonconvex and there is no guarantie for a
global optimum. The model is also rather dense for large I which means
that the model can be expensive to solve. And finally, this one of the
classes of models for which CONOPT seems to be better than MINOS!
The construct shown here is also useful for other purposes. It is fairly
easy to compute the inverse of L (also in GAMS) and from this you can
compute the inverse of A, which is needed for some statistical
estimation models.
PS: The material above was extracted from an estimation model that
required some intermediate results to be positive definite. I received
the model from Agapi at the US Department of Agriculture several years
ago.
-----------------------------------------------------------------------
$offtext
*       Revised program:
scalar  psdtarget       Switch for a PSD target matrix /1/;
set     i  row and column indices /1*20/;
alias (i,j,k);
parameter       a0(i,j)         Target matrix
                w0(i,j)         Generator for the target matrix;
option seed=101;
if (psdtarget,
        w0(i,j) = uniform(-1,1);
        a0(i,j) = sum(k, w0(k,i)*w0(k,j));
else
        a0(i,j) = uniform(-1,1);
);
variables       obj     objective function (square norm)
                w(i,j)  upper triangular matrix which determines a;
equations       objdef  defines the deviation;
set     L(i,j)  identifies the lower triangular elements of W;
L(i,j) = yes$(ord(i) le ord(j));
objdef..        obj =e=   sum((i,j), sqr(a0(i,j) -
                          sum(k$(L(k,i)*L(k,j)), w(k,i)*w(k,j))));
model approx /all/;
*       Place lower bounds on diagonals to assure that W is
*       invertible:
w.lo(i,i) = 0.01;
*       As per Arne's suggestion, install uniform initial values:
w.L(i,j) = 1$L(i,j);
*       Solve the problem:
solve approx using nlp minimizing obj;
*       Report some results:
parameter       a(i,j)  Estimated PSD matrix;
a(i,j) = sum(k$(L(k,i)*L(k,j)), w.L(k,i)*w.L(k,j));
display obj.l, a0, a, w.l;
```

Answer from adrud@arki.dk:

Tom Rutherford and I have posted a few notes on Symmetric
Positive Definite (SPD) Matrices, and Michael Ferris objected in

a private note saying:

1) Our suggestions are only good for matrices that are nicely positive definite. It is both slow and unrealiable for matrices with small diagonal elements (and an example showed how poorly it worked - se below).
2) Methods from numerical analysis, e.g. incomplete Cholesky factorization, works much better and faster.

I agree with Michael on both points. However, as I understood the original question about a week ago, the SPD approximation was just one small part of a larger model. I have worked with a model where the objective was to estimate a production function given some observations, with the constraint that a derivative matrix for the production function had to be positive definite. The model components shown by Tom and I can be used as building blocks in larger models to represent constraints like these. The example given by Tom has one little misprint relative to my original suggestion, namely in the initialization. You should use a diagonal matrix, W(i,i) = 1, and not W(i,j)$L(i,j) = 1. The diagonal puts equal emphasis on the variables. Results with Tom's start and Michaels data (shown below) on my Pentium/133 are:

	Iterations	Seconds	Objective
MINOS	3579	237	1.5729
CONOPT	902	61.8	1.5729

and with the diagonal start:

	Iterations	Seconds	Objective
MINOS	1163	75.9	0.0065
CONOPT	191	9.9	0.0000

If you remove the lower bound on the diagonal of the factor, thereby changing the requirement from strictly positive definite to positive semi definite (we do not have a good measure for how close we can accept it anyway and the model becomes unconstrained without bounds -- the easiest class we have) then the results become:

	Iterations	Seconds	Objective
MINOS	133	8.9	0.0000
CONOPT	212	10.8	0.0000

As a comparions to these results I have also written a small GAMS program to do an incomplete Cholesky factorization. In model 1 I simply ignored small pivots, and in model 2 I perform pivoting. The GAMS executing times were around 0.150 seconds. The models with Michael Ferris "bad" data follows.

```
*------------------Start of model 1--------------------
$Title Incomplete Cholesky factorization of a symmetric matrix
```

```
set     i  row and column indices /1*20/;
alias (i,j,k);
$offdigit
parameter a0(i,j) /
       1   .    1   5.7516984919844127e-01
       4   .    1   2.3583722392844148e-02
      11   .    1   8.6180214149803910e-02
      13   .    1  -7.8774210794328989e-02
       2   .    2   2.5933699113962794e+00
      16   .    2  -5.6469681886321366e-01
       3   .    3   1.0094267925511156e-02
      20   .    3  -5.2013816131375535e-03
       1   .    4   2.3583722392844148e-02
       4   .    4   2.0122300331621105e-02
      12   .    4   5.9836243816531627e-04
       5   .    5   9.2897962109258558e-01
       8   .    5   5.4654560970479460e-01
       6   .    6   8.0318799781596225e-04
       7   .    7   8.0318799781596225e-04
       5   .    8   5.4654560970479460e-01
       8   .    8   3.5911350730110653e-01
       9   .    9   1.8967890006517129e-03
      11   .    9   7.9469458457900512e-03
      10   .   10   5.3208344220487769e-01
      15   .   10   3.6665175309008494e-02
      18   .   10  -3.3533052550801780e-02
       1   .   11   8.6180214149803910e-02
       9   .   11   7.9469458457900512e-03
      11   .   11   3.2876798770161975e-01
       4   .   12   5.9836243816531627e-04
      12   .   12   3.3958658705181964e-02
      17   .   12  -1.0873218783097420e-01
       1   .   13  -7.8774210794328989e-02
      13   .   13   1.2805496980093140e-02
      14   .   14   1.5646490670026322e-02
      19   .   14   9.2180704053456998e-03
      10   .   15   3.6665175309008494e-02
      15   .   15   4.3655877525130863e-03
       2   .   16  -5.6469681886321366e-01
      16   .   16   1.2380194994143388e-01
      12   .   17  -1.0873218783097420e-01
      17   .   17   3.6300494372786224e-01
      10   .   18  -3.3533052550801780e-02
      18   .   18   1.1824288891662375e-02
      20   .   18   2.9993058788637686e-02
      14   .   19   9.2180704053456998e-03
      19   .   19   6.5278453651413893e-03
       3   .   20  -5.2013816131375535e-03
      18   .   20   2.9993058788637686e-02
      20   .   20   2.4584964256367958e-01
/;
a0(i,j) = 100*a0(i,j);
set l(i,j) defines the lower triangle;
parameters  w(i,j)    factor of a
parameters  ww(i,j)   part of a not yet factored
sets   r(i)   the pivot row
       npr(i) rows not yet used for pivot
       npc(i) columns not yet used for pivot
scalar big    the diagonal pivot at each point in the inversion
       piv    the square root of the pivot
       tol    pivot tolerance;
r(i)    = yes;
npr(i)  = yes;
npc(i)  = yes;
ww(i,j) = a0(i,j);
tol     = 1e-5;
loop(r, big = ww(r,r);
   if ( big ge tol,
      piv = sqrt(big);
      l(npr,r) = yes;
      npr(r)   = no;
      npc(r)   = no;
      w(r  ,r) = piv;
      w(npr,r) = ww(npr,r) / piv;
      ww(npr,npc) = ww(npr,npc) - ww(npr,r)*ww(npc,r)/big;
      ww(npr,r) = 0;
      ww(r,npc) = 0;
      ww(r,r)   = 0;
      );
   );
parameter a(i,j) the estimated psd matrix,
          adif(i,j) the difference matrix
          obj objective function;
a(i,j)    = sum(k$(l(i,k)*l(j,k)), w(i,k)*w(j,k) );
adif(i,j) = round(a(i,j) - a0(i,j),8);
obj       = sum((i,j), sqr(a0(i,j)-a(i,j)));
display l, w, ww, a, a0, adif, obj;
*------------------end of model 1--------------------
*------------------start of model 2 (without data)----
* The last part of the model can be replaced by the following
* where the pivots are selected in decreasing order. The ideas
* are taken from the GAUSS model in GAMSLIB.
set l(i,j) defines the permuted lower triangle;
parameters  w(i,j)    factor of a
parameters  ww(i,j)   part of a not yet factored
sets   r(i)   the selected pivot row
       npr(i) rows not yet used for pivot
       npc(i) columns not yet used for pivot
scalar big    the best diagonal pivot at each point in the inversion
       piv    the square root of the pivot
       tol    pivot tolerance;
r(i)    = yes;
npr(i)  = yes;
npc(i)  = yes;
ww(i,j) = a0(i,j);
tol     = 1e-5;
loop(j, big = smax(npr, ww(npr,npr));
   big$(big lt tol) = 0;
   loop(r$(big and big eq ww(r,r)),
      piv = sqrt(big);
      l(npr,r) = yes;
      npr(r)   = no;
      npc(r)   = no;
      w(r  ,r) = piv;
      w(npr,r) = ww(npr,r) / piv;
      ww(npr,npc) = ww(npr,npc) - ww(npr,r)*ww(npc,r)/big;
      ww(npr,r) = 0;
      ww(r,npc) = 0;
      ww(r,r)   = 0;
      big = 0;
      );
   );
parameter a(i,j) the estimated psd matrix,
          adif(i,j) the difference matrix
          obj objective function;
a(i,j)    = sum(k$(l(i,k)*l(j,k)), w(i,k)*w(j,k) );
adif(i,j) = round(a(i,j) - a0(i,j),8);
obj       = sum((i,j), sqr(a0(i,j)-a(i,j)));
display l, w, ww, a, a0, adif, obj;
*------------------end of model 2 ------------
```

3.24 Time optimal dynamic optimization by GAMS

Can GAMS be used to solve time optimal dynamic optimization problems and how? Is there any numerical time minimization optimum growth modelling study?

Answer from adrud@arki.dk:

You cannot make the number of elements in a time set variable and minimize it. On the other hand, you can create a model with a fixed number of time steps and make the length of each time step variable and then minimize this variable. The model becomes more complex and more nonlinear since the length of the time step probably appears many places in the model and most places nonlinearly.

3.25 Modelling convergence

I've got a model which optimises farm hold decisions for different households, subject to expected prices. The aggregate supply of these households is linked to an aggreagte demand function which gives me a new equilibrium price. This process can be itterated to derive a new stable equilibrium price.

price ==> expected price ==> production ==> supply ==> new price

Does anyone have experience with this type of modelling. What I am interested in is what is the most efficient way to reach the stable equilibrium, where:

```
(new price - price)/price < threshold
```

Answer from sdirkse@gams.com:

From the description above, it looks like the dual variables (prices) must satisfy some additional constraints determined by the supply and demand equations, in addition to being a proper set of duals for the optimization problem at hand. For problems like this, I have found it helpful to use a complementarity framework.

Briefly, the KKT conditions of the optimization problem are first written down as an MCP model, and then additional constraints or conditions are imposed by augmenting the MCP. There are tools to do the first step automatically; we hope to release those soon. The second step depends on the model at hand, and is not so simple to automate.

As examples, I attach two models, one dealing with a joint maximization of utility, the other with an energy system (PIES). In each example, an NLP views as fixed a value that really depends on the optimal dual variables for the NLP, and an MCP is used to compute everything in one shot.

```
$TITLE Negishi weight example $offsymxref offsymlist offuellist
offuelxref

$inlinecom /* */
*
* In this example, we have a number of regions,
* each of which produces a unique good
* Each region consumes goods in order to
* maximize utility, where utility is
* given by a Cobb-Douglas function of the form
*
*       U = prod(goods, consum(goods)**alpha(goods))
*
*
* We would like to find Negishi weights for the objective
* such that the budget constraint for each region is satisfied
* (i.e. we have a balance of trade)

set regions     / 1 * 3 /;
alias (regions, goods, regions2, goods2);

* alpha determines utility functions
table alpha(goods,regions)
        1       2       3
1       .7      .4      .2
2       .2      .3      .4
3       .1      .3      .4 ;

parameter production(goods) /
```

```
1       10
2       8
3       3
/ ;

* scale alpha, just in case
alpha(goods,regions) = alpha(goods,regions)/sum(goods2,alpha(goods2,regions));

* optimal weights:
* 1     1
* 2     1.37723778
* 3     1.32621111

display alpha;

scalar jointutil;
parameter tbalance(regions);
parameter negi_parm(regions) /
1       1
2       1
3       1
/ ;

variables
        utility,
        consum(goods,regions);

consum.lo(goods,regions) = .1;
consum.l(goods,regions) = 5;

equations
        utilDef,
        prodLimit(goods);

utilDef ..
        utility =e=
        sum(regions, negi_parm(regions) *
            prod(goods, consum(goods,regions)**alpha(goods,regions) )
        );

prodLimit(goods) ..
        production(goods) =g= sum(regions, consum(goods,regions));

model nnlp      / utilDef, prodLimit /;

option nlp=conopt;
option limrow = 0;
option limcol = 0;

solve nnlp using nlp maximizing utility;

tbalance(regions) = production(regions)*(-prodLimit.m(regions))
        + sum(goods, prodLimit.m(goods) * consum.l(goods,regions));
jointutil = utility.l;
```

```
file dfile / results.out /;
put dfile /;
put "NLP solution, fixed weights:"/;
put "joint utility:" jointutil:10:5//;
put "good     price       region  consumption" /;
put "----    -------      ------  -----------" /;
loop (goods,
  put "  ", goods.te(goods):3:0, (-prodLimit.m(goods)):10:5 /;
  loop (regions,
    put "                       ",
        regions.te(regions):5:0, consum.l(goods,regions):12:5 /;
  )
);
put/"region       trade balance     negishi weight" /;
put "------       -------------     --------------" /;
loop (regions,
  put "  ", regions.te(regions):5:0, tbalance(regions):18:8,
        negi_parm(regions):18:8/;
);

* now we make the dual variables (prices) in the NLP
* explicit by writing out the KKT (first order) conditions,
* but we keep the negishi weights fixed.

positive variable
    pi(goods);
variable
    negi(regions);

equations
    dutil(goods,regions),
    budget(regions);

dutil(goods,regions) ..          /* d(utildef)/dconsum */
        - prod(goods2$(ord(goods2) ne ord(goods)),
          consum(goods2,regions)**alpha(goods2,regions) )
        * consum(goods,regions)**(alpha(goods,regions)-1)
           * alpha(goods,regions)
        * negi(regions)
                                /* dual variables to prodLimit constraint */
        + pi(goods)
        =g= 0;

budget(regions) ..
        production(regions) * pi(regions)
        =e= sum(goods, consum(goods,regions)*pi(goods) );

model negKKT  / dutil.consum, prodLimit.pi /;

* fixing the weights gives the same answer as the NLP above
negi.fx(regions) = negi_parm(regions);
solve negKKT using MCP;
```

```
put //;
put "MCP (KKT) solution, fixed weights:"/;
put "joint utility:" jointutil:10:5//;
put "good     price       region  consumption" /;
put "----    -------      ------  -----------" /;
loop (goods,
  put "  ", goods.te(goods):3:0, pi.l(goods):10:5 /;
  loop (regions,
    put "                       ",
        regions.te(regions):5:0, consum.l(goods,regions):12:5 /;
  )
);
put/"region       trade balance     negishi weight" /;
put "------       -------------     --------------" /;
loop (regions,
  put "  ", regions.te(regions):5:0, tbalance(regions):18:8,
        negi.l(regions):18:8/;
);


** now, we make the Negishi weights variable and
** enforce budget constraints in each region.

model negmcp    / dutil.consum, prodLimit.pi, budget.negi /;

negi.lo(regions) = -INF;
negi.up(regions) = INF;
* add a numeraire, since the weights are unique only
* in a relative sense
negi.fx('1') = 1;

solve negmcp using mcp;

tbalance(regions) = production(regions)*pi.l(regions)
        - sum(goods, pi.l(goods) * consum.l(goods,regions) );
jointutil = sum(regions, negi.l(regions) *
        prod(goods, consum.l(goods,regions)**alpha(goods,regions) ) );

put //;
put "MCP solution, optimal weights:"/;
put/"joint utility:" jointutil:10:5//;
put "good     price       region  consumption" /;
put "----    -------      ------  -----------" /;
loop (goods,
  put "  ", goods.te(goods):3:0, pi.l(goods):10:5 /;
  loop (regions,
    put "                       ",
        regions.te(regions):5:0, consum.l(goods,regions):12:5 /;
  )
);
put/"region       trade balance     negishi weight" /;
put "------       -------------     --------------" /;
loop (regions,
  put "  ", regions.te(regions):5:0, tbalance(regions):18:8,
        negi.l(regions):18:8/;
);


* ==>pies.gms

$TITLE PIES Energy Equilibrium
$offsymxref offsymlist offuellist offuelxref
$inlinecom /* */
/* %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A linear program with a variable rhs in the constraint system
is expressed as a complementarity problem.

LP:     min     <c,x>
                Ax = q(p),
        s.t.    Bx = b,
                x >= 0

where the prices p are the duals to the first constraint.

MCP:           A'p + B'v + c >= 0, x >= 0, comp.
        -Ax + q(p)           =  0, p free, comp.
        -Bx              + b =  0, v free, comp.

Of course, the variables x and v are going to be
split up further in the GAMS model.

References:
William W. Hogan, Energy Policy Models for Project Independence,
Computers \& Operations Research (2), 1975.

N. Josephy, A Newton Method for the PIES energy model,
Tech Report, Mathematics Research Center, UW-Madison, 1979.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% */

sets
run     / run1 * run8 /,
comod   / C, L, H /,            /* coal and light and heavy oil */
R       / C, S /,               /* resources (capital, steel) */
creg    / 1 * 2 /,              /* coal producing regions */
oreg    / 1 * 2 /,              /* crude oil producing regions */
ctyp    / 1 * 3 /,              /* increments of coal production */
otyp    / 1 * 2 /,              /* increments of oil production */
refin   / 1 * 2 /,              /* refineries */
users   / 1 * 2 /;              /* consumption regions */

alias (comod,cc);

parameters
rmax(R) /               /* maximum resource usage */
C       35000
S       12000
```

```
/,
cmax(creg,ctyp) /       /* coal prod. limits */
1.1     300
1.2     300
1.3     400
2.1     200
2.2     300
2.3     600
/,
omax(oreg,otyp) /       /* oil prod. limits */
1.1     1100
1.2     1200
2.1     1300
2.2     1100
/,
rcost(refin) /          /* refining cost */
1       6.5
2       5
/,
q0(comod) /             /* base demand for commodities */
C       1000
L       1200
H       1000
/,
p0(comod) /             /* base prices for commodities */
C       12
L       16
H       12
/,
demand(comod,users),    /* computed at optimality */
output(refin,*) /       /* % output of light/heavy oil */
1.L     .6
1.H     .4
2.L     .5
2.H     .5
/;

table esub(comod,cc)    /* cross-elasticities of substitution */
        C       L       H
C       -.75    .1      .2
L       .1      -.5     .2
H       .2      .1      -.5 ;

table cruse(R,creg,ctyp) /* resource use in coal prod */
        1       2       3
C.1     1       5       10
C.2     1       5       6
S.1     1       2       3
S.2     1       4       5 ;

table oruse(R,oreg,otyp) /* resource use in oil prod */
        1       2
C.1     0       10
C.2     0       15
```

```
S.1     0       4
S.2     0       2 ;

table ccost(creg,ctyp)  /* coal prod. cost */
        1       2       3
1       5       6       8
2       4       5       7 ;

table ocost(oreg,otyp)  /* oil prod. cost */
        1       2
1       1       1.5
2       1.25    1.5 ;

table ctcost(creg,users)
        1       2
1       1       2.5
2       .75     2.75 ;

table otcost(oreg,refin)
        1       2
1       2       3
2       4       2 ;

table ltcost(refin,users)       /* light oil trans costs */
        1       2
1       1       1.2
2       1       1.5 ;

table htcost(refin,users)       /* heavy oil trans costs */
        1       2
1       1       1.2
2       1       1.5 ;


positive variables
c(creg,ctyp),           /* coal production */
o(oreg,otyp),           /* oil production */
ct(creg,users),         /* coal transportation levels */
ot(oreg,refin),         /* crude oil transportation levels */
lt(refin,users),        /* light transportation levels */
ht(refin,users),        /* heavy transportation levels */
p(comod,users),         /* commodity prices */
mu(R);                  /* dual to ruse cons.; marginal utility */

positive variables
cv(creg),               /* dual to cmbal */
ov(oreg),
lv(refin),
hv(refin);

c.up(creg,ctyp) = cmax(creg,ctyp);
o.up(oreg,otyp) = omax(oreg,otyp);

equations
```

```
delc(creg, ctyp),
delo(oreg, otyp),
delct(creg,users),
delot(oreg,refin),
dellt(refin,users),
delht(refin,users),
dembal(comod,users),            /* excess supply of product */
cmbal(creg),                    /* coal material balance */
ombal(oreg),                    /* oil material balance */
lmbal(refin),                   /* light material balance */
hmbal(refin),                   /* heavy material balance */
ruse(R);                        /* resource use constraints */

delc(creg,ctyp) ..
  ccost(creg,ctyp) + sum(R, cruse(R,creg,ctyp)*mu(R))
  =g=   cv(creg);

delo(oreg,otyp) ..
  ocost(oreg,otyp) + sum(R, oruse(R,oreg,otyp)*mu(R))
  =g=   ov(oreg);

delct(creg,users) ..
  ctcost(creg,users) + cv(creg)  =g=  p("C",users);

delot(oreg,refin) ..
  otcost(oreg,refin) + rcost(refin) + ov(oreg)   =g=
  output(refin,"L") * lv(refin) + output(refin,"H") * hv(refin);

dellt(refin,users) ..
  ltcost(refin,users) + lv(refin)
  =g=  p("L",users);

delht(refin,users) ..
  htcost(refin,users) + hv(refin)
  =g=  p("H",users);

dembal(comod,users) ..
  (sum(creg,ct(creg,users)))$(ord(comod) eq 1)
  + (sum(refin,lt(refin,users)))$(ord(comod) eq 2)
  + (sum(refin,ht(refin,users)))$(ord(comod) eq 3)
  =g=
  q0(comod) * prod(cc, (p(cc,users)/p0(cc))**esub(comod,cc));

cmbal(creg) ..
  sum(ctyp,c(creg,ctyp))  =e=  sum(users,ct(creg,users));

ombal(oreg) ..
  sum(otyp,o(oreg,otyp))  =e=  sum(refin,ot(oreg,refin));

lmbal(refin) ..
  sum(oreg, ot(oreg,refin)) * output(refin,"L") =e=
  sum(users,lt(refin,users));

hmbal(refin) ..
```

```
  sum(oreg, ot(oreg,refin)) * output(refin,"H") =e=
  sum(users,ht(refin,users));

ruse(R) ..
  rmax(R) =g=
  sum(creg, sum(ctyp, c(creg,ctyp)*cruse(R,creg,ctyp)))
  + sum(oreg, sum(otyp, o(oreg,otyp)*oruse(R,oreg,otyp)));


model pies / delc.c, delo.o, delct.ct, delot.ot, dellt.lt, delht.ht,
        dembal.p, cmbal.cv, ombal.ov, lmbal.lv, hmbal.hv, ruse.mu /;

option limrow = 0;
option limcol = 0;
option iterlim = 1000;
option reslim = 120;

table i_c(creg,ctyp)
        1       2       3
1       300     300     400
2       200     300     600 ;

table i_o(oreg,otyp)
        1       2
1       1100    1000
2       1300    1000 ;

table i_ct(creg,users)  /* initial trans */
        1       2
1       0       828
2       1016    84 ;

table i_ot(oreg,refin)  /* initial trans */
        1       2
1       2075    0
2       0       2358 ;

table i_lt(refin,users) /* initial trans */
        1       2
1       22      1223
2       1179    0 ;

table i_ht(refin,users) /* initial trans */
        1       2
1       0       830
2       998     180 ;

table iprice(comod,users)       /* initial price estimate */
        1       2
C       11.7    13.7
L       15.8    16.0
H       11.9    12.4 ;

c.l(creg,ctyp) = i_c(creg,ctyp);
```

```
o.l(oreg,otyp) = i_o(oreg,otyp);
ct.l(creg,users) = i_ct(creg,users);
ot.l(oreg,refin) = i_ot(oreg,refin);
lt.l(refin,users) = i_lt(refin,users);
ht.l(refin,users) = i_ht(refin,users);
p.lo(comod,users) = .1;
* p.fx(comod,users) = iprice(comod,users);
p.l(comod,users) = iprice(comod,users);

cv.l(creg) = 1;
ov.l(oreg) = 1;
lv.l(refin) = 1;
hv.l(refin) = 1;
mu.l(R) = 1;

solve pies using mcp;

$call echo foo > %gams.scrdir%xxyy1.txt
$call echo foo > %gams.scrdir%xxyy2.txt
$call echo foo > %gams.scrdir%xxyy3.txt

* execute ' ';

file  out /pies.out/;
put out;

put "Coal Prod:      type 1     type 2     type 3" /;
put "region 1   ", c.l("1","1"):11:3, c.l("1","2"):11:3, c.l("1","3"):11:3 /;
put "region 2   ", c.l("2","1"):11:3, c.l("2","2"):11:3, c.l("2","3"):11:3 /;
put /;
put "Oil Prod:       type 1     type 2" /;
put "region 1   ", o.l("1","1"):11:3, o.l("1","2"):11:3 /;
put "region 2   ", o.l("2","1"):11:3, o.l("2","2"):11:3 /;
put /;
put "Coal Trans:     user 1     user 2" /;
put "region 1   ", ct.l("1","1"):11:3, ct.l("1","2"):11:3 /;
put "region 2   ", ct.l("2","1"):11:3, ct.l("2","2"):11:3 /;
put /;
put "Oil Trans:     refin 1    refin 2" /;
put "region 1   ", ot.l("1","1"):11:3, ot.l("1","2"):11:3 /;
put "region 2   ", ot.l("2","1"):11:3, ot.l("2","2"):11:3 /;
put /;
put "Light Trans:    user 1     user 2" /;
put "refin 1    ", lt.l("1","1"):11:3, lt.l("1","2"):11:3 /;
put "refin 2    ", lt.l("2","1"):11:3, lt.l("2","2"):11:3 /;
put /;
put "Heavy Trans:    user 1     user 2" /;
put "refin 1    ", ht.l("1","1"):11:3, ht.l("1","2"):11:3 /;
put "refin 2    ", ht.l("2","1"):11:3, ht.l("2","2"):11:3 /;
put /;
put "Prices:         user 1     user 2" /;
put "Coal       ", p.l("C","1"):11:3, p.l("C","2"):11:3 /;
put "Light oil  ", p.l("L","1"):11:3, p.l("L","2"):11:3 /;
put "Heavy oil  ", p.l("H","1"):11:3, p.l("H","2"):11:3 /;
put /;
demand(comod,users) = q0(comod) *
         prod(cc, (p.l(cc,users)/p0(cc))**esub(comod,cc));
put "Demand:         user 1     user 2" /;
put "Coal        ", demand("C","1"):11:3, demand("C","2"):11:3 /;
put "Light oil  ", demand("L","1"):11:3, demand("L","2"):11:3 /;
put "Heavy oil  ", demand("H","1"):11:3, demand("H","2"):11:3 /;
put /;
put "Capital usage: ", (0-ruse.l("C")):10:2, "    dual price: ", mu.l("C"):9:3 /;
put "  Steel usage: ", (0-ruse.l("S")):10:2, "    dual price: ", mu.l("S"):9:3 /;
```

Answer from mccarl@masked.tamu.edu:

```
What you are doing was called reactive programming years ago.
There were algorithms and discussions of approaches in AJAE. A
little of it appears in a paper by chang et al in the american
journal of ag econ in 1992. Some references to reactive prog
appear in the ajae in an article in 1980 by mccarl and spreen .
Generally one uses a damping factor so when a new price is
predicted one uses something like half the difference from the
old price. Many today just use a nonlinear price endogenous
formulation and do not iterate
```

Answer from oyvind.hoveid@nlh10.nlh.no:

```
I have been thinking in a
similar direction and have learned at least some lessons.

Your scheme:

>> price ==> expected price ==> production ==> supply ==> new price

is one of sequential temporary equilibria. Dynamic stability is a keyword
here.Your overall model:

        price ==> new price

should have some type of dynamic stability along its path.
Some economists like to think that such models should have a
long-run solution in terms of a stable stationary state.
Possibly, you can find some expectations rule which leads to this
outcome. The difficulty here lies in the fact that the
appropriate expectations rule depends on the other elements of
your model. Look at the cob-web model: in certain circumstances
myopic expectations lead to long-run stability, in others it does
not.\\ A stationary long-run is nevertheless much to ask for. A
cyclic or chaotic long-run can be acceptable. What you definitely
want to avoid is an exploding model. But again, the remedy lies
in expectations in relation to the other parts of the model. If
you are unable to find suitable rules, you can resort to perfect
foresight or rational expectations by solving the model
simultaneously for a number of periods. Then you should either
choose a time horizon so distant that your exogenous end-point
does not matter for the time interval of interest, or you should
do some clever things at the end-point.
```

Answer from gideon.kruseman@alg.oe.wau.nl:

Normally one can assume that expectations and hence the expectations rule depends on the level of variability of the prices. If prices tend to be chaotic adjustment patterns are slow, while if prices tend to be stable (with for instance a steady trend) adjustment patterns will be fast.

new expected price = alpha * old expected price + (1- alpha)*
observed price

with alpha fluctuating between 0 and 1. Alpha tends towards zero if prices are chaotic and towards 1 if prices are stable. The problem is with modelling the occurrence of a (positive or neagtive) shock. Things are knocked about and it is unclear how to define expectation rules.

Answer from oyvind.hoveid@nlh10.nlh.no:

Yes, it is unclear indeed. Actually, there is no single theory saying how this should be modelled. Some appealing rules will not fit the remaining parts of your model (will lead to explosive behavior), and some rules that leads to sufficient stability, will not be appealing.

For a start try the model you suggest above, but let "alpha" be close to 0 when prices are unstable and closer to 1 when not. That model seems promising with respect to stability.

3.26 Integration

It is possible to use an integral in the GAMS program?

Answer from rutherford@colorado.edu:

I did some calculus of variations problems in GAMS as part of a course in trade and growth this past spring. In most of these applications, I illustrate how to discretize the time dimension in order to produce a finite-dimensional model which can be solved in GAMS. These are posted at:
http://robles.Colorado.EDU/~tomruth/8443/ks.htm

3.27 Help

The objective of the model has a charateristic that one of two exponential cost functions should be chosen to minimise total system costs subject to several constraints.

For example,

Suppose that there are two cost functions as below.
Cost function 1 : F(X)\\
Cost function 2 : G(X)
where X is a variable, not a parameter.

Two cost functions are all monotonously incrasing, as the value of X increases.   F(X) intersects G(X) on X=1000.

Cost function 1 should be chosen when 0<X<1000, whereas cost function 2 should be selected when 1000<X<+inf.

Answer from adrud@arki.dk:

You say that the two functions cross at 1000. Is this the only crossing point and which finction is higher above 1000? If F is higher that G below 1000 and F is lower than G above 1000 then you are infact minimizing the maximum value. This is done by introducing an extra variable and two extra equations:

```
variable Z;
equation Z_F, Z_G;
Z_F .. Z =3DG=3D F(X);
Z_G .. Z =3DG=3D G(X);

solve ... minimizing Z;
```

Answer from vlampe@agp.uni-bonn.de:

Test the following:

```
positive variables ff,gg
ff.up = 1
gg.up = 1

add. equations:

ff*(X-1000) =L= 0
gg*(X-1000) =G= 0
gg+ff=E=1

Minimize H(X)=ff*F(X)+gg*G(X)
```

Answer from mccarl@masked.tamu.edu:

```
This requires one of two approaches
integer programming if the marginal cost of producing x
at
x > 1000 is for any x value less with f than g
x < 1000 is for any x value greater with f than g

otherwise use
min f(x1)+g(x2)
x1+x2=x
x1<= 1000

if integer is needed introduce binary variable y

x1-1000y<=0
x2-1000(1-y)>=0
```

```
x2<=10000(1-y)
then you musy use dnlp

see coverage of this in my book on agrinet.tamu.edu/mccarl
in the integer programming chapter
```

3.28 Availability of livestock models in developing countries

```
I'm interested in finding out if anyone out there has any
knowledge of livestock models (farm level or sector) in
developing countries, not in the GAMS library.  My interest is in
finding out if any that are suitable for modifying to assess
livestock methane mitigation options (which affect feed
costs/intake, and animal productivity).

The most effective options seem to lie in dairy herds in the
"intensive" part of the sector.
```

Answer from anantha.duraiappah@ivm.vu.nl:

```
You may want to try RANGEPACK from Mark Stafford Smith in
Australia CISERO. It is however not a GAMS model and I am afraid
I have yet to come across one. I and a colleague developed a
livestock model in GAMS but this is for RAngelands where there is
no  artificial feeding. You are more than happy to have a look at
it. If yes, send me a note and I will send you the paper and you
can decide if it can be used for your study.
```

Answer from owitt@primal.ucdavis.edu:

```
I have a working template for a self calibrating sectoral or farm
level model for a mixed crop and livestock system that can be
downloaded from my website. The example is not for a developing
country, but could certainly be modified to reflect local
conditions as it has a dairy production, and a cropping, feeding
and grazing component. My website address is
http://www.agecon.ucdavis.edu/Faculty/Dick.H

Let me know if you have any questions. I have further developed
the model, but that version is not up on the site at the moment.
```

3.29 Integer constraint

```
How can I program in GAMS the following restriction:

                Y equals either 0 or 3.
                Y is an integer variable.
```

Answer from rutherford@colorado.edu:

```
How about:

binary variable x;
                variable y;

ydef..          y =e= 3 * x;
```

Answer from rrosenthal@monterey.nps.navy.mil:

To generalize the question, say the model is

```
min sum( j, c(j)*X(j) )
s.t.
sum( j, a(i,j)*X(j) )  b(i), for all i,
and X(j) must belong to the set { 0, n(j,1), n(j,2),..., n(j,10) }
```

X is a discrete variable that has at most 10 possible nonzero values for each j. The parameters n,a, and b are given.

You can solve this nonstandard problem with ordinary integer programming, using binary variables, as follows:

```
min sum( j, c(j)* sum( k, n(j,k)*Y(j,k) ) )
s.t.
sum( j, a(i,j)*sum( k, n(j,k)*Y(j,k) ) )  =  b(i), for all i
sum( k, Y(j,k) ) <= 1, for all j
Y binary.
```

You can of course simplify the nested summations but I wrote them this = way so you can see that all we are doing is substituting an expression = involving the binary variables for each X(j).

3.30 Delayed Response Models

I am attempting to model different fertilizer application levels taking into consideration groundwater pollution from leached nitrate. I regard nitrate pollution as cost to society while fertilizer applied as benefit to same society. The problem I face is Nitrate reaches ground water after a number of years from the time fertilizer was applied. This delayed response affects modeling since the cost of nitrate pollution is not incurred at the same time benefit from fertilizer application obtains.

Answer from mccarl@masked.tamu.edu:

You need a dynamic model where fertilization in year t worsens ground water quality in perid t+k to do this you would have accounts of nitrate in the groundwater for period t,t+1,...t+k,...t+n and do the accounting

3.31 Illustration of how to estimate and then simulate

Answer from rutherford@colorado.edu:

```
*----------------- main.gms -------------------

$title Illustration of how to estimate and then simulate

$ontext
```

```
This program illustrates ways that you might connect two GAMS models,
one which estimates parameters and another which does optimization on
the estimated model.

I've included flexibility with respect to how elasticities are to
be determined.  There are basically three options:

        (i) mode=estimate

            First estimate and then proceed directly to optimization.
            This is the classical approach in GAMS.  It would be
            possible to break this into two steps with a SAVE and
            RESTART, but it is essentially a linear flow of control.

        (ii) mode=datafile

           Read some elasticities values which have been computed
           in a previous estimation.  (This is what most CGE
           modelers assert to be doing.)

        (ii) mode=exogenous

            Specify all elasticities values exogenously.  (This is
            what most CGE actually do.)

I've included a fourth option call "compare" simply to make the
example more interesting.  In this mode of operation we first
estimate the parameter values and then compare the estimated
resulting equilibrium with the "true" outcome.  Finally, we compare
the estimated optimal tax with the true optimal taxes.

$offtext


set     j       markets         /a, b, c/;

parameter

  alpha(j)  elasticity of demand (true values) / a 0.7, b 0.2, c 1.4/,
  beta(j)   elasticity of demand used in optimal tax calculation;

*       Set the mode operation here: datafile, estimate,
*       exogenous, compare

$set mode estimate

$goto %mode%


*       Read a previously computed estimate and then
*       compute the optimal tax:

$label datafile
$include beta.dat
```

```
beta(j) = beta_est(j);
$include opttax
$exit

*       Exogenous -- use given values and skip estimation:

$label exogenous
beta(j) = alpha(j);
$include opttax
$exit

*       Estimate and then compute the optimal tax:

$label estimate
$label compare

*       Invoke GAMS code to estimate the model:

$include estimate

*       Assign a value of beta to use in the
*       subsequent optimal tax caluculation:

beta(j) = beta_v.l(j);

*       Compute some optimal taxes using the estimates:

$include opttax

*----------------- estimate.gms -------------------

$stitle Generate random data and then estimate beta

*       Generate a random dataset:

set     i       observations    /1*10/;

scalar          sigma   variance of the error term     / 0.5/;
parameter       d0(i,j) observed demand
                p0(i,j) observed price;

p0(i,j) = uniform(0.25, 4);
d0(i,j) = p0(i,j)**(-alpha(j)) + normal(0,sigma);


*       Estimate demand function parameters:

variable        beta_v(j)       estimated demand elasticities
                obj             ols minimand;

equations       objdef  defines the objective function;


objdef..
```

```
       obj =e= sum((i,j), sqr( p0(i,j)**(-beta_v(j)) - d0(i,j) ));


model ols /all/;

beta_v.lo(j) = 0.05;
beta_v.l(j) = 1;

solve ols using nlp minimizing obj;

*       Save the estimates of beta for subsequent use:

file kout /beta.dat/; put kout 'parameter beta_est /'/;
loop(j, put j.tl, beta_v.l(j)/;);
put '/;';

*----------------- opttax.gms -------------------
$stitle use beta to design an optimal tax

*       Assume that elasticities of supply are known:

parameter               eta(j)          elasticity of supply;

eta(j) = uniform(0.1, 0.8);


*       Now do an optimal tax calculation:

variables       p(j)    market price (net of tax)
                t(j)    ad-valorem tax rate
                r       public sector revenue
                d(j)    demand
                loss    dead weight loss of tax;

equations       market  market clearance
                revenue tax revenue
                demand  defines demand function d
                cost    defines marginal excess burden;

demand(j)..
  d(j) =e= (p(j)*(1+t(j)))**(-beta(j));

market(j)..
  p(j)**eta(j) =e= d(j);

revenue..
  r =e= sum(j, p(j) * t(j) * d(j));

cost..

  loss =e= sum(j, 0.5 * p(j) * t(j) * (p(j)**(-beta(j)) - d(j)));

model opttax / demand, market, revenue, cost/;
```

```
p.lo(j) = 0.01;
t.lo(j) = 0;
r.lo    = 0.5;

p.up(j) = 10;
t.up(j) = 1;

p.l(j) = 1;
t.l(j) = 0;

solve opttax using nlp minimizing loss;

$if not %mode%==compare $exit

*       mode=compare:

*       Do a comparison of optimal policy based on estimated
*       demand parameters with optimal policy based on true
*       parameters:

parameter       report          summary of optimal tax program;

report(j,"beta") = beta(j);
report(j,"alpha") = alpha(j);
report(j,"est_opttax") =
        100 * t.l(j);
report(j,"est_burden") =
         0.5 * p.l(j) * t.l(j) * (p.l(j)**(-beta(j)) - d.l(j));
report("total","est_burden") = loss.l;

*       Now see what the actual outcome would be:

beta(j) = alpha(j);
t.fx(j) = t.l(j);
solve opttax using nlp minimizing loss;

report(j,"act_burden") =
        0.5 * p.l(j) * t.l(j) * (p.l(j)**(-beta(j)) - d.l(j));
report("total","act_burden") = loss.l;

*       Finally, compute the true optimal policy assuming that we
*       knew the actual demand parameter values:

t.lo(j) = 0;
t.up(j) = 1;

beta(j) = alpha(j);

solve opttax using nlp minimizing loss;

report(j,"act_opttax") = 100 * t.l(j);
report(j,"min_burden") =
        0.5 * p.l(j) * t.l(j) * (p.l(j)**(-beta(j)) - d.l(j));
report("total","min_burden") = loss.l;
```

```
display report;
```

3.32 Sensitivity analysis

Answer from rrosenthal@monterey.nps.navy.mil:

```
Here is a simple example of solve-within-a-loop used for
sensitivity analysis in the Markowitz portfolio model.

$TITLE   THE MARKOWITZ MODEL FOR PORTFOLIO SELECTION
$offsymxref offsymlist
$ontext
     This nonlinear optimization model was developed by Harry
Markowitz in the 1950s.  At the time it was considered an academic
exercise.  Now it underlies much of the financial analysis that is
done routinely in the investment industry, though current models
are far more elaborate.  Markowitz received the 1990 Nobel Prize
in economics for this work.

    This instance of the Markowitz model was designed by Alan S.
Manne of the Department of Operations Research, Stanford
University.

    The problem is to find the optimal portfolio of investments
given a set of available securities to invest in, where "optimal"
is defined as the portfolio with minimum variability subject to
the constraint of achieving a desired mean return.

    Exercise:  Create efficient frontier by solving for optimal
portfolio with several possible values for target return.

$offtext

 SET  I  securities   /HARDWARE, SOFTWARE, SHOW-BIZ, T-BILLS/ ;

 SET  L  runs        / RUN-1 * RUN-7 / ;

 PARAMETER  TARGET(L) target mean annual return on portfolio (%)
 /
    RUN-1   10.75
    RUN-2   10.50
    RUN-3   10.25
    RUN-4   10.00
    RUN-5    9.75
    RUN-6    9.55
    RUN-7    9.25
 / ;

* Do you know why I put the targets in descending order?

 SCALAR  CURRENT  target value for current run (controlled by loop) ;

 PARAMETERS  MEAN(I)  mean annual return on security i (%)
```

```
     /HARDWARE   8
      SOFTWARE   9
      SHOW-BIZ  12
      T-BILLS    7 / ;

ALIAS (I,J) ;

TABLE V(I,J)  variance-covariance matrix (%-squared annual return)

                 HARDWARE  SOFTWARE  SHOW-BIZ  T-BILLS

      HARDWARE      4         3         -1        0
      SOFTWARE      3         6          1        0
      SHOW-BIZ     -1         1         10        0
      T-BILLS       0         0          0        0   ;

VARIABLES  X(I)      Fraction of portfolio invested in security i
           VARIANCE  Variance of portfolio ;

POSITIVE VARIABLE X;

EQUATIONS  FSUM    fractions must sum to 1.0
           DMEAN   define mean return of portfolio
           DVAR    define variance of portfolio ;

FSUM..     SUM(I, X(I))                     =E=  1.0  ;

DMEAN..    SUM(I, MEAN(I)*X(I))             =G=  CURRENT ;

DVAR..     SUM( (I,J), X(I)*V(I,J)*X(J) )   =E=  VARIANCE ;

MODEL PORTFOLIO  /ALL/ ;

PARAMETER REPORT(*,*)  Efficient Frontier of Optimal Portfolios ;

OPTION LIMROW = 0, LIMCOL = 0, SOLPRINT = OFF ;

LOOP (L,

   CURRENT = TARGET(L) ;

   SOLVE PORTFOLIO USING NLP MINIMIZING VARIANCE ;

   REPORT(I,L) = X.L(I);

   REPORT("MEAN",L) = DMEAN.L;

   REPORT("VARIANCE",L) = VARIANCE.L;

) ;  {end loop}

DISPLAY REPORT ;
```

3.33 Problems with regression model

I have specified that I want to maximize r squared , but the objective function value reported in the lst file is 0.15 and this made me wonder whether I had made a mistake somewhere in writing the program. In my program I have included a table which is actually TIME Vs REACTANT CONCENTRATION data. So the value in the 'time' column increases, while the value in next column decreases. In order to check the program , I added a simple table (where both columns data increase or both data decrease), and the program runs perfectly and gives a good solution with an r squared value 0.999.

Basically, my problem is one of regression. I have defined my r squared (r^2) just as in linear regression in terms of x and y. and have redefined x and y in terms of some other constants and variables. All the parameters are known except for 'a' which occurs in the equation for x. What I would like to do is determine the values of a, b and c associated with the best straight line fit by solving this optimization problem. Objective function to maximize r^2 or to minimize $-r^2$. subject to a >0 and the equations for b and c. So, essentially I need to get r^2 value (objective function) approximately equal to 1. But after I run my program I get 0.15, which is a poor straight line.

```
SET

DAT Data for expt /TIME,SUB/
j/1*9/;


TABLE D(j,DAT)

      TIME         SUB
1     0.01          11.5
2      2            8.8
3      3            6.7
4      5            5.4
5      7            2.0
6      9            0.18
7     11            0.16
8     13            0.15
9     15            0.01
;

PARAMETER SUB(J); SUB(J)= D(j,"SUB");

PARAMETER TIME(J); TIME(J)= D(j,"TIME");


VARIABLES
       x, y, r, a, b, c, n ,p, q, l, m;

EQUATIONS
```

```
    OBJ, V1, V2, V3, V4, V5 ,V6, V7, V8, V9, V10;

OBJ..     sqr(r) =E= p/q;

V1..      p =E= SQR((9*(SUM((j),x(j)*y(j)))) - SUM((j),x(j))*SUM((j),y(j)));


V2..      q =E= ((9*SUM((j),SQR(x(j))) - SQR(SUM((j),x(j))))
                 * ((9 * SUM((j),SQR(y(j)))) - SQR(SUM((j),y(j)))));

V3..      a =G= 0.00001;

V4(j)..    x(j) =E= log(1+ (a*(11.5-(SUB(J)))))/TIME(J) ;

V5(j)..     y(j) =E= (log(SUB(J)/11.5))/TIME(J);

V6..     b =E= l/m;

V7..     l =E= 9*(SUM((j),x(j)*y(j))) - SUM((j),x(j))*SUM((j),y(j));

V8..    m =E= (9* SUM((j),SQR(x(j)))) - (SQR(SUM((j),x(j))));

V9..     c =E= -(SUM((j),y(j)) - (b*SUM((j),x(j))))/9;

V10..     n =E= 9;

*LOWER BOUNDS

q.lo = 0.01;
m.lo = 0.01;
n.lo = 0.01;

MODEL am3 /ALL/ ;

SOLVE AM3 USING NLP MAXIMIZING r;
DISPLAY a.l, b.l, c.l, x.l, y.l, r.l;
```

Answer from adrud@arki.dk:

When using GAMS to do regression you should think in a different way. You have written down all the textbook equations used to compute R**2. These equations are actually the optimality conditions for the solution algorithm, so you are asking the solver to find a solution that both satisfies the optimality conditions and is optimal. Unfortunately, the optimality conditions are terribly non-convex, and a solution algorithm that tries to use gradual changes is likely to get stuck in a local solution. MINOS5 and our coming SNOPT solver gives the 0.15 you mentions, the new MINOS gives 0.30, and CONOPT cannot find a feasible solution.\\ The way to go is to define r**2 or the sum of squared diviations directly in terms of the parameters you are trying to estimate, something like this mini-model for linear regression:

```
variable a slope
```

```
         b intercept
         dsqr sum of squared diviations;

equation dsqrdef definition of rdsqr;

dsqrdef .. dsqr =e= sum(j, sqr( D(j,"SUB") - ( b + a * D(J,"TIME") ) ) );

model minsqr / dsqrdef /;

solve minsqr using nlp minimizing dsqr;
```

You are probably trying to solve a slightly more complex problem, but you can still use this general approach.

3.34 Calculation of initial values for NLP models

I‘m trying to solve a quite complex nonlinear model, with a lot of variables and equations. The model should plan a cost-optimal chemical mass exchange network. The main problem is that I had to construct as many non-zero initial values for the variables as it is possible, but I don‘t know how it can be done. If the set of initial values contains only a few non-zero elements than the solver (MINOS5) doesn‘t converge or gets stuck at the initial value.

Answer from jonathan_rubin@umit.maine.edu:

This is a common problem and I’m sure many list members will have their own solution. One that Paul Leiby and I have used is to find any acceptable solution to the model and use a post-processing routine to generate a file of initial values. This file then gets named, say "report.val" and then included in the GMS file. Paul has adapted a sub-routine written by Thomas Rutherford "wrtprm.inc" . Both files are attached below, but will need modifying by deleting our variable names and using yours. You will have to play around to get a solution that at least solves, use this (possibly bad) starting point and re-solve the model until you get to a local optimum. The sub-routines work from your saved work files.

3.35 Endogenous variable becomes exogenous

Under the assumption that I want to run only one .gms file, consider the following:
First step: A variable X is endogenous. I run the model to get the optimal value for X.
Second step: The optimal value of X is exogenous in order to derive the optimal value for a variable Y.

Answer from adrud@arki.dk:

After you have determined the value of X, for example using a SOLVE statement, you add the line

```
X.FX = X.L;
```

and X if now fixed at the solution value. If you want to free it again, you must change the lower and upper bounds bact to their original values, for example

```
X.LO = -INF; X.UP = +INF;
```

3.36 Cholesky Decomposition

I have an estimate of a variance-covariance matrix and I need the cholesky decomposition of it. Does anybody have any suggestions on how to program this in GAMS?

Answer from oyvind.hoveid@nlh10.nlh.no:

It is not that bad. The program enclosed seems to work. Basically, you need only one additional equation and one variable.

```
SETS
 j          variables                           /j1*j10/
 n         observations                         /n1*n10/
 b_l(j,j)   lower triangular marix;

ALIAS (j,jj,jjj);

b_l(j,jj) $ (ORD(j) GE ORD(jj)) = YES;

PARAMETER
 jn1(j,n)  variable observations
 b1(j,jj)   covariance matrix;

* forming a covariance matrix
jn1(j,n) = NORMAL(0,1); b1(j,j) = SUM(n, jn1(j,n)) /CARD(n);
DISPLAY b1; b1(j,jj) = SUM(n, (jn1(j,n) - b1(j,j))*(jn1(jj,n) -
b1(jj,jj)))/CARD(n); DISPLAY b1;

VARIABLES
 BB(j,jj)  Cholesky factors
 OBJ       objective

EQUATIONS
 B1_EQ(j,jj) Cholesky factorisation
 OBJ_DF ;

B1_EQ(j,jj) $ b_l(j,jj)..
 b1(j,jj) =E= SUM(jjj, (BB(j,jjj) $ b_l(j,jjj))*(BB(jj,jjj) $ b_l(jj,jjj)));

OBJ_DF..
 OBJ =E= 1;

MODEL CHOLESKY /ALL/ ;
```

```
BB.L(j,j) = 1;

SOLVE CHOLESKY MAXIMISING OBJ USING NLP
```

Answer from adrud@arki.dk:

```
Oyvind Hoveid has a solution based on solving a model. You can include
this inside a larger model so it is very flexible. If you just need the
Cholesky decomposition then you can compute it in GAMS with a loop as
shown below. It is an extension of Oyvinds program in which I increased
the number of observations to ensure that the matrix was positive
definite (10 observations gives a matrix that is very close to being
singluar).

There are additional example discussing Cholesky decompositions and
positive semidefinite matrices in the GAMS-L archive -- it is a
popular area.


SETS
 j           variables                            /j1*j10/
 n          observations                          /n1*n15/
 b_l(j,j)   lower triangular marix;

ALIAS (j,jj,jjj);

b_l(j,jj) $ (ORD(j) GE ORD(jj)) = YES;

PARAMETER
 jn1(j,n)  variable observations
 b1(j,jj)   covariance matrix;

* forming a covariance matrix
jn1(j,n) = NORMAL(0,1); b1(j,j) = SUM(n, jn1(j,n)) /CARD(n);
DISPLAY b1; b1(j,jj) = SUM(n, (jn1(j,n) - b1(j,j))*(jn1(jj,n) -
b1(jj,jj)))/CARD(n); DISPLAY b1;

set l(j,j) defines the lower triangle; parameters  w(j,j)    factor
of b parameters  ww(j,j)   part of b not yet factored sets   r(j)
the pivot row
       npr(j) rows not yet used for pivot
       npc(j) columns not yet used for pivot
scalar big    the diagonal pivot at each point in the factorization
       piv    the square root of the pivot
       tol    pivot tolerance;
r(j)     = yes; npr(j)   = yes; npc(j)   = yes; ww(j,jj) =
b1(j,jj); tol      = 1e-10; loop(r, big = ww(r,r);
   if ( big ge tol,
      piv = sqrt(big);
      l(npr,r) = yes;
      npr(r)   = no;
      npc(r)   = no;
      w(r   ,r) = piv;
      w(npr,r) = ww(npr,r) / piv;
      ww(npr,npc) = ww(npr,npc) - ww(npr,r)*ww(npc,r)/big;
      ww(npr,r) = 0;
      ww(r,npc) = 0;
      ww(r,r)   = 0;
      );
   );

display w, l;
```

Chapter 4

Modeling Examples and Tricks for MPSGE

4.1 Using set in endogenous tax field

I am currently working on a MPSGE model and I want to create an endogenous tax that look like that:
+ A:CN(IND) N:TXCOMSC M:TXCCOMM(IND,COMM,CONS)
However, even if the agents CN(IND) are defined, I get this error message from MPSGE:
*** CHKSET: unrecognized set argument: IND.
Is it possible do use set in an endogenous tax field if it not control in the O: or I: field? **Answer from rutherford@COLORADO.EDU:**

```
Sorry, but you are requesting a feature which is not provided by the
MPSGE compiler at present.  The only way to generate set drivers for an
input or output field is through the I: field.

Now, in your example if IND is a small dimensional set, then you could
enter each endogenous tax flow as a separate entry:

        A:CN("IND1")    N:TXCOMSC       M:TXCCOM("IND1",COMM,CONS)
+       A(CN("IND2") ...

This is, however, inelegant and somewhat inefficient -- the N: field
generates lots of entries in the Jacobian.

I am not sure how TXCCOM() is defined, but it may be the case that you
are simply sharing out the tax revening across a number of consumers
with fixed shares.  If this is the case, you could consider adding a tax
collector to the model, e.g.

        A:TXCOMSC_AGT   N:TXCOMSC

and then use a commodity market to allocate these tax revenues to the
various agents:

        $DEMAND:CN(IND)
               E:P_TXCOMSC     Q:THETA_TX(IND)
               ...

        $DEMAND:TXCOMSC_AGT
```

```
                D:P_TXCOMSC
```

If you use this formulation, you have the additional benefit of reading directly from the solution listing the aggregate tax revenue which appears as TXCOMSC_AGT.L.

4.2 An overlapping generations example model

Answer from rutherford@colorado.edu:

In response to a request for an overlapping generations example, I am attaching a simple model in which each generation lives for two periods. This is presented first as an MPSGE model, and the model is then restated as a complementarity problem using GAMS algebra.I would be reluctant to claim that every example in the Aurbach-Kotlikoff book can be processed in GAMS. They use a model-specific Gauss-Seidel algorithm and they begin from a calibrated benchmark. Here I start from a more-or-less arbitrary point, and I use a general-purpose complementarity algorithm which is effectively a Newton method. Perhaps someone with more algorithmic expertise could comment on the relative strengths of these approaches. From my perspective, the GAMS approach is far more appealing because I find it much easier to alter the model specification, leaving the job of finding an equilibrium to Michael Ferris (i.e., PATH). One thing you might note about these examples concerns the discretization of time. A-K use single year periods, and their models therefore have a lots of periods. In this model and another model which I will post momentarily, we use 10 year time intervals to reduce dimensions. My experience suggests that for many problems, this does not introduce a significant approximation error, but the magnitude of this error depends on the specific model structure and the nature of the policy questions being addressed. The program generates some graphics which can be viewed if you have the gnuplot package installed (see contributed software at http://www.gams.com). Remove the comments identified as "*." to include the gnuplot calls (note: I am currently working on a version of gnuplot which runs as a console application under Windows NT. If anyone has a compiled version for this target environment, I would love to hear from you!)

```
$Title  A simple overlapping generations model -- agents live two periods

set t time periods /1990,2000,2010,2020,2030,2040, 2050,2060,2070,2080,2090,2100/,

t1(t)   first time period,
        tl(t)   last time period;

scalar  k0 initial capital stock                /0.05 /
    kvs capital value share                 /0.3 /
    rk0 base year return to capital per unit of capital     /0.3/
    srvshr one period capital survival share        /0.5/
cshr    value share of first period consumption         /0.5/;

* Identify the first and last periods generically so that
* set t can be freely modified:
```

```
t1(t) = yes$(ord(t) eq 1);
tl(t) = yes$(ord(t) eq card(t));

*       Declare the MPSGE model:

$ontext

$model:olg

$sectors:
  y(t)    ! Output
  i(t)    ! Investment
  k(t)    ! Capital stock

$commodities:
  p(t)    ! Price index for output
  pk(t)   ! Price index for capital
  rk(t)   ! Rental price of capital
  w(t)    ! Wage index
  pkt     ! Post-terminal capital and consumption

*       One representative agent for each generation in the model.
*       The generation which is old in the first period is
*       generation 0:

$consumers:
        ra0
        ra(t)

*       Production in period t requires input of labor and capital
*       services:

$prod:y(t) s:1
        o:p(t)  q:1
        i:w(t)  q:1     p:(1-kvs)
        i:rk(t) q:1     p:kvs

*       Investment in period t generates capital in the subsequent
*       period.  Investment in the final period generates second
*       period consumption for the cohort born in the terminal period:

$prod:i(t)
        o:pk(t+1)       q:1
        o:pkt$tl(t)     q:1
        i:p(t)          q:1
*       Capital entering period t generates capital services in that
*       period and depreciated capital in the subsequent period.
*       Capital carried over from the final period produces second
*       period consumption for the terminal period cohort:
$prod:k(t)
        o:pk(t+1)       q:srvshr
        o:pkt$tl(t)     q:srvshr
        o:rk(t)         q:rk0
        i:pk(t)         q:1
```

```
*       The generation which is old in the first period is endowed
*       with capital:
$demand:ra0
        e:pk(t1)        q:k0
        d:p(t1)
*       Each generation is endowed with 1 unit of labor in the
*       first period of life and consumes output in both the first
*       and second periods of life.  The agent born in the final
*       period purchases terminal capital in place of output in
*       the post-terminal period.
$offtext
$sysinclude mpsgeset olg
*       Normalize prices using the income level of the
*       first generation:
ra0.fx = 1;
*       Solve the MPSGE model (from an arbitrary starting point):
$include olg.gen
solve olg using mcp;

*       Write out the MCP version of the model:

equations       prf_y(t)        Zero profit for production
                prf_i(t)        Zero profit for investment
                prf_k(t)        Zero profit for capital
                mkt_y(t)        Market clearance for output
                mkt_l(t)        Market clearance for labor
                mkt_rk(t)       Market clearance for capital services
                mkt_k(t)        Market clearance for capital stock
                mkt_kt          Market clearance for terminal capital
                inc0            Income balance for cohort 0,
                inc(t)          Income balance for period t cohort;
prf_y(t)..      (w(t)/(1-kvs))**(1-kvs) * (rk(t)/kvs)**kvs =e= p(t);
prf_i(t)..      p(t) =e= pk(t+1) + pkt$tl(t);
prf_k(t)..      pk(t) =e= rk0 * rk(t) + srvshr * (pk(t+1) + pkt$tl(t));
inc0..          ra0 =e= k0 * sum(t1, pk(t1));
inc(t)..        ra(t) =e= w(t);
mkt_y(t)..      p(t) * y(t) =e=
                cshr * ra(t) + (1-cshr) * ra(t-1) + ra0$t1(t) + p(t) *
i(t);
mkt_l(t)..      w(t) * 1 =e= 0.7 * p(t) * y(t);
mkt_rk(t)..     rk0 * rk(t) * k(t) =e= kvs * p(t) * y(t);
mkt_k(t)..      k0$t1(t) + srvshr * k(t-1) + i(t-1) =e= k(t);
mkt_kt..        pkt*sum(tl,srvshr*k(tl)+i(tl)) =e=
sum(tl,(1-cshr)*ra(tl));
model algebraic / prf_y.y, prf_i.i, prf_k.k, inc0.ra0, inc.ra,
                  mkt_y.p, mkt_l.w, mkt_rk.rk, mkt_k.pk, mkt_kt.pkt/;
*       Verify that the algebraic model has the same equilibrium
*       as the MPSGE model by issuing a solve with no iterations:
algebraic.iterlim = 0;
solve algebraic using mcp;
*       Generate a graphical report:
parameter int_r(t,*) interest rates;
int_r(t,"p")$(not tl(t)) = 100 * (1-p.l(t+1)/p.l(t));
int_r(t,"w")$(not tl(t)) = 100 * (1-w.l(t+1)/w.l(t));
int_r(t,"pk")$(not tl(t)) = 100 * (1-pk.l(t+1)/pk.l(t));
*.$libinclude gnuplot int_r
parameter price(t,*) future value prices;
price(t,"w")  = w.l(t)/p.l(t);
price(t,"pk") = pk.l(t)/p.l(t);
price(t,"rk") = rk.l(t)/p.l(t);
*.$libinclude gnuplot price
parameter quant(t,*) quantities of output investment and capital;
quant(t,"y")  = y.l(t);
quant(t,"i") = i.l(t);
quant(t,"k") = k.l(t);
*.$libinclude gnuplot quant
Back up to the top.
```

4.3 Another overlapping generations example model

Answer from rutherford@colorado.edu:

This message provides a second example of an overlapping generations model solved using GAMS, courtesy of Alan Manne. The model is formulated in the Negishi joint maximization framework and solved as a sequence of nonlinear programs. The files here also illustrate how the model can be formulated as a mixed complementarity problem. Find the following files:

- olg.gms Manne's model solved by sequential joint maximization
- disp.gms GAMS program to display results (restart from olg)
- olgmcp.gms Mixed complementarity model (restart from olg)
- olg.bat Batch file which executes olg, disp and olgmcp.

Incidentally, this model provides some useful insights into economic issues related to climate change and inter-generational equity. See the cited paper by Manne and related work by Schelling.

```
*----------------- cut here for olg.gms -----------------------
$TITLE   An Overlapping Generations Model of Carbon Dioxide Abatement
$ONTEXT
For a general description of this model, see Alan Manne, "Equity,
Efficiency and Discounting", Stanford University, December 1996.  He
employed an NLP formulation based on sequential joint maximization.
The MCP formulation is due to Tom Rutherford.
For details on abatement costs, see Tim Olsen, "Greenhouse Gas
Abatement - Joint Maximization under Uncertainty", doctoral
dissertation, Department of Operations Research, Stanford University,
November 1994.  The following file represents a deterministic
simplification of Olsen's original stochastic decision model.  The
world is treated as a single region.
$OFFTEXT

$OFFSYMXREF OFFSYMLIST
SETS
         YR        YEARS         / 1990*2200 /
         TP(YR)    TIME PERIODS (STARTING YEAR)
              /1990, 2000, 2010, 2020, 2030, 2040,
               2050, 2060, 2070, 2080, 2090, 2100,
               2110, 2120, 2130, 2140, 2150, 2160,
               2170, 2180, 2190, 2200/
         ITER0     USED FOR DISPLAY PURPOSES
                   /IT0*IT150/
         ITER(ITER0)   NEGISHI ITERATIONS
                   /IT1*IT150/;
SETS     PP(TP)     PROJECTION PERIODS (ALL PERIODS BUT THE BASE PERIOD)
         TBASE(TP)  BASE TIME PERIOD
         TLAST(TP)  LAST TIME PERIOD
         TPTC(TP)   PERIODS IN WHICH TERMINAL CONDITION APPLIES
                     /2200/ ;
PP(TP)    = YES$(ORD(TP) NE 1);
TBASE(TP) = YES$(ORD(TP) EQ 1);
TLAST(TP) = YES$(ORD(TP) EQ CARD(TP));
TABLE FYR(*,*)   FIRST YEAR OF TIME PERIOD
                    FY
           1990    1990
           2000    2000
           2010    2010
           2020    2020
           2030    2030
           2040    2040
           2050    2050
           2060    2060
           2070    2070
           2080    2080
           2090    2090
           2100    2100
           2110    2110
           2120    2120
           2130    2130
           2140    2140
           2150    2150
           2160    2160
           2170    2170
           2180    2180
           2190    2190
           2200    2200;

PARAMETERS
   NYPER(TP)       NUMBER OF YEARS IN A GIVEN PERIOD - FORWARD DIFF
   TE(TP)          YEARS ELAPSED SINCE BEGINNING OF FIRST PERIOD;
LOOP(TBASE,
     TE(TP) = FYR(TP,"FY") - FYR(TBASE, "FY")
    );
NYPER(TP) = TE(TP+1) - TE(TP);
NYPER(TP)$(ORD(TP) EQ CARD(TP)) = NYPER(TP-1);

* Each cohort lives for 8 decades.  No income or expenses during first
2;
* capital is accumulated from labor income during next 4; retirement in
last 2.
SET
```

```
         AC        AGE COHORTS (YEAR OF ENTRANCE INTO LABOR FORCE -
AFTER 1900)
                                      / 50,  60,  70,  80,  90, 100,
                   110, 120, 130, 140, 150, 160, 170, 180, 190, 200,
                   210, 220, 230, 240, 250, 260, 270, 280, 290, 300/
TABLE CONAC(TP, AC)   PERIODS IN WHICH COHORT'S CONSUMPTION IS POSITIVE
          50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210
220 230 240 250 260 270 280 290 300
2000       1  1  1  1  1   1
2010          1  1  1  1   1   1
2020             1  1  1   1   1   1
2030                1  1   1   1   1   1
2040                   1   1   1   1   1   1
2050                       1   1   1   1   1   1
2060                           1   1   1   1   1   1
2070                               1   1   1   1   1   1
2080                                   1   1   1   1   1   1
2090                                       1   1   1   1   1   1
2100                                           1   1   1   1   1   1
2110                                               1   1   1   1   1   1
2120                                                   1   1   1   1
1   1
2130                                                       1   1   1
1   1   1
2140                                                           1   1
1   1   1   1
2150                                                               1
1   1   1   1   1
2160
1   1   1   1   1   1
2170
1   1   1   1   1   1
2180
1   1   1   1   1   1
2190
1   1   1   1   1   1
2200
1   1   1   1   1   1

TABLE LFRAC(TP, AC) LABOR ENDOWMENT - FRACTION OF TOTAL OWNED BY COHORT
                 70  80  90 100 110 120 130 140 150 160 170 180 190 200
210 220 230 240 250 260 270 280 290 300
2000            .20 .30 .30 .20
2010                .20 .30 .30 .20
2020                    .20 .30 .30 .20
2030                        .20 .30 .30 .20
2040                            .20 .30 .30 .20
2050                                .20 .30 .30 .20
2060                                    .20 .30 .30 .20
2070                                        .20 .30 .30 .20
2080                                            .20 .30 .30 .20
2090                                                .20 .30 .30 .20
2100                                                    .20 .30 .30 .20
2110                                                        .20 .30 .30
```

```
.20
2120                                                             .20 .30
.30 .20
2130                                                                 .20
.30 .30 .20
2140
.20 .30 .30 .20
2150
.20 .30 .30 .20
2160
.20 .30 .30 .20
2170
.20 .30 .30 .20
2180
.20 .30 .30 .20
2190
.20 .30 .30 .20
2200
.20 .30 .30 .20

*  Labor earnings peak during second and third decades of working
career.

TABLE KFRAC(*,AC)  CAPITAL ENDOWMENT - FRACTION OWNED BY COHORT
          50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210
220 230 240 250 260 270 280 290 300
2000      .3 .6 .1

$ONTEXT
Stylized facts governing initial capital endowment:
       As of the initial projection period (2000), cohorts of 1950 and
1960 have each worked for 4 decades, and are currently retired.
       In 2000, cohort of 1960 has accumulated capital from 40 years'
worth
of earnings, and is about to enter its retirement phase.  The oldest
cohort
(1950) has spent some of its capital, and owns 30 years' worth of
earnings.
Its utility depends only on consumption in 2000.  Younger cohorts have
smaller
accumulations.
$OFFTEXT
DISPLAY CONAC, LFRAC, KFRAC;
OPTION SOLPRINT  = OFF;
OPTION LIMROW    = 0;
OPTION LIMCOL    = 0;
OPTION ITERLIM   = 100000;
OPTION RESLIM    = 20000;
PARAMETERS
     UDR(TP)         UTILITY DISCOUNT RATE FOR PERIOD TP
     UDF(TP)         UTILITY DISCOUNT FACTOR FOR PERIOD TP
     UDFSUM          UTILITY DISCOUNT FACTOR NORMALIZING CONSTANT
     UDFA(TP,AC)     UTILITY DISCOUNT FACTOR FOR PERIOD TP - AGE COHORT
AC
```

```
     UDFASUM(AC)     UTILITY DISCOUNT FACTOR NORMALIZING CONSTANT
     L(TP)           CURRENT LABOR FORCE (EFFICIENCY UNITS)
     GDP             INITIAL GDP ($ TRILLIONS 1990)
     KGDP            INITIAL CAPITAL-GDP RATIO
     ACQ(TP)         ABATEMENT COST PARAMETER - QUADRATIC
     ACB(TP)         ABATEMENT COST PARAMETER - BACKSTOP
     DEPR            ANNUAL DEPRECIATION RATE OF CAPITAL
     KSUR(TP)        CAPITAL SURVIVAL FACTOR
     KPVS            CAPITAL VALUE SHARE OF GDP
     LBVS            LABOR VALUE SHARE OF GDP
     NWT(AC)         NEGISHI WEIGHTS FOR EACH AGE COHORT
     GROW(TP)        ANNUAL LABOR FORCE GROWTH RATE
     A               PRODUCTION FUNCTION SCALE FACTOR
     I0              INITIAL INVESTMENT
     C0              INITIAL CONSUMPTION
     K0              INITIAL CAPITAL
     Y0              INITIAL GDP;

PARAMETER ZCAT       CATASTROPHIC CUM EMISSIONS LEVEL - BILLION TONS;
     ZCAT =  8000;
$ ONTEXT
    When the ZCAT parameter is set at 8000, zero abatement leads to the
following  values of EKF:
----    563 PARAMETER EKF           ENVIRONMENTAL CAPITAL FACTOR
1990 1.000,    2000 1.000,    2010 1.000,    2020 0.999,    2030 0.998
2040 0.997,    2050 0.994,    2060 0.991,    2070 0.986,    2080 0.979
2090 0.970,    2100 0.957,    2110 0.941,    2120 0.921
$OFFTEXT
TABLE MACRO(*, * )  MACROECONOMIC AND OTHER PARAMETERS
            WORLD
GDP         22.92
KGDP         2.8
DEPR         0.05
KPVS         0.28;
$ONTEXT
Terminology and sources used in MACRO table:
 GDP    INITIAL GDP - TRILLION $ 1990
 KGDP   INITIAL CAPITAL-GDP RATIO
 DEPR   ANNUAL DEPRECIATION RATE OF CAPITAL
 KPVS   CAPITAL'S VALUE SHARE IN COBB-DOUGLAS PRODUCTION FUNCTION
$OFFTEXT

TABLE PGROW(*,*)  POTENTIAL GDP GROWTH RATES - ANNUAL PERCENT
              WORLD
1990          2.50
2000          2.50
2010          2.50
2020          2.50
2030          2.50
2040          2.50
2050          2.50
2060          2.50
2070          2.50
2080          2.50
```

```
2090          2.50
2100          2.50
2110          2.50
2120          2.50
2130          2.50
2140          2.50
2150          2.50
2160          2.50
2170          2.50
2180          2.50
2190          2.50
2200          2.50 ;
TABLE ZBASE(*,*)  BASE (WITHOUT ANY ABATEMENT) CARBON EMISSIONS LEVELS
                   WORLD
1990               6.003
2000               7.225
2010               8.311
2020               9.902
2030              11.298
2040              13.441
2050              15.085
2060              17.191
2070              19.296
2080              21.838
2090              24.817
2100              27.796
2110              29.689
2120              31.582
2130              31.808
2140              30.367
2150              28.925
2160              27.432
2170              25.939
2180              24.277
2190              22.445
2200              20.613;

TABLE AQPER(*,*)  MAXIMUM LEVEL OF QUADRATIC ABATEMENT - PERCENTAGE OF
BAU
*  Based on Olsen's estimates for OOECD region.
             WORLD
2000         22.412
2010         33.134
2020         41.934
2030         48.810
2040         53.763
2050         56.793
2060         57.900
2070         57.084
2080         54.345
2090         49.684
2100         43.099
2110         43.099
2120         43.099
```

```
2130         43.099
2140         43.099
2150         43.099
2160         43.099
2170         43.099
2180         43.099
2190         43.099
2200         43.099;

*   Limit on abatement costs of $200 per ton of carbon.
ACB(PP) = 0.2;
ACQ(PP) = (0.5*ACB(PP)) / (0.01*AQPER(PP,"WORLD")*ZBASE(PP,"WORLD"));
DISPLAY ACQ;
*  Set initial macroeconomic parameters
GDP           = MACRO("GDP", "WORLD");
KGDP          = MACRO("KGDP","WORLD");
DEPR          = MACRO("DEPR","WORLD");
KPVS          = MACRO("KPVS","WORLD");
GROW(TP)      = PGROW(TP,"WORLD")/100;
KSUR(TP)      = (1-DEPR)**NYPER(TP);
LBVS          = 1 - KPVS;
UDR(TP)  =  .040;
*  Initial value of Negishi weights
NWT(AC) =  1/CARD(AC);
*  Labor force and utility discount factor for each time period
UDF(TBASE)  = 1;
L(TBASE)    = 1;
LOOP(TP,
     L(TP+1)      = L(TP)*(1 + GROW(TP))**NYPER(TP) ;
     UDF(TP+1)    = UDF(TP)*1/((1 + UDR(TP))**NYPER(TP)) ;
    );
UDFSUM       =  SUM(PP, UDF(PP));
UDF(TBASE)   =  0.0;
*  The following normalizations are required for Rutherford's SJM
algorithm.
UDF(PP)      =  UDF(PP)/UDFSUM;
*  Next we compute the utility discount factors for each age cohort.
UDFA(PP,AC)  =  UDF(PP) $ CONAC(PP,AC);
UDFASUM(AC)  =  SUM (PP $ CONAC(PP,AC) , UDFA(PP,AC));
UDFA(PP,AC)  =  UDFA(PP,AC)/UDFASUM(AC);
OPTION DECIMALS = 6;
DISPLAY UDF, UDFASUM, UDFA;

*  Inital capital and investment and consumption values.
K0 = KGDP*GDP;
LOOP(TBASE,
     I0 =   K0 * (GROW(TBASE) + DEPR);
    );
C0      =  GDP - I0;
Y0      =  GDP;
*  Cobb-Douglas production function scale factor.
A= Y0/(K0**KPVS);
POSITIVE VARIABLES
    K(TP)         CAPITAL STOCK - TRILLION $
    Y(TP)          CONVENTIONAL GDP - TRILLION $
    C(TP)          CONSUMPTION - TRILLION $
    CC(TP,AC)      CONSUMPTION BY AGE COHORT - TRILLION $
    I(TP)          INVESTMENT - TRILLION $
    AQ(TP)         QUADRATIC ABATEMENT LEVEL - BILLION TONS
    AB(TP)         BACKSTOP ABATEMENT LEVEL - BILLION TONS
    Z(TP)          CARBON EMISSIONS - BILLION TONS
    ZC(TP)         CUMULATIVE CARBON EMISSIONS - BILLION TONS;
VARIABLES
    NWEL              NEGISHI WELFARE;

EQUATIONS
*   MACRO SUBMODEL
    PRODT(TP)      COBB-DOUGLAS PRODUCTION FUNCTION
    CAPACCUM(TP)   CAPITAL ACCUMULATION
    TC(TP)         TERMINAL CONDITION ON INVESTMENT
    ALC(TP)        ALLOCATION OF AGGREGATE CONSUMPTION BETWEEN AGE
COHORTS
    NWELDF         NEGISHI WELFARE OBJECTIVE FUNCTION
*   CARBON ACCUMULATION
    EMIS(TP)       EMISSIONS DEFINITION
    EXP(TP)        BACKSTOP EXPANSION LIMITS
    CUME(TP)       CUMULATIVE CARBON EMISSIONS
    GG(TP)         GREEN GDP - AVAILABILITY AND ALLOCATION;
*   MACRO SUBMODEL - CONVENTIONAL NEOCLASSICAL GROWTH ASSUMPTIONS
PRODT(PP)..
        A * (K(PP)**KPVS * (L(PP)**LBVS)) =E= Y(PP);
CAPACCUM(TP+1)..
KSUR(TP) * K(TP) + .5 * NYPER(TP)   * (KSUR(TP) * I(TP) + I(TP+1))
        =E= K(TP+1);
TC(TPTC)..
     I(TPTC)  =G=  (GROW(TPTC) + DEPR) * K(TPTC);
*   AGGREGATE CONSUMPTION EQUALS SUM OF AMOUNTS CONSUMED BY EACH AGE
COHORT.
ALC(PP)..    C(PP)  =E=  SUM(AC $ CONAC(PP,AC) , CC(PP,AC));

*   NEGISHI WELFARE DEPENDS ON DISCOUNTED UTILITY OF EACH AGE COHORT.
NWELDF..
     NWEL  =E=  100*1000*SUM((AC), NWT(AC) *
                SUM(PP $ CONAC(PP,AC), UDFA(PP,AC) * LOG(CC(PP,AC))));

*   EMISSIONS DETERMINED BY BASE TRAJECTORY LESS ABATEMENT.
EMIS(PP)..
     Z(PP)   =E=  ZBASE(PP,"WORLD") - AQ(PP) - AB(PP);

*   BACKSTOP TECHNOLOGY CAN BEGIN IN 2050, BUT CANNOT EXPAND TOO
RAPIDLY.
EXP(TP) $(ORD(TP) GT 6) ..
        0.02 * ZBASE(TP, "WORLD")  +   2 * AB(TP) =G= AB(TP+1);
*   CUMULATIVE EMISSIONS - NEGLECTING ABSORPTION OF CARBON IN BIOSPHERE.
CUME(TP+1)..
     ZC(TP+1)  =E=  ZC(TP)  + .5 * NYPER(TP) * (Z(TP) + Z(TP+1));
*   CONVENTIONAL GDP IS REDUCED BY QUADRATIC FUNCTION OF CUMULATIVE
EMISSIONS.
```

```
*   THIS LINKS THE MACRO AND THE ENVIRONMENTAL SUBMODELS.
GG(PP)..
     (1 - (ZC(PP)/ZCAT)**2) * Y(PP)  =E=
                 C(PP) + I(PP)
               + ACQ(PP)*(AQ(PP)**2) + ACB(PP)*AB(PP);
MODEL  OLG  /ALL/;
OLG.OPTFILE = 1;
*  The following equations fix the base year values of the decision
variables.
K.FX(TBASE)   =  K0;
Y.FX(TBASE)   =  Y0;
C.FX(TBASE)   =  C0;
I.FX(TBASE)   =  I0;

AQ.FX(TBASE)  =  0;
AB.FX(TBASE)  =  0;
Z.FX(TBASE)   =     ZBASE(TBASE,"WORLD");
ZC.FX(TBASE)     =  0;

*  The following ensures constancy of EKF and post-terminal
sustainability.
Z.FX(TLAST)   =  0;
*  The following bounds help to avoid nasty program calls.
K.LO(PP)  =  K0;
Y.LO(PP)  =  Y0;
C.LO(PP)  =  C0;
CC.LO(PP,AC) = CONAC(PP,AC)*C0/CARD(AC);
CC.FX(PP,AC) $ (NOT CONAC(PP,AC)) = 0;
I.LO(PP)  =  I0;
ZC.LO(PP)    =  1.0;
ZC.UP(PP)    =  0.95*ZCAT;
*  Inserted following to fix capital endowment in 2000.
*  5% adjustment factor to smooth investment growth post-2000.
K.FX("2000")  =  1.05*L("2000")*K0;

*   LOOP FOR ITERATING OVER NEGISHI WEIGHTS
PARAMETERS
    NWTITR(ITER0,AC) NEGISHI WEIGHTS
    NDL(TP,AC)       PRESENT VALUE OF LABOR ENDOWMENT
    NDK(TP,AC)       PRESENT VALUE OF CAPITAL ENDOWMENT
    NDLSUM(AC)       CUMULATIVE PV OF LABOR ENDOWMENT
    NDKSUM(AC)       CUMULATIVE PV OF CAPITAL ENDOWMENT
    CSH(AC)          COHORT'S SHARE OF REGIONAL ENDOWMENT
    ENDOW            REGIONAL PV OF LABOR AND CAPITAL ENDOWMENTS
    NWT(AC)          REGIONAL COHORT'S NEGISHI WEIGHT;
    NWTITR("IT0",AC) = NWT(AC);
LOOP(ITER$(ORD(ITER) LE CARD(ITER)),
     SOLVE OLG MAXIMIZING NWEL USING NLP;
*    Begin computation of NWT(AC).
     NDL(PP,AC)   = LFRAC(PP,AC)*LBVS*Y.L(PP)*PRODT.M(PP);
     NDK(PP,AC)   = KFRAC(PP,AC)*KPVS*Y.L(PP)*PRODT.M(PP);
     NDLSUM(AC)   = SUM(PP, NDL(PP,AC));
     NDKSUM(AC)   = SUM(PP, NDK(PP,AC));
     NWT(AC)      = NDLSUM(AC) + NDKSUM(AC);
     ENDOW        = SUM(AC, NWT(AC));
     NWT(AC)      = NWT(AC)/ ENDOW;
     NWTITR(ITER,AC) = NWT(AC);
OPTION DECIMALS = 6;
    );
* The following parameters are all calculated for display purposes.
PARAMETERS
    ABC(TP)     ABATEMENT COSTS - TRILLION DOLLARS
    EKF(TP)     ENVIRONMENTAL LOSS FACTOR
    DFAC(TP)    DISCOUNT FACTOR FOR DCON - 5 PERCENT
    DCON        DISCOUNTED CONSUMPTION - 5 PERCENT
    PRC(TP)     FUTURE VALUE PRICE OF CARBON - DOLLARS PER TON
    PVG(TP)     PV PRICE OF CONSUMPTION AND GREEN GDP - NORMALIZED TO
2000
    CDR(TP)     CONSUMPTION AND GREEN GDP DISCOUNT RATE
    KL(TP)      CAPITAL-LABOR INDEX - YEAR 1990 EQUALS 1;

ABC(TBASE) = 0;
ABC(PP) = ACQ(PP)*(AQ.L(PP)**2) + ACB(PP)*AB.L(PP);
EKF(TBASE) = 1;
EKF(PP) = 1 - (ZC.L(PP)/ZCAT)**2;
DFAC(TBASE) = 1;
LOOP(TP,
     DFAC(TP+1) = ((1/1.05)**NYPER(TP)) * DFAC(TP);
    );
DCON      = 10*SUM((TP), DFAC(TP) * C.L(TP));
PRC(PP)                 = 1000 * EMIS.M(PP) / GG.M(PP);
PVG(PP)                 = GG.M(PP) /GG.M("2000");
CDR(TP)$(ORD(TP) GT 1)  = GG.M(TP+1) /GG.M(TP);
CDR(PP)                 = -100*((CDR(PP)**.1) - 1);
KL(TBASE) = 1;
KL(PP)    = K.L(PP)/(L(PP)*K0);
OPTION DECIMALS = 3;
DISPLAY K.L, I.L, C.L, Y.L;
DISPLAY CC.L;
DISPLAY AQ.L, AB.L, ABC;
DISPLAY ZBASE, Z.L, ZC.L, EKF, DCON;
DISPLAY PRC, PVG, CDR;
DISPLAY KL;
OPTION DECIMALS = 6;
DISPLAY NWTITR;
OPTION SOLPRINT=ON;
SOLVE OLG MAXIMIZING NWEL USING NLP;
*----------------- cut here for disp.gms ------------------------
OPTION DECIMALS = 3;
DISPLAY I.L, CDR, CC.L, PRC, Z.L, ZBASE, EKF;
PARAMETERS  KL(TP)      CAPITAL-LABOR INDEX - YEAR 1990 EQUALS 1;
KL(TBASE) = 1;
KL(PP)    = K.L(PP)/(L(PP)*K0);
DISPLAY K.L, KL;
*----------------- cut here for olgmcp.gms ------------------------
*   THESE ARE THE VARIABLE MULTIPLIERS ADDED FOR THE MCP MODEL
ALIAS (AC,AAC);
ALIAS (TP,TTP);
ALIAS (PP,PPP);
VARIABLE
INCOME(AC)      PRESENT VALUE OF FACTOR INCOME
SHARE(AC)       INCOME SHARE
PI_PRODT(TP)    COBB-DOUGLAS PRODUCTION FUNCTION
PI_CAPACC(TP)   CAPITAL ACCUMULATION
PI_TC(TP)       TERMINAL CONDITION ON INVESTMENT
PI_ALC(TP)      ALLOCATION OF AGGREGATE CONSUMPTION BETWEEN AGE COHORTS
PI_EMIS(TP)     EMISSIONS DEFINITION
PI_EXP(TP)      BACKSTOP EXPANSION LIMITS
PI_CUME(TP)     CUMULATIVE CARBON EMISSIONS
PI_GG(TP)       GREEN GDP - AVAILABILITY AND ALLOCATION;

*   ADDITIONAL EQUATIONS FOR MCP
EQUATIONS
    INCDEF(AC)        INCOME DEFINITION
    SHRDEF(AC)        INCOME SHARE
    PRF_K(TP)         CAPITAL STOCK - TRILLION $
    PRF_Y(TP)         CONVENTIONAL GDP - TRILLION $
    PRF_C(TP)         CONSUMPTION - TRILLION $
    PRF_CC(TP,AC)     CONSUMPTION BY AGE COHORT - TRILLION $
    PRF_I(TP)         INVESTMENT - TRILLION $
    PRF_AQ(TP)        QUADRATIC ABATEMENT LEVEL - BILLION TONS
    PRF_AB(TP)        BACKSTOP ABATEMENT LEVEL - BILLION TONS
    PRF_Z(TP)         CARBON EMISSIONS - BILLION TONS
    PRF_ZC(TP)        CUMULATIVE CARBON EMISSIONS - BILLION TONS;
*     THESE ARE THE ADDITIONAL MCP EQUATIONS
PRF_K(TP)..   - (PI_PRODT(TP) * A * KPVS * (K(TP) ** (KPVS - 1))
                         * (L(TP) ** LBVS))$PP(TP)
              - PI_CAPACC(TP+1) * KSUR(TP)
              + PI_CAPACC(TP)$(ORD(TP) GT 1)
              + (PI_TC(TP) * (GROW(TP) + DEPR))$TPTC(TP)
              =G= 0;

PRF_Y(TP)..   + PI_PRODT(TP)$PP(TP)
              - (PI_GG(TP)*(1 - (ZC(TP)/ZCAT)**2))$PP(TP)
              =G= 0;
INCDEF(AC)..    INCOME(AC) =E= SUM(PP, Y(PP)*PI_PRODT(PP) *
                        (LFRAC(PP,AC)*LBVS+KFRAC(PP,AC)*KPVS));
SHRDEF(AC)..    SUM(AAC, INCOME(AAC)) * SHARE(AC) =E= INCOME(AC);
PRF_C(TP)..   - PI_ALC(TP)$PP(TP) + PI_GG(TP)$PP(TP) =G= 0;

PRF_CC(PP,AC)$CONAC(PP,AC)..
              - PI_ALC(PP)
              + 100 * 1000 * SHARE(AC)* (UDFA(PP,AC) / CC(PP,AC))
              =E= 0;
PRF_I(TP)..   - (PI_CAPACC(TP) * .5 * NYPER(TP-1))$(ORD(TP) GT 1)
              - PI_CAPACC(TP + 1) * .5 * NYPER(TP) * KSUR(TP)
              - PI_TC(TP)$TPTC(TP)
              + PI_GG(TP)$PP(TP)
              =G= 0;
PRF_AQ(TP)..  - PI_EMIS(TP)$PP(TP)
              + (PI_GG(TP) * 2 * ACQ(TP) * AQ(TP))$PP(TP)
              =G= 0;
```

```
PRF_AB(TP)..  - PI_EMIS(TP)$PP(TP)
              - (PI_EXP(TP) * 2)$(ORD(TP) GT 6)
              + PI_EXP(TP-1)$(ORD(TP) GT 7)
              + (PI_GG(TP) * ACB(TP))$PP(TP)
              =G= 0;
PRF_Z(TP)..   - PI_EMIS(TP)$PP(TP)
              + PI_CUME(TP) * .5 * NYPER(TP-1)
              + PI_CUME(TP + 1) * .5 * NYPER(TP)
              =G= 0;
PRF_ZC(TP)..  + PI_CUME(TP+1)
              - PI_CUME(TP)$(ORD(TP) GT 1)
              - (PI_GG(TP) * (- 2 * Y(TP) * ZC(TP) / (ZCAT**2)))$PP(TP)
              =G= 0;

MODEL MCPMOD /
        PRODT.PI_PRODT, CAPACCUM.PI_CAPACC, TC.PI_TC, ALC.PI_ALC,
        EMIS.PI_EMIS, EXP.PI_EXP, CUME.PI_CUME, GG.PI_GG,
        PRF_K.K, PRF_Y.Y, PRF_C.C, PRF_CC.CC, PRF_I.I, PRF_AQ.AQ,
        PRF_AB.AB, PRF_Z.Z, PRF_ZC.ZC, INCDEF.INCOME, SHRDEF.SHARE/;
*       Assign values from the NLP solution:
PI_PRODT.L(TP)  = -PRODT.M(TP);
PI_CAPACC.L(TP) = -CAPACCUM.M(TP);
PI_TC.L(TP)     = -TC.M(TP);
PI_ALC.L(TP)    = -ALC.M(TP);
PI_EMIS.L(TP)   = -EMIS.M(TP);
PI_EXP.L(TP)    = -EXP.M(TP);
PI_CUME.L(TP)   = -CUME.M(TP);
PI_GG.L(TP)     = -GG.M(TP);
PRF_CC.LO(PP,AC) = 0;
PI_TC.LO(TP) = 0;
PI_EXP.LO(TP) = 0;
PI_CAPACC.FX(TBASE) = 0;
PI_CUME.FX(TBASE) = 0;
INCOME.L(AC) = SUM(PP, Y.L(PP)*PI_PRODT.L(PP) *
                        (LFRAC(PP,AC)*LBVS+KFRAC(PP,AC)*KPVS));
SHARE.L(AC) = NWT(AC);
OPTION SOLPRINT=ON;
MCPMOD.ITERLIM=0;
SOLVE MCPMOD USING MCP;
DISPLAY K.L, I.L, C.L, Y.L;
DISPLAY AQ.L, AB.L;
DISPLAY Z.L, ZC.L;
DISPLAY CC.L;
*       Normalize prices by fixing one income level:
INCOME.FX("50") = INCOME.L("50");
MCPMOD.ITERLIM=1000;
SOLVE MCPMOD USING MCP;

*----------------- cut here for olg.bat ------------------------
CALL GAMS OLG    S=OLG PS=9999 PW=85
CALL GAMS DISP   R=OLG PS=9999 PW=82
CALL GAMS OLGMCP R=OLG
```

4.4 A spatial equilibrium model with continuous piece-wise linear cost functions with discontinuous derivatives

Answer from rutherford@colorado.edu:

A graduate student from the University of Pennsylvania asked me how to formulation a spatial equilibrium model with continuous piece-wise linear cost functions with discontinuous derivatives. Here is a simple MCP example.

```
 $TITLE  Market Equilibrium with Step Functions - MCP Formulation
set i /1, 2/, t /t1,t2/;
alias (i,j);
parameter k(i,t) capacity;
k("1","t1") = 8/3;
k("1","t2") = 4-8/3;
k("2","t1") = 5/3;
k("2","t2") = 3-5/3;
table cost(i,t) cost
        t1      t2
1       .5      1.25
2       .8      2;
parameter d0(i) /1 71, 2 54/;
parameter c0(i) /1 1, 2 1.5/;
table alpha(i,j) transport cost cross-elasticity matrix
        1       2
1       1       0.5
2       2       1
table beta(i,j)  demand cross-elasticity matrix
        1       2
1       -3      1
2       1       -2
positive variables rho, d, p, y, r;
equations  demand, supply, mkt, profit, capacity;
demand(i)..     d(i) =g= d0(i) + sum(j, beta(i,j) * rho(j));
supply(i)..     p(i) + c0(i) + sum(j, alpha(i,j) * d(j))  =g= rho(i);
mkt(i)..        sum(t, y(i,t)) =g= d(i);
profit(i,t)..   r(i,t) + cost(i,t) =g= p(i);
capacity(i,t).. k(i,t) =g= y(i,t);
model stepsupply /demand.rho, supply.d, mkt.p, profit.y, capacity.r/;
solve stepsupply using mcp;
```

4.5 Changes in the I-O Matrix

I am working on Computable General Equilibrium(CGE) using Input-Output matrix to analyse agricultural pollution. I want to treat one of intermediate inputs, say, chemicals as valued added factor such as capital and labor. The underlying main reason is that the substitution between chemicals and other primary factors should be highlighted in the ag-pollution study. That is, 'chemicals' is NOT FIXED in my framework. Specifically, the concern is; If we classify 'chemicals' into 'value added' sector, how can we adjust the I-O matrix including coefficients and total inputs and outputs?

Answer from glennwh@ix.netcom.com:

Take a look at the GAMS web page at http://www.gams.com, particularly the MPSGE web pages under "solvers". You will see lots of template CGE models illyustrated there. You can formulate CGE models without MPSGE, say as NLP or MCP problems, and that is illustrated in the general GAMS model library.

Answer from rutherford@colorado.edu:

In short, the answer is you do not need to adjust the I-O matrix in order to convert chemicals into a price-responsive input. If you original cost function linear in intermediate inputs and Cobb-Douglas in value-added, then it has the form:

```
       C(p,w,r) = sum(i, a(i) * p(i)) + phi * w**alpha * r**(1-alpha)
```

You can make some of the inputs price-responsive by adding them to the value-added nest. Let v(i) denote the subset of intermediates to be added to the value-added nest, you then have:

```
       C(p,w,r) = sum(i$(not v(i)), a(i) * p(i)) +
                   gamma * prod(v, p(i)**beta(i)) * w**betal * r**betak;
```

where gamma, beta(i), betal and betak are all calibrated to the reference prices which apply for the variable inputs. The way to think about this is that the input-output matrix provides a reference quantity point but it provides not information about the relative price of variable inputs, labor and capital. For a concrete example, suppose that the input-output coefficients are:

```
       c, chemicals    1
        x, other inputs 2
        L, labor        1
        K, capital      1
```

and you assume that all benchmark prices equal unity. If this technology is linear in intermediate inputs and Cobb-Douglas in value added, it is then written:

```
       C(pc, px, pL, pK) = 0.2 * pc + 0.4 * px + 0.4 pL**0.5 * pK**0.5
```

(Note that the exponent for pL = 0.5 because the labor share in value added is 1/2.)

Now, if you add chemicals to the value-added nest, again assuming that all benchmark prices are unity, we have:

```
       C(pc, px, pL, pK) = 0.4 * px + 0.6 * pL**(1/3) * pK**(1/3) * pc**(1/3)
```

If this methodology strikes you as mysterious, I suggest that you have a look at Nicholson's graduate (masters-level) microeconomics text related to "anchored cost function". For a concise represenation of the use of calibrated functions, see http://www.gams.com/solvers/mpsge/cesfun.htm. This page is very informative, but I generally find that students generally have trouble getting the idea only from the mathematics. You have to solve some modelling problems to catch on to what it happening. Use GAMS, begin with a dataset, construct a small model, change the production structure and see how this change This is the only way to get a feeling for how to build models.

4.6 Marginal and average taxes in MPSGE

Answer from rutherford@colorado.edu:

```
$ontext
Here is a maquette which illustrates the endogenous
representation of marginal and average tax rates in a CGE model.  I
show how to do the calculation with and without the exact adjustment
of the average tax rate.  For this example, the precise
representation of the average tax is important only for calculating
the necessary replacement rate.
Given the following tax instruments:
       Marginal tax rate on factor income:
                       tm(Y) = 0.4 + (Y/Y0 - 1)**2
       Average tax rate on factor income:
                       ta = 0.3
       Consumption tax tc determined to assure constant
        level of public expenditure.
Social accounting matrix (rectangular form - row and column sums = 0):
       x       g        ls      c       ra      govt
----------------------------------------------------
px      100     -35             -65
pg              35                              -35
pl      -60             60
pls                     -36     -20     56
pk      -40                             28      12
pc                              90      -90
----------------------------------------------------
tm                      -24                     24
ta                                      6       -6
tc                              -5              5
----------------------------------------------------
Markets:
        px      Gross output (public and private consumption)
        pg      Public sector output
        pl      Wage payments (gross of income tax)
        pls     Value of time (net of income tax)
        pk      Return to capital (inelastically supplied)
        pc      Aggregate consumption
Production sectors:
       x       Gross output
        g       Public provision
        ls      Labor supply
        c       Final consumption
Consumers / budget balance conditions:
       ra      Final representative consumer
        govt    Public sector budget
Tax accounts:
       tm      Tax flow on labor supply at marginal tax rate of 40%
       ta      Lump-sum adjustment to produce an average tax rate of 30%
        tc      Consumption tax rate
```

```
Assume Cobb-Douglas production technology (sector X), CES preferences
over goods and leisure.
$offtext
scalar  ta      Average tax rate / 0.30 /
        phi     Productive efficiency /1/
        sigma   Elasticity of leisure-consumption substitution  /1/;
$ontext
$model:avemrg
$sectors:
        x       ! aggregate production
        g       ! public provision
        ls      ! labor supply
        c       ! final consumption
$commodities:
        px      ! output
        pg      ! public output
        pl      ! wage rate (gross of tax)
        pls     ! leisure value (net of tax)
        pk      ! capital return
        pc      ! final consumption
$consumers:
        ra      ! representative agent (tax payer)
        govt    ! government
$auxiliary:
        tc      ! consumption tax rate
        tm      ! marginal income tax rate
        tadj    ! income tax adjustment
$prod:x  s:1
        o:px    q:(100*phi)
        i:pl    q:60
        i:pk    q:40
$prod:g
        o:pg    q:35
        i:px    q:(35*phi)
$prod:ls
        o:pl    q:60    a:govt  n:tm
        i:pls   q:36
$prod:c  s:sigma
        o:pc    q:90
        i:px    q:65    a:govt n:tc  p:(1+5/65)
        i:pls   q:20
$demand:ra
        e:pk    q:40
        e:pk    q:(-ta*40)
        e:pls   q:56
        e:px    q:1     r:tadj
        d:pc    q:90
$demand:govt
        e:pk    q:(ta*40)
        e:px    q:-1    r:tadj
        d:pg
$constraint:tc
        g =e= 1;
*       This defines the marginal tax schedule:
```

```
$constraint:tm
        tm =e= 0.4 + ( (40*pk + 60*pl*ls) / (100 * px) - 1)**2;
*       The adjustment transfer equals the value of
*       tax payments at the marginal rate less the
*       tax liability calculated at the average rate:
$constraint:tadj
        tadj*px =e= 60*(tm-ta)*pl*ls;
$offtext
$sysinclude mpsgeset avemrg
*       Remove the default lower bound of zero on tc to
*       hold public expenditure fixed:
tc.lo = -inf;
tc.l = 5/65;
tm.l = 0.40;
tadj.l = 6;
*       Check the benchmark replication:
avemrg.iterlim = 0;
$include avemrg.gen
solve avemrg using mcp;
*       Consider the implications of a 10% increase in
*       efficiency and public goods requirements.  If the
*       tax system were non-distortionary, this should produce
*       a 10% increase in consumer welfare.  How do you think
*       the welfare gain depends on the elasticity of labor
*       supply?  The higher the elasticity, the higher the
*       excess burden of the tax system and the smaller the
*       gain, right?  Wrong.  Have a look at this simulation!
phi = 1.10;
parameter       results   Percentage changes (10% increase in
productivity);
$setglobal gp_xl sc
$setglobal gp_xlabel "Leisure-Consumption Elasticity"
*       If you want to do the comparison with a constant marginal
*       tax rate, uncomment the following line:
*.tm.fx = 0.4;
set sc /0*6/;
loop(sc,
        sigma = ord(sc);
        avemrg.iterlim = 2000;
$include avemrg.gen
        solve avemrg using mcp;
       results(sc,"ev") = 100 * (c.l-1);
        results(sc,"tm") = 100 * (tm.l/0.4-1);
        results(sc,"tc") = 100 * (tc.l/(5/65) - 1);
);
tadj.fx = 6;
loop(sc,
        sigma = ord(sc);
        avemrg.iterlim = 2000;
$include avemrg.gen
        solve avemrg using mcp;
       results(sc,"ev_fx") = 100 * (c.l-1);
        results(sc,"tm_fx") = 100 * (tm.l/0.4-1);
        results(sc,"tc_fx") = 100 * (tc.l/(5/65) - 1);
);
display results;
* Get gnuplot from the GAMS web site if you want to see the graph!
$libinclude gnuplot results
*       Do a second display so that it is possible to
*       estimate the minimum leisure-consumption elasticity
*       for which consumption increases at least proportionally
*       with productivity:
results(sc,"tc") = 0;
results(sc,"tc_fx") = 0;
$libinclude gnuplot results
```

4.7 GE modeling with transport emissions

In the context of an economic assessment of abatement measures for precursors of tropospheric ozone, I am looking for literature on the modeling of transport emissions (and related reduction strategies) in a General Equilibrium framework. Can anybody give me a hint?

Answer from rutherford@colorado.edu:

David Montgomery, Paul Bernstein and I built a model this past summer in which we modeled the effect of a carbon tax on the demand for automobiles in different countries. In this model we maintained a stock model of the automotive fleet in different countries. (You can obtain a copy by contacting the American Automobile Manufacturers Association in DC). We obtained data as best we could from the PARC -- a publication listing characteristics of current passenger autos and trucks in different countries. This sort of data is generally available from specialized consulting firms which put out news letters and such. The more detailed the data, the more expensive. Considering your location, you might give a call to an economist at Damler Benz and see if they have any suggestions concerning these statistics. So far as the general equilibrium framework, once you have the auto/truck statistics, it is then relatively straigtforward to merge the sectoral data into the GE dataset. An added benefit for this type of work is that the new GTAP release 4, due in April, will include a break-out for automobile trade.

Answer from glennwh@ix.netcom.com:

Bengt Kristrom and I did some simulations with a CGE model of Sweden, to look at the effects of some proposed transport policy reforms on CO2 emissions. This is coming out in some conference volume, but you can access the paper in PDF format from my web page CV.

4.8 A Primer in dynamic GE modeling

Answer from rutherford@colorado.edu:

GAMS general equilibrium modellers: Morten Lau, Andreas Pahlke and I have just finished an expository paper on dynamic general equilibrium modeling with GAMS. The paper is accessible in HTML (Netscape) format at http://robles.Colorado.EDU/~tomruth/primer/paper.htm. You can solve alternative scenarios over the net by connecting to http://nash.colorado.edu. (If you have not used the GAMS/CGI test site, you will be asked to create an account, but this is free and only takes a minute.)

4.9 Building Applied General Equilibrium Models in GAMS. A users guide and a library of applications

```
Based on the book "The Structure of Applied General Equilibrium Models" by
Victor Ginsburgh and Michiel Keyzer (MIT-Press), Michiel Keyzer has written a
library of numerical applications in GAMS which can be used with the book. The
programs and a Windows 95 or DOS interface, including a user's guide can be
downloaded from http://mitpress.mit.edu/books/GINTH/GAMS.html
```

4.10 Negative Inventories in MPSGE

```
I have included  the changes in inventories (several of them are
negative) in a short-run AGEM. When I put the model in MPSGE
format I get the correct warning (Q: value less than zero) and the
model calibrates correctly. Can these negative quantities affect
the counterfactual equilibria (or better, the work of the solvers
MILES and PATH) in any way?
```

Answer from rutherford@colorado.edu:

```
In some circumstances, you can include a negative Q: field in a
production record, but this is not always the best way to handle
things.

Consider a model with investment demand modeled as a constant
marginal propensity to save, and where inputs to investment flow
through a production activity which produces a composite investment
commodity, i.e.:

        $prod:invest
                o:pinv  q:i0
                i:p(i)  q:id0(i)

        $demand:ra  s:1
                d:pinv  q:i0
                ...

Now, suppose that some of the id0() vector are negative values,
corresponding to inventory reductions over the base year.  I would
probably model those elements of investment demand as fixed
quantities, rather than to assume that they would increase or
decrease with savings.

To make the negative entries exogenous, you could include the
following code:

        parameter       ix0(i)  exogenous investment demand vector;

        ix0(i) = min( 0, id0(i) );

        id0(i) = max(0, id0(i) );

        i0 = sum(i, id0(i));
```

Then alter the final demand block to read:

```
        $demand:ra  s:1
                e:p(i)  q:(-ix0(i))
                d:pinv  q:i0
```

Now the investment sector has nonnegative inputs, and the level of stock change is exogenously fixed at the base year level.

4.11 CGE with Integer Number

I am working with Gams on CGE (Computable General Equilibrium) models . I have just bought the version with MINLP solver (Dicopt) because I would like to build a CGE with constraints in integer number (for example the production capacity of a sector depending on the number of factory built, which arise only in integer). For the moment I have a version that runs with rminlp but that does not run with minlp and I do not understand why . So I would have liked to know if someone tried already to build this kind of models and if the solver Dicopt allows to do it. Thanks in advance

Answer from rutherford@colorado.edu:

Think carefully about your equilibrium concept before you start trying to compute equilibria with discrete numbers of firms. Once the production set is discrete, then you lose the basis for perfectly competitive behavior. Computation of equilibria becomes problematic for all but the simplest of models -- for starters, there is no guarantee of existence of equilibria in many cases. (See Scarf's various publications during the 80s beginning with the two Econometrica articles. He wrote several nice expository pieces including, I believe, a paper which was published in one of the OR journals.)

I wrote a paper with Jim Markusen in which we assumed discreteness of choice for auto producers (Weltwirtshchaftliches Archiv, 1994 -- see also the work of our CU graduate who is at Hawaii, Denise Eby). The equilibria we computed in the auto model were selected by computing a grid of outcomes and then finding a Nash equilibrium in the game between automobile producers.

4.12 Perfect elasticity in CGE

I am working on CGE using MINOS5. In the model I divide ag-products into risky and safe ones depending upon the land types. In reality risky and safe ones are homogeneous to consumers. So, I have applied the CES with high values of elasticities(say, 999999) between the risk and the safe in intermediate inputs, household consumption, and trade. Some problems are 1) the replication is not comfortable: some have significant discrepancy between results and base year data. Especially by taking policy scenarios(price wedge of fertilizers or reduction of risky land), the degree of changes are so different depending on the number of 9s. I use nesting CES in production and consumption, and intermediate inputs (say, cotton-risky and cotton-safe)having high elasticities with holding some of risky and safe cottons are constrained by given IO coefficients. 2)with the above problem in mind, I think that the discrepancy is inherent when we use high value of elasticities in CES. Is there any way that we can apply CES having high values of elasticites?

Answer from adrud@arki.dk:

Elasticities above 10 will usually give you severe numerical problems in the solver, and the results you report are probably just reflections of 'noise' in the solution.

Ideally, you should change the model so the CES between safe and risky products is replaced by

P(safe) = P(risky) = P(composite)

and

QDem(composite) = Q(Safe) + Q(Risky)

which are the mathematical equivalents of the usual equations in case of an infinite elasticity.

This formulation works fine as long as both safe and risky production tecnologies can be used at positive levels. If prices are such that one become unprofitable, then the production level Q should become zero. In this case you must change the price relationship and use a complementarity formulation, and you must change to a CNS solver such as PATH.

4.13 Balancing SAM

For my Jakarta cge model, I am expanding an I-O of Jakarta into a SAM data set. However, I have a problem in balancing my SAM. Does any body has a program to balance my SAM.

Answer from glennwh@ix.netcom.com:

The SAMBAL program in the GAMS library gives you some clues, but once you know what constraints you want to impose on the final matrix, just use GAMS to set up the coinstrained optimization problem directly. The one thing I have noticed in some of these problems, working with Tom Rutherford over the years to balance disparate data sets in this way, is that you want to use the flexibility of GAMS to set up several objective functions and see which ones work best. In any event, focus on the economics first and then use GAMS directly, rather than looking for some canned routine to do the job. In the end you will find it much more satisfactory.

4.14 CES function in MPSGE syntax

I have a basic question about how to consider the scaling parameter (gamma) of a CES production function in MPSGE syntax:

Given the following algebraic function:

```
  Q =E= gamma*[delta*LABOR**((s-1)/s) +
(1-delta)*CAPITAL**((s-1)/s)]**(s/(s-1))

where   gamma  = 1.5
        delta  = 0.6
        s      = 2
```

This example is part of a closed 2-2 economy CGE given by Shoven and Whalley ("Journal of Economic Literature", Vol. XXII, 1984, pp. 1010). I try to translate it into MPSGE syntax, but using the following formulation does not lead to a solution:

```
  $PROD:MF        s:2
              O:PMF   Q:1.5   P:(2/3)
              I:PL    Q:0.6
              I:PK    Q:0.4
}
```

Answer from rutherford@colorado.edu:

Try:

```
        $prod:mf  s:1
                o:pmf   q:1.5
                i:pl    q:1     p:0.6
                i:pk    q:1     p:0.4
```

There are an infinite number of points on a given isoquant, each of which could be taken to calibrate a given CES function. The trick is to choose a point which is convenient. Given your functional form, the easiest point to use for calibration is one where LABOR=CAPITAL=1. At that point Q=gamma. Finally, at that point the value share of labor is delta and the value share of capital is 1-delta. If the quantities equal unity, then the reference prices must be proportional to the value share.

Your use of a reference price on a single output coefficient suggests that you are having some problems understanding what is going on.

The reference price fields are used solely to establish the marginal rate of substitution or transformation (i.e. the slope of an isoquant) at the benchmark point. If there is a single output, the reference price is irrelevant. If there is a single input, the reference price is also irrelevant. The same is true if all inputs enter a Leotief aggregate.

What matters is the slope of the isoquant which equals the ratio of the reference prices. In the example shown above, you would obtain the same function if you were to write:

```
        $prod:mf  s:1
                o:pmf   q:1.5
                i:pl    q:1     p:1.5
                i:pk    q:1
```

in which the reference price for PK is taken to be 1 by default.

4.15 MPSGE question

I have a question concerning the P: and the T: fields of the following production block: (this is a part of Tom's GTAP3 model):

```
$PROD:Y(I,R)$VOM(I,R)  s:0  t:ETA  VA:1
        O:PY(I,R)       Q:VDM(I,R)      A:RA(R) T:TY(I,R)
        O:PX(I,R)       Q:VXM(I,R)      A:RA(R) T:TY(I,R)
        I:PA("I",J,R)   Q:VAFM(J,I,R)   A:RA(R) T:TI(J,I,R)
        I:PF(F,R)       Q:VFM(F,I,R) P:PF0(F,I,R) VA: A:RA(R)
T:TF(F,I,R)
```

For outputs and inputs taxes are being charged (TY, TI and TF) as you can see. The P: field signifies the reference price which, including the tax rate, should be 1 + tax rate. This means that (assuming positive taxes) in every line of the $PROD-block we should find a P: field (which is greater than 1). How come that only in the last line such a P: field has been defined, despite the fact that the reference price in all the other lines is also greater than 1? There should be P: fields in the other lines as well, right?

Answer from rwigle@wlu.ca:

No... It depends on what the units are. In the case of the outputs, I expect the quantities (VDM, VXM) are in gross of tax "units." in that case, one unit means the amount puchased for $1 at gross of tax prices. In the case of the inputs, the units are likely net of the factor tax.

Answer from rutherford@colorado.edu:

Consider the following calibrated unit cost functions,   CES(p) = c0 * sum(i, theta(i) * (p(i)/p0(i))**(1-sigma))**(1-sigma)

and

LEO(p) = sum(i, q0(i) * p(i))

Both functions are calibrated to a benchmark point defined by q0(i), the benchmark demand, and p0(i), the benchmark reference price. We then compute share and scaling parameters:

```
        c0 = sum(i, q0(i) * p0(i));

        theta(i) = p0(i) * q0(i) / c0;
```

Notice that the second function is a special case of the first when we consider sigma=0. As most intermediate micro students know, Leontief is a special case of CES.

Now, if we agree on this I will turn to the question.

> How come that only in the last line such a P: field has been defined,
> despite the fact that the reference price in all
> the other lines is also greater than 1? There should be P:
> fields in the other lines as well, right?

The answer is simply no. If the top level elasticity in a CES cost function is zero, then all inputs entering at the top level can be calibrated using the reference demand alone (Note that p0(i) does not appear in the Leontief cost function.)

If you draw a two factor Leontief input demand function, you will note that representation of the function at a given point is independent of the slope of the benchmark price line tangent to the corner of the L.

Of course, it would not create a problem having a reference price for a Leontief cost input. I guess that by habit I omit the reference price when it is not required in order to economize on generation time.

Answer from rwigle@wlu.ca:

I also posted an "answer" to this question, but I assumed that the answer was a different one. However in the interest of self education, my proposed answer suggests a follow-up:

When benchmarking, we normally choose units for the goods such that 1 unit buys $1 (million or billion) dollars something at some valuation which is as common as possible. Often it's the world price or some price that's net of tax. So here is my follow-up

If the units for the inputs in the data are gross of the relevant tax, is it possible for the P: fields to be left out in the $PROD record? (If so do we need to have a P: field specified somewhere else?)

Excuse me if Tom has already ‘corrected’ me.

Answer from rutherford@colorado.edu:

The P: field tells MPSGE what to

assume about the local marginal rate of substitution at the benchmark point. If no P: field is specified, MPSGE assumes that the MRS is unity.

When the elasticity of substitution between two inputs is zero, then the benchmark MRS has no effect on the cost function coefficients, so the P: field can be omitted. This was the basic message in my last email.

All of the forgoing is completely independent of the choice of units. It applies if you measure apples in pounds, kilos, bushels, tons, billions of dollars gross of tax, or billions of dollars net of tax.

If you wish to always specify a benchmark price, whether or not one is necessary, then the price field should represent the benchmark price level of the input good (at market prices) multiplied by one plus all taxes which apply in the benchmark.

As you point out, the definition of market prices is relevant. If units of market goods are measured as monetary values gross of tax, then the reference price of inputs would be unity and the P: field could be omitted. If market goods are measured as values net of tax, then the tax presumably applies on the input and the benchmark price would be 1 plus the tax rate. In this case the P: field is optional if the function is Leontief, required otherwise.

4.16 Congestion model

Answer from rutherford@colorado.edu:

GAMS/MPSGE programmers -- here is an answer to an offline question I received which may be of interest to others.

I've not had a chance to fully investigate your model, but on a cursory inspection I encountered a programming problem. You have specified a function:

```
$PROD: CAR s:0 DRV:9999
        O: PCAR         Q: car0
        I: PX           Q: carcost0     A: RA   T: TAX
        I: PROAD        Q: road0        DRV:
        I: PL           Q: lcar0        DRV:
```

in which you apparently want to have PL and PROAD entering as perfect substitutes. The way to specify two inputs as perfect substitutes within an MPSGE production sector is to represent the sector with three activities, as illustrated here:

```
$PROD:CAR  s:0
        O:PCAR          Q:car0
        I:PX            Q:carcost0  A:RA  T:TAX
        I:PDRV          Q:(road0+lcar0)

$PROD:DRV_ROAD
        O:PDRV          Q:road0
        I:PROAD         Q:road0

$PROD:DRV_PL
        O:PDRV          Q:lcar0
        I:PL            Q:lcar0
```

If the original benchmark is calibrated, then note that this specification retains market clearance with all prices and activity levels equal to unity. When relative prices of PL and PROAD change, then only the less expensive of these will be used as an input to CAR, and the activity corresponding to the more expensive input will fall to zero.

At times I have flirted with the idea of having the language "automatically" interpret an infinite elasticity of substitution to add the appropriate additional activities; but I decided that this was too complicated -- it conflicts with my view of MPSGE as a "reduced instruction set language". So, for the time being, if you want to make inputs into perfect substitutes; then you need to do the coding yourself.

4.17 Income transfers and negative savings in CGE-models

Currently I try to calibrate a balanced SAM (see attachment) to a CGE using the Cameroon-model in MPSGE-Syntax ('CAMmge.gms', see GAMS library) as a blueprint. Now I have two questions:

1) In contrast to the Cameroon data, there are a lot of inter-institutional transfers, e.g. transfers from firms to households or the income tax payment from household to government. I specify them as Endowments (positive for income, negative for expenditure / transfers are either specified as constant payments or, in the case of taxes, as constant ratio of the relevant income) in the demand block. Now my question is, which price index refers to these payments in the MPSGE demand block?
The model consists of the price indices listed below. I have tried to include the transfers into the PSAV variable. However, I am not sure wheather this is correct...(as usual, an $INVESTOR endows the total net savings and demands investment goods produced by a NEW CAPITAL FORMATION $PROD-block):

```
$COMMODITIES:
        PFX             ! REAL EXCHANGE RATE
        PD(I)           ! DOMESTIC SUPPLY PRICE
        P(CM)           ! PRICE INDEX FOR ARMINGTON AGGREGATE
        PE(I)$IT(I)     ! EXPORT PRICE INDEX
        PM(CM)$CMT(CM)  ! IMPORT PRICE INDEX
        PL(LC)          ! LABOR PRICE INDEX
        RK(I)           ! PRICE INDEX FOR EXISTING CAPITAL
        PK(I)           ! PRICE INDEX FOR NEW CAPITAL
        PSAV            ! PRICE INDEX FOR SAVINGS TRANSFER


$DEMAND:GOVT
```

```
        E:P(CM)          Q:(-government consumption)

        E:PSAV           Q:(tax income + transfer income
                            - transfer expenditure)

        E:PFX            Q:( transfer income from abroad
                            - transfer expenditure from abroad)

        D:PSAV           Q: government savings
```

2) In my present SAM, ENTERPRISE and GOVERNMENT savings are negative (see attached file). According to the CAMEROON-model, Government-Consumption is assumed to be fix (specified as a negative endowment). Thus, the only 'demanded thing' in the DEMAND block are the savings. Given the negative savings in my data set, I am not sure, whether this lead to problems when I calibrate the model ($FUNLOG and $WALCHK give me the follwing warning message:

```
--- Warning: income for GOVT =   -2.417E+04 is negative.  Reset to zero.
```

Answer from rutherford@colorado.edu:

OK, lets first address the issue of what the message means. When you see a record in the listing file like:

```
--- Warning: income for GOVT =   -2.417E+04 is negative.  Reset to zero.
```

this means that you have a real problem in the model formulation. When MPSGE get a starting point (a vector of prices, p, activity levels, y, and auxiliary variables, u), the function evaluation determines income levels as a function of these variables, something like:

```
M(h) = sum(i, p(i) * E(i,h)) + sum((i,j), p(i) * y(j) * T(i,j,h;u))
```

where I am using E(i,h) to stand for the endowment of good i by household h, and T(i,j,h;u) to stand for the tax payment per unit activity by sector j to household h based on transactions in commodity i. (T() is actually a function of the input and output coefficients, but specifying it in this way makes the notation more compact).

Provided that an agent's income is not fixed exogenously MPSGE recomputes the income levels in this way in every iteration. (The default specification is for one agent's income to be specified exogenously, provided that you have not fixed a price or income level explicitly.)

Now, what do you think that MPSGE should do if one or more M(h) values in a given iteration are negative? We have a problem which does not relate to the solution algorithm -- this has to do with the economic theory. This situation is ruled out by assumption in Debreu's formulation, because he assumes that all endowments are non-negative.

It is my experience that this rarely occurs during the solution process when the equilibrium is well specified, but it can occur. A typical problem would be a situation in which the public sector subsidies to firms are large relative to their purchases.

I'll pass your SAM along to one of my graduate students and have him produce a sample model consistent with the data. That is probabaly the most efficient way to help you with the problem. The basic approach I advocate for dealing with these inter-agent transfers is to represent them initially as a single lump-sum transfer. For example, you should be able to extract base year consumption expenditures (C0(i,h)) and primary factor endowments (E0(f,h)) from the SAM. Then you would specify final demand as:

```
        $DEMAND:RA(h)
                E:PL    Q:E0("L",h)
                E:RK    Q:E0("K",h)
                E:PL    Q:LSUM(h)
                D:P(i)  Q:C0(i,h)
```

where $LSUM(h) = sum(i, C0(i,h)) - sum(f, E0(f,h))$

Then you have:

```
        $DEMAND:GOVT
                E:PL    Q:(-sum(h, LSUM(h)))
                ...
```

Arne's discussion is on target regarding the representation of "institutions" in an MPSGE model. The MPSGE formulation is close to the original Shoven-Whalley model, so the only explicit components of the model are markets, producers and consumers. Producers convert goods -from some markets into goods in other markets. Consumers sell initial endowments and purchase goods. That is it so far as the core elements of an MPSGE model.

Now, in many social accounting matrices you will find detailed accounts of financial transfers from institutions to households, or from firms to households. I believe that most of these transfers are essentially represented exogenously in any model; or these levels are determined by ad-hoc rules. If these flows are exogenous, then it is simply a matter of representing them as lump-sum transfers between consumers in the model, e.g.:

```
        $demand:consumer_a
                e:pc    q:(-payment)
                ...

        $demand:consumer_b
                e:pc    q:(+payment)
```

Now some transfers will be represented as part of a model's closure. For example, suppose that you have a government which collects taxes, buys goods, and invests; and you want to have the investment level be a residual. Assuming that all government income arises from taxes on production, you might want the level of government investment to be chosen such that:

```
        psav * gsav = T - sum(i, p(i) * g0(i));
```

where gsav is the endogenous level of public savings (a real value), T is government tax revenue, and g0(i) is the exogenous level of public expenditure on goods. If you were sure that gsav would always be positive, this closure could be represented as:

```
        $demand:govt
                e:p(i)  q:(-g0(i))
                d:psav
```

A problem arises when you want to have this closure formulated so that gsav may be either positive or negative. To do this, you need to add an auxiliary variable, with a side constraint. I would add a non-negative variable auxvar such that

```
        $demand:govt
                e:p(i)  q:(-g0(i))
                e:psav  q:1     r:auxvar
                d:psav

        $constraint:auxvar
                govt =g= epsilon * psav;
```

where epsilon is some small number, the precise value of which depends on how the dataset is scaled.

The reason that these aspects of the GE formulation do not appear in the Shoven-Whalley model is that they make it impossible to do formal welfare analysis. If the government budget is not balanced, then it is impossible to distinguish between a change in consumer welfare which is due to efficiency improvements and one which arise solely because the government is running a deficit. In the long-run the government budget has to be balanced; and only in that setting does the model provide consistent estimates of welfare.

If you wanted to have the same model I outline above, but balance the budget through lump-sum transfers from one or more consumers, then it is simply a matter of introducing the auxiliary variable with offsetting entries in the government and household demand blocks:

```
        $demand:household(h)

*       theta(h) is the fraction of the lumpsum tax
*       paid by household h; where sum(h, theta(h)) = 1:

                e:psav  q:(-theta(h))   r:lstax
                ...

        $demand:govt  s:0
                d:p(i)  q:g0(i)
                e:psav  q:(-gsav)
                e:psav  q:1     r:lstax

        $constraint:lstax
                govt =e= sum(i, p(i) * g0(i)) + psav * gsav;
```

Depending on the scenario, the lumpsum tax may be positive or negative, so you need to remove the deault lower bound of zero on auxiliary variables:

```
lstax.lo = -inf;
```

In rereading what I just posted, it occured to me that the meaning of the side constraint could probably be better explained. I suggested in the case of having negative government savings, you could add an nonnegative auxiliary variable and specify:

```
      $demand:govt
                e:p(i)  q:(-g0(i))
                e:psav  q:1     r:auxvar
                d:psav

        $constraint:auxvar
                govt =g= epsilon * psav;
```

In the constraint govt represents government income, a variable from the MPSGE mode which would be equal to:

```
        govt = T - sum(i, p(i) * g0(i)) + psav * auxvar;
```

Note that we do not need to add this equation as a constraint -- it is included automatically as an income constraint for the govt consumer.

When auxvar is positive, i.e., when the government is running a negative savings level, we have:

```
        epsilon * psav = T - sum(i, p(i) * g0(i)) + psav * auxvar;
```

or

```
        -(auxvar - epsilon) * psav = T - sum(i, p(i) * g0(i))
```

Here we would identify gsav = -(auxvar - epsilon).

Answer from ferdinand.pavel@rz.hu-berlin.de:

At first I would like to thank Arne Stolbjerg and Glenn Harrison for their answers to my question and especially Tom Rutherford for his helpful support! I guess Ph-D students working on GAMS/MPSGE based CGEs around the world would be worse off without them!
I have adopted Tom's suggestion and feel much better now! However, I still have a problem:

The SAM I send around is for Bulgaria. As its exchange rate is fix (1.000

BgLev = 1DM), I would like to change the macro-closure. Tom suggested a variable tariff rate for balancing the state budget. In contrast, I would like to reduce tariffs in my analyses. Thus, the money for balancing the budget must come from somewhere else. In the case of a fixed exchange rate, it seems reasonable to let the government finance a shortfall in the budget by foreign borrowing, as I found it in an example for the CFA zone countries. Thus, in Tom's maquette I want to have the level of 'public sector imports' to be chosen such that:

```
        pfx * mg = T + M - pc * gc - pinv * ginv
```

where T + M are government tax and tariff revenue (assumed to be the only source of income), gc government consumption, ginv public savings and mg the level of public sector imports.

In MPSGE syntax I would specify this as follows (analogous to the approach in Tom's second mail, where the investment level was assumed to be the residual):

```
        $demand:govt
                e:p        q:gc
                e:pinv    q:ginv
              e:pfx     q:1     r:auxvar
                d:pfx

        $constraint:auxvar
                govt =g= epsilon * pfx;
```

I have included this into the model by defining the import duty in the normal way and replace the government demand block by the one given above. Now, when I compute a zero tariff equilibrium, this again leads to a negative income for GOVERNMENT, because the level of 'auxvar' does not change:

```
                       LOWER     LEVEL     UPPER    MARGINAL

----- VAR RK              .        1.155     +INF       .
----- VAR PL              .        1.140     +INF       .
----- VAR PINV            .        0.957     +INF       .
----- VAR RA              .      407.445     +INF       .
----- VAR GOVT            .         .        +INF     58.700
----- VAR FINV            .       14.296     +INF       .
----- VAR AUXVAR          .         .        +INF       EPS
```

I have attached the modified version of the model to this mail. Hope, there is still someone willing to help! Thanks in advance,

```
$title  Building an MPSGE Model from a Balanced SAM

*       -------------------------------------------------------------------------------
*       This set declaration simply assures that my summary
*       report has rows and columns in the desired order:

SET     REPORT /Tariff, EV, ER, Wage, "Wage$","Return","Percent","Price","Employ"/;
SET     ACCT    Accounts in the SAM
                / FOODPRD, NFOODPD, FOOD, NFOOD, LABOR, CAPITAL,
                  HHOLD, ENTRPS, GOVNT, CAPFRM, ROW, TOTAL      /

        H(ACCT) Representative agent    /HHOLD,ENTRPS/,
        S(ACCT) Sectors                 /FOOD, NFOOD /,
        I(ACCT) Commodities             /FOODPRD, NFOODPD/,
        F(ACCT) Factors                 /LABOR, CAPITAL/;


ALIAS(H,HH);
ALIAS(ACCT,ROWS,COLUMNS);

TABLE SAM(*,*)  The Social Accounting Matrix

*       Rows represent "sinks" -- where goods are distributed
*       Columns represent "sources" -- where goods come from.

        FOODPRD NFOODPD FOOD    NFOOD   LABOR   CAPITAL HHOLD   ENTRPS  GOVNT   CAPFRM  ROW
FOODPRD                 123608  13790                   118961                  3564
NFOODPD                 43543   493637                  268689          90497   45796
FOOD    206187  2996                                                                    49125
NFOOD   6244    674996                                                                  187645
LABOR                   76443   249786
CAPITAL                 8080    71235
HHOLD                                   326229  51176           112020  74573
ENTRPS                                          28138   9787    128502  76805           8984
GOVNT   23541   48066   57      -6572           1       102513  52918   88              11962
CAPFRM                  6577    47009                   60637   -57608  -20119          12864
ROW     23951   216104                                  3411    16384   10730                   ;

*       Scale the SAM so that a typical transaction is roughly:

SAM(ROWS,COLUMNS) = SAM(ROWS,COLUMNS) / 1000;

*       Check that the SAM balances:

SAM("TOTAL",COLUMNS) = SUM(ROWS, SAM(ROWS,COLUMNS));
SAM(ROWS,"TOTAL") = SUM(COLUMNS, SAM(ROWS,COLUMNS));
SAM(ACCT,"ERROR") = ABS(SAM(ACCT,"TOTAL")-SAM("TOTAL",ACCT));

ABORT$(SMAX(ACCT, ABS(SAM("TOTAL",ACCT)-SAM(ACCT,"TOTAL"))) GT 1.E-5)
        "Benchmark SAM does not balance?", SAM;

*       --------------------------------------------------
*       After a few minutes looking at the transactions, I started to
*       extract submatrices to compose the MPSGE model.  I began with
*       the most obvious stuff and moved on, declaring parameters,
*       checking consistency and adding lines to the model.  By the
*       time I had extracted all the numbers, I was ready to do a
*       calibration check.
*       Extract transactions related to production and check for zero profit:

parameter       FD0(F,S)        Factor demand by sector
                S0(I,S)         Commodity supply by sector
                ID0(I,S)        Intermediate demand by sector
                X0(S)           Sectoral exports
                STAX0(S)        Sectoral tax payments
                TL(S)           Implicit labor tax net subsidy
                PL0(S)          Benchmark user cost of labor
                DEPR0(S)        Depreciation
                PROFIT(S)       Benchmark profit (should be zero);

FD0(F,S) = SAM(F,S);
S0(I,S) = SAM(S,I);
ID0(I,S) = SAM(I,S);
X0(S)   = SAM(S,"ROW");
STAX0(S)  = SAM("GOVNT",S);
DEPR0(S) = SAM("CAPFRM",S);

PROFIT(S) = SUM(I, S0(I,S)) + X0(S)
            - SUM(F, FD0(F,S)) - SUM(I, ID0(I,S)) - STAX0(S) - DEPR0(S);
DISPLAY PROFIT;

TL(S) = STAX0(S) / FD0("LABOR",S);
PL0(S) = 1 + TL(S);

PROFIT(S) = SUM(I, S0(I,S)) + X0(S)
            - SUM(F, FD0(F,S)) - SUM(I, ID0(I,S)) - DEPR0(S) - TL(S) * FD0("LABOR",S);
DISPLAY "Profit check with labor tax:", PROFIT;

*       Extract transactions related to representative agent and check for
*       income balance:

parameter       FE0(F)          Primary factor endowment
                TRNSF0          Transfers from government
                FEXCH0          Private receipts of foreign transfers
                C0(I)           Private final demand
                ITAX0           Income tax payment
                SAV0            Net private saving
                MC0             Imports to final demand
                BUDGET          Private budget balance;

FE0(F)  = SUM(HH,SAM(HH,F));
TRNSF0  = SUM(HH,SAM(HH,"GOVNT"));
FEXCH0  = SUM(HH,SAM(HH,"ROW"));
C0(I)   = SUM(HH,SAM(I,HH));
ITAX0   = SUM(HH,SAM("GOVNT",HH));
SAV0    = SUM(HH,SAM("CAPFRM",HH));
MC0     = SUM(HH,SAM("ROW",HH));

BUDGET = SUM(F, FE0(F)) + TRNSF0 + FEXCH0
       - SUM(I,C0(I)) - ITAX0 - SAV0 - MC0;
DISPLAY BUDGET;
```

```
*       Extract transactions related to government and check for
*       public sector balance:

PARAMETER       G0(I)           Public demand for goods
                GC0             Total public demand
                MG0             Public sector imports
                GINV0           Public sector savings
                GE0(F)          Public sector factor demand
                GEXCH0          Government receipt of foreign remittances
                MTAX0(I)        Import tariff revenue
                PUBBDGT         Public sector budget;

G0(I)   = SAM(I,"GOVNT");
GC0 = SUM(I, G0(I));
MG0     = SAM("ROW","GOVNT");
GINV0   = SAM("CAPFRM","GOVNT");;
GE0(F)  = SAM("GOVNT",F);
GEXCH0  = SAM("GOVNT","ROW");
MTAX0(I) = SAM("GOVNT",I);
PUBBDGT = SUM(I, G0(I)) + TRNSF0 + GINV0 + MG0 - SUM(F, GE0(F))
        - SUM(S, STAX0(S)) - SUM(I, MTAX0(I)) - ITAX0 - GEXCH0;
DISPLAY PUBBDGT;

*       Extract transactions related to imports and check
*       for market clearance in markets for the Armington
*       aggregate:

PARAMETER       A0(I)           Aggregate Armington supply
                D0(I)           Total domestic supply
                M0(I)           Total imports
                TM(I)           Import tariff rate
                PM0(I)          Import price
                I0(I)           Investment demand
                MKT(I)          Market clearance;

D0(I) = SUM(S, S0(I,S));
M0(I) = SAM("ROW",I);
TM(I) = SAM("GOVNT",I) / M0(I);
PM0(I) = 1 + TM(I);
A0(I) = D0(I) + M0(I) * PM0(I);
I0(I) = SAM(I,"CAPFRM");

MKT(I) = A0(I) - SUM(S, ID0(I,S)) - C0(I) - G0(I) - I0(I);
DISPLAY MKT;


*       Extract transactions related to investment:

PARAMETER       INV0    Total investment
                FINV0   Foreign investment
                MKTINV  Market clearance for investment;

INV0 = SUM(I, I0(I));
FINV0 = SAM("CAPFRM","ROW");
```

```
MKTINV = INV0 - SUM(S, DEPR0(S)) - GINV0 - SAV0 - FINV0;
DISPLAY MKTINV;

*       Check foreign exchange balance:

PARAMETER       MKTFX   Market clearance for foreign exchange;
MKTFX = SUM(S, X0(S)) + FINV0 + FEXCH0 + GEXCH0
        - SUM(I, M0(I)) - MG0 - MC0 ;
DISPLAY MKTFX;

$ONTEXT
$FUNLOG:.TRUE.

$MODEL:MAQUETTE

$SECTORS:
        Y(S)    ! Sectoral output
        INV     ! Investment
        A(I)    ! Armington aggregation
        C       ! Consumer demand

$COMMODITIES:
        PFX     ! Foreign exchange
        PC      ! Consumer price index
        PD(I)   ! Domestic output
        P(I)    ! Supply price of the Armington composite
        RK      ! Return to capital
        PL      ! Wage rate
        PINV    ! Investment

$CONSUMERS:
        RA      ! Private consumers
        GOVT    ! Government
        FINV    ! Foreign investment

$AUXILIARY:
        AUXVAR     ! Equal yield multiplier

$PROD:INV
        O:PINV  Q:INV0
        I:P(I)  Q:I0(I)

$PROD:Y(S)  t:4  D(t):0  s:0 VA:1
        O:PFX   Q:X0(S)
        O:PD(I) Q:S0(I,S)       D:
        I:P(I)  Q:ID0(I,S)
        I:PINV  Q:DEPR0(S)
        I:RK    Q:FD0("CAPITAL",S)      A:GOVT  VA:
        I:PL    Q:FD0("LABOR",S)        A:GOVT  T:TL(S)  P:PL0(S) VA:

$REPORT:
        V:EMPLOY(S)     I:PL    PROD:Y(S)

$PROD:A(I)   s:4
```

```
        O:P(I)  Q:A0(I)
        I:PFX   Q:M0(I)         A:GOVT  T:TM(I)  P:PM0(I)
        I:PD(I) Q:D0(I)

$DEMAND:GOVT
        D:PFX
        E:PFX   Q:1     R:AUXVAR
        E:P(I)  Q:(-G0(I))
        E:PINV  Q:(-GINV0)
        E:RK    Q:GE0("CAPITAL")
        E:PFX   Q:(GEXCH0+ITAX0-TRNSF0)

$PROD:C
        O:PC    Q:(SUM(I,C0(I))+MC0)
        I:P(I)  Q:C0(I)
        I:PFX   Q:MC0


$DEMAND:RA  s:1
        D:PC
        E:PFX   Q:(TRNSF0+FEXCH0-ITAX0)
        E:PINV  Q:(-SAV0)
        E:RK    Q:FE0("CAPITAL")
        E:PL    Q:FE0("LABOR")

$DEMAND:FINV
        E:PFX   Q:FINV0
        D:PINV  Q:FINV0



$CONSTRAINT:AUXVAR
        GOVT =G= EPS * PFX;

$OFFTEXT
$SYSINCLUDE mpsgeset MAQUETTE


AUXVAR.L = 0;
MAQUETTE.ITERLIM = 0;
$INCLUDE MAQUETTE.GEN
SOLVE MAQUETTE USING MCP;

*       Compute the zero tariff equilibrium:

TM(I) = 0;
MAQUETTE.ITERLIM = 8000;
$INCLUDE MAQUETTE.GEN
SOLVE MAQUETTE USING MCP;

PARAMETER       SUMMARY         Results of uniform tariff;
SUMMARY("EV","Percent") = 100 * (C.L-1);
SUMMARY("Tariff","Percent") = AUXVAR.L;
SUMMARY("ER","Percent") = 100 * (PC.L/PFX.L-1);
SUMMARY("Wage","Percent") = 100 * (PL.L/PC.L-1);
SUMMARY("Wage$","Percent") = 100 * (PL.L/PFX.L-1);
SUMMARY("Return","Percent") = 100 *  (RK.L/PINV.L-1);
SUMMARY(I,"Price") = 100 * (P.L(I)/PFX.L - 1);
SUMMARY(S,"Employ") = 100 * (EMPLOY.L(S)/FD0("LABOR",S) - 1);
DISPLAY SUMMARY,
        "Tariff         Equal-yield uniform tariff rate"
        "EV             Equivalent variation in income"
        "ER             Exchange rate (Rp/$)"
        "Wage           Change real wage rate"
        "Wage$          Change in $-denominated wage rate"
        "Return         Change in Tobin's q"
        "Price          Change in $-denominated market price (%)"
        "Employ         Change in sectoral employment (%)";
```

Answer from rutherford@colorado.edu:

> In contrast, I would like to reduce tariffs in my analyses. Thus,
> the money for balancing the budget must come from somewhere else.
> In the case of a fixed exchange rate, it seems reasonable to let
> the government finance a shortfall in the budget by foreign borrowing

I think that the idea that you can assess a tariff reform with revenue replacement through foreign borrowing is utter nonsense. How can you expect this to provide any legitimate basis for comparison? If you replace revenue through a costless transfer, then any tax reduction will look good. The reason to have a model is to trade off one second-best instrument with another.

I've taken your example, added a labor leisure choice, and then computed counterfactuals based on alternative assumptions about revenue replacement. The four scenarios include "Gift" where there tariff revenue is replaced by a costless transfer from abroad, "LS_Tax" in which tariff revenue is replaced by lumpsum payments from private consumers, "WageTax" in which revenues are replaced by a uniform wage tax, and "Exemption" in which tariff revenue is replaced by a non-uniform wage tax.

The results are listed below. In my view, neither the "Gift" nor "LS_Tax" scenarios are particularly interesting. Both suggest enormous welfare gains, yet they provide little basis for concluding that tariff reform makes sense or not.

The WageTax and Exemption scenarios illustrate the importance of the revenue replacement policy. I find that when the tariff revenue is replaced by a uniform wage tax (subject to the implicit elasticity of labor supply), then welfare rises. If, however, the tariff revenue is replaced by a wage tax which applies primarily to the FOOD sector, the the tariff reform is not desireable.

Incidentally, this model illustrates how to do lump-sum equal yield. The GOVT agent spends money on goods purchased in fixed proportion. The tax instrument is then varied to achive a target level of demand.

Incidentally, I feel that you missed the point of my earlier example. I wanted to illustrate that in this model, as in many empirical models, a substantial welfare improvement can be obtained by simply moving to a uniform tariff rate. This sort of reform is desireable because it can be carried out with introducing any new taxes into the economy.

The model is at:

http://robles.Colorado.EDU/~tomruth/reform.txt

Here are illustrative results:

```
-----   1834 PARAMETER SUMMARY       Results of uniform tariff

                Gift      LS_Tax     WageTax   Exemption

EV            36.134      22.312       6.783      -2.198
ER           -17.662     -18.559     -18.578     -16.486
Wage          24.820      23.763       2.921      -4.546
Wage$          2.774       0.794     -16.200     -20.282
Return        26.384      30.042      17.403      26.619
FOODPRD      -32.674     -33.817     -33.384     -21.643
NFOODPD      -11.266     -12.064     -12.349     -15.228
FOOD          22.473      36.411       8.467     -86.316
NFOOD          5.486       8.592      -3.382      19.227
Transfer      83.351      82.221
WageTax                               22.808      21.751

Key:

EV             Equivalent variation in income
Transfer       Tariff replacement transfer (% of public expenditure)
ER             Exchange rate (Rp/$)
Wage           Change real wage rate
Wage$          Change in $-denominated wage rate
Return         Change in Tobin's q
Price          Change in $-denominated market price (%)
Employ         Change in sectoral employment (%)
```

Answer from glennwh@ix.netcom.com:

Tom is quite right here, and unusually direct in his choice of words! There is a very important policy issue in here which is coming to dominate most of the trade policy analyses that we see being done for people like the World Bank. That is, how to replace tariff revenue that might be lost due to foreign trade liberalization? Since LDCs get a huge share of their government revenues from foreign trade, relative to DCs, and tend to have extreordinarily distorted domestic tax systems, this is a central issue. It is quite easy to come up with scenarios where sensible trade policy reform from a welfare point of view is just swamped by the welfare costs of raising domestic taxes to make up the revenue.

This is one reason why Tom also pointed you towards looking at uniform tariff rates as a possible policy, since MPSGE allows

you to trivially calculate the uniform rate that generates just the same revenue as the benchmark. This provides a pure measure of the distortionary effects of the tariffs, in a much less fanciful way than using lump-sum taxes. (Our complaint with LS taxes is not that they might be pedagogically useful devices to help isolate the second-best effects of using non-LS replacement taxes, but that their use by Mere (Trade) Theorists leads us to forget that no such tax ever existed in the real world.)
In any event, these are themes that you will see that Tom and I and David Tarr have been exploring in a number of studies over the years for the Bank and others. See publications 15 and 18-23 at http://theweb.badm.sc.edu/glenn/cv.htm\#II

Answer from khak0002@tc.umn.edu:

Let me join the discussion regarding foreign borrowing during the tariff reduction and the trade-offs between the instruments.

It is true that most tax instruments are unavailable to the governments of developing countries that embark on trade liberalization (mainly because of lousy tax administration). That is why I think sovereign borrowing is the option to conduct the reform without shock therapy (gradual Ramsey type liberalization may be an alternative). Once the low tariffs are in place, cheaper imports make production more efficient, the investment grows, so do the exports, the tax base and the tax collections. Or so the government (and IMF) hopes they will.

However, there is one problem with this line of reasoning. The reform may not be sustainable. How it happens? Here is the story of the reform reversal (and the trade offs of foreign financing of the reform).

The key idea is the government's inability to commit itself to permanently low tariffs that produces multiple equilibria. In one, the reform is successful because the investors, who believe the reform will last, increase the investment. Eventually this expands the tax base for domestic taxes. Meanwhile, the negative budgetary effect of tariff removal warrants a deficit. However, the foreign bankers, the only source of financing the deficit, would not let the government borrow more than it will want to repay. Therefore, if the debt is high, the government delays the reform while reducing the debt in order to be able to reform later. When the debt is sufficiently low, the reform is immediate and successful.

For certain initial conditions there is another, Pareto inferior equilibrium. In it, fearful of the reversal, domestic investors do not restructure in response to a reform in order to avoid capital income losses. At this point, if the debt is high enough, the government can not borrow anymore and the only way to continue the reform is to cut the expenditures drastically. Instead, the government prefers to raise the tariffs, making investors' fears self-fulfilling.

Thus, for a sufficiently large initial debt, pessimistic expectations of the investors may tip the reform to a reversal. Still, the optimal government policy may be 'to reform' if the net expected benefits of the reform exceed the net expected costs of the reversal.

If anyone is interested in the details, I will email the paper. I am also interested in the critical comments both as to the validity of the model and empirical evidence supporting-contradicting the stiry.

Answer from sdevarajan@worldbank.org:

I agree with Tom and you that welfare calculations based on replacing tariff revenues with foreign borrowing (in a static model) are meaningless.

However, I am not sure I agree with you that replacing a given tariff structure with uniform tariffs provides a "pure measure" of the "distortionary effects of tariffs." The reason is that uniform tariffs may not even be second-best, for at least two reasons: the existence of other tax distortions in the system; and the modified Ramsey optimal tax rule, that would say that the optimal set of tariffs are those that are inversely proportional to the elasticity of import demand. Indeed, there are situations where going from a variegated tariff structure to a uniform one is welfare-decreasing.

I'll send you a paper I did some years ago with Sweder van Wijnbergen where we derived the optimal tariff formulae, compared them with uniform tariffs and then simulated them with a CGE model of Cameroon.

Answer from glennwh@ix.netcom.com:

You are quite right, in the sense that the Ramsey-optimal tariffs (or taxes, in general) are not going to be uniform without silly restrictions on parameter choices. More generally, and this is the important GE point you make, the "dirtortionary effect" of any instrument cannot be studied by just varying that instrument ... one needs to take the rest of the economy into account. That was the general point I was trying to stress, in terms of the relationship between reforms of foreign taxes and implied reforms of domestic taxes to ensure budget balance for the government, and we agree on that.

This is worth stressing for a broader group, since there is a lot of policy pressure to avoid extending analyses of trade reforms to include domestic tax and subsidy structures, and one simply cannot avoid doing so if an honest evaluation of the distortionary effect of the trade reform is to be obtained.

Answer from ecscj@frost.csv.warwick.ac.uk:

I have a very little experience of using general equilibrium models for policy analysis. However, I see a missing point in Pavel-Tom-Shanta-Glenn discussion on revenue-neutral impact of uniform tariff reform by using "gift", "ls-tax", "wage tax" and "exemption" as replacement alternatives.

My major concern is that development issue should have been a primary focus of a model of a developing economy such as Cameroon.

My intuition is that if we consider development as an objective of an economic policy both forreign borrowing and tariff reform might become sensible thing to do. Lower prices of consumption goods through tariff reform not only reduces project costs but also promotes income distribution. It enhances capital accumulation when borrowed funds are used high-return-projects.

Basic neoclassical principle is that scarcity of cpaital in developing economy with abundant labour resources make the rate of return to capital to be higher in those countries than in developed economies. In think it is general opinion that the foreign borrowing is bad only when wasted/corrupted away or used according to some non-economic criteria. Otherwise it will be self-financing and a good outlet for the idle capita (low return) in donar countries.

We need an "international development model" to replace "international dole model". A well formulated multi-regional Ramsey model (preferabley with an overlapping generation in it) could replace dole models that many times have been wrogly put in massive use in recommending liberalization policies in developing economies.

But the irony of development process and global relations is that it is hard to get to the point.

Answer from glennwh@ix.netcom.com:

I do not believe you read Tom's initial
response on this issue carefully. Here is what he said: I think that the idea that you can assess a tariff reform with revenue replacement through foreign borrowing is utter nonsense. How can you expect this to provide any legitimate basis for comparison? If you replace revenue through a costless transfer, then any tax reduction will look good. The reason to have a model is to trade off one second-best instrument with another.

This was said in the context of a discussion on using a small-open-economy model to undetrtake these comparisons, so that in the typical formulation the loan would be a costless gift from some arbitrary foreign agent. If you build in some constraint that the loan had to be repaid over time with (possibly subsidized) interest, as you offer by way of alternative, then it is not the costless instrument that Tom correctly had concerns with.
This is a fair point to make, since there are lots of dynamic OLG models being used for policy work that builds in inter-generational constraints. See, for example, his lovely paper on the intergenerational burden-sharing effects of carbon taxes:
http://www.gams.com/projects/dk/conf98/MobiDKolg.pdf

An earlier paper in the same tradition, by Tom and me, can be found at http://www.gams.com/projects/dk/venice.pdf

Now these models do not build in budget constraints that span time, but some neat work that does is by Tom and David:
http://www.gams.com/projects/dk/conf98/tom\&david.pdf
in the context of trade policy reforms for Chile.
I will politely ignore some of your other comments about a devlopment issue not being the primary focus of this discussion.

4.18 SAM vs. CGE Models

I'm going to work with Multi-regional models for Colombia. I want to know if you could send me some references about this topic, or if is possible to build this models with SAM or Input-Output Matrix instead of CGE Models.

Answer from rwigle@wlu.ca:

This question raises many others. First, lots of people start from a SAM when they do CGE modeling. A SAM on it's own isn't a complete model, it's really a dataset with some structure imposed on it.

You could make an I-O model, but <bold>depending on your purpose</bold>, you probably wouldn't want to.

If you DO want to do a CGE model look at GTAPinGAMS. The fixed cost of starting CGE modelling is greatly reduced if this is your starting point.

http://robles.Colorado.EDU/~tomruth/gtapingams/html/gtapgams.html

Answer from r.desantis@ifw.uni-kiel.de:

One uses a model according to his aim.

SAMs and Input-Output Matrices are not models, they are data set. In the literature, you might find models with fixed prices, which use SAMs or I-O tables often to compute multipliers; and models with endogenous prices, such as the CGE models. You decide to choose to model you need according to your policy question.

4.19 Help needed: differential tax policy analysis

I am the beginner of GAMS/MPSGE program and have been struggling with the following problem for quite a while.

I am doing differential tax policy analysis. The pre-existing taxes are payroll tax, coporate income tax and personal income tax. I want to look at the economic incidence if payroll tax is replaced with consumption tax.

I interprete payroll tax as the tax on Labor, coporate income tax as the tax on Capital, personal income tax as the tax on two commodities X and Y. By using THREEMGE.gms as the blueprint, I wrote the program(attached). But it turns out that the counterfactual result(replacing payroll tax with consumption tax) is exactly the same as benchmark result.

I read the materials that I can find from website, but still couldn't figure out what's wrong. I would really appreciate it if you could take a minute to look at my program and give me hints.

```
SETS        G           GOODS AND SECTORS /X, Y/,
            F           PRIMARY FACTORS   /K, L/,
            H           HOUSEHOLDS     /YR, OR, YP, OP/;

ALIAS (S,G);
```

```
TABLE A(G,S) MAKE MATRIX (GOODS OUTPUTS BY SECTOR)

            X           Y
X        940237708      0
Y           0           6695788998;



TABLE C(G,H)  HOUSEHOLD DEMAND

        YR             OR           YP          OP
X      379330313     377660874    64008531      119237990
Y      3267521978    2186549025   551363951     690354044;

TABLE FD(F,S)  FACTOR DEMAND BY SECTOR

            X                  Y
K          533103414          1502181238
L          354014087          4366820504;

TABLE E(F,H)  FACTOR ENDOWMENTS

        YR             OR             YP          OP
K     104821         1989334850      104821     45740160
L     5554040057              0      937105487         0;

TABLE D(F,H)  FACTOR DEMAND BY HOUSEHOLDS


       YR              OR            YP         OP
K      0                0             0          0

L      1514716611       0         255594342     0;

TABLE T(F,S)  TAX PAYMENT BY FACTOR BY SECTOR

            X           Y
K           0          171764181

L       53120207      655023075;


Parameter TC(H)   TOTAL PERSONAL INCOME TAX BY HOUSEHOLD
        /YR  392575976, OR  193355546,  YP  66243484,  OP  4445304/;

PARAMETER TRN(H) TRANSFER REVENUE

        / YR  0, OR 768210595, YP  0, OP 768297178 /;



Parameter I(H)    Personal income from factors
                  /YR 4039428267, OR 1989334850, YP 681615966, OP 45740160/;
```

```
PARAMETER ELAS(S)  ELASTICITY OF SUBSTITUTION IN PRODUCTION;

ELAS(S) = 1;

PARAMETER ESUB(H)  ELASTICITY OF SUBSTITUTION IN DEMAND;

ESUB(H) = 0.8;

SCALAR  GREV  BENCHMARK GOVERNMENT REVENUE;

GREV     = SUM(H, TRN(H));

PARAMETER TF(F,S)  FACTOR TAX RATE;

TF(F,S) = T(F,S) / FD(F,S);

Parameter TQ(g)  consumption tax from personal income tax rate;
TQ(g)=0.108;


PARAMETER PF(F,S)  BENCHMARK FACTOR PRICES GROSS OF TAX;

PF(F,S) = 1 + TF(F,S);

parameter PQ(g)  Benchmark good prices gross of consumption tax;
PQ(g)= 1 + TQ(G)

PARAMETER  THETA(G)  WEIGHTS IN NUMERAIRE PRICE INDEX;
THETA(G) = SUM(H, C(G,H));
THETA(G) = THETA(G) / SUM(S, THETA(S));

PARAMETER WBAR(H)  BENCHMARK WELFARE INDEX;
WBAR(H) = SUM(G, C(G,H)) + SUM(F, D(F,H));




$ONTEXT

$MODEL:Doug

$SECTORS:
      AL(S)

$COMMODITIES:
      P(G)  W(F)  PT

$CONSUMERS:
      RA(H) GOVT

$AUXILIARY:
      TAU
```

```
$REPORT:
        V:CD(G,H)       D:P(G)          DEMAND:RA(H)
        V:DF(F,H)       D:W(F)          DEMAND:RA(H)
        V:EMPLOY(S)     I:W("L")        PROD:AL(S)
        V:WLF(H)        W:RA(H)

$PROD:AL(S)  s:1
      O:P(G)      Q:A(G,S)    P:PQ(G)     A:GOVT   N:TAU   M:TQ(G)
      I:W(F)      Q:FD(F,S)   P:PF(F,S)   A:GOVT   T:TF(F,S)


$DEMAND:RA(H)  s:0.8  a:1
      D:P(G)      Q:C(G,H)   a:
      D:W(F)      Q:D(F,H)
      E:W(F)      Q:E(F,H)
      E:PT        Q:TRN(H)

$DEMAND:GOVT
      D:PT        Q:GREV

$Constraint:TAU
        PT =G= SUM(G, Theta(g)*P(G));

$OFFTEXT

$SYSINCLUDE mpsgeset Doug


* Benchmark Replication

TAU.L=1;

* Reinstall the benchmark tax rate:

TF(F,S)=PF(F,S) - 1;

* Verify that the benchmark equilibrium is replicated:

Doug.iterlim = 0;


*Counterfactual: replace labor tax with uniform consumption tax

TF("L", S)=0;

$INCLUDE doug.GEN
SOLVE doug USING MCP;
```

Answer from glennwh@ix.netcom.com:

You set

```
    Doug.iterlim = 0;
```

for the benchmark, and then try to solve the counterfactual without relaxing this. So the solver is not allowed to iterate because you tell it not to.

4.20 MPSGE ...

I have specified a single-country model for Finland recently and tried out various things with the MPSGE-programme. One thing would interest me beyond the use of, say, $FUNLOG namely:

If I only wanted to print out the CES-share parameters of my calibrated CGE-model, how would that be possible?

Answer from rutherford@colorado.edu:

Sorry, Ralph, but the closest you come to the share parameters is with the FUNLOG report. This reports reference prices and quantities, so it is then possible to determine the relevant share parameters. The way the function evaluator works, these shares are not stored but are instead recomputed in each evaluation. The funlog output only reports data which are stored for the function.

If you specify $datech:.true. then you get a report which reveals all the compensated demand and Slutsky Jacobian elements at each evaluation point. This might hold some interest for you.

I've included a simple model to be completely concrete about how this works.

```
$ontext

$model:cesfun

$funlog:.true.
$datech:.true.

$sectors:
        x

$commodities:
        px
        pe
        pk
        pl

$consumers:
        ra

$prod:x  s:0.8  ke:0.2
        o:px    q:1
        i:pe    q:0.05  ke:
        i:pk    q:0.60  ke:
        i:pl    q:0.35
```

```
$demand:ra
        d:px    q:1
        e:pe    q:0.05
        e:pk    q:0.60
        e:pl    q:0.35

$offtext
$sysinclude mpsgeset cesfun

option sysout=on;

cesfun.iterlim = 0;

$include cesfun.gen
solve cesfun using mcp;
```

4.21 Welfare measures

```
Why do most CGE modelers  use only changes in the  household utility
level (cost of per unit of utility) to measure the welfare changes
between a benchmark and a counterfactual solutions? Such a measure cannot
possibly be an aggregate measure of the welfare becasue it does not
assign any utility to changes in the government and investment
demands between those two solutions. Should not we also need to consider
what happens to these demands while making value judgements based on welfare figures
coming out of a model?
```

Answer from r.desantis@ifw.uni-kiel.de:

```
I believe you must consider that. To avoid this problem, most of static
AGE models assume that both gevernment consumption and total investment
do not change in the conterfactual. Regarding government consumption,
you might consider it as a public good. It would enter in the household
utility function and, as a result, in the computation of welfare.
```

4.22 Recursive dynamic modeling using MPSGE

```
At the Institute of Economic Studies in Iceland we have for some time been
working on a CGE model.  These models have been static models in the Shoven
and Whalley tradition.  Now all the models in MPSGE code that I have seen
so far have been static in the form of Single period models.  Does anyone
have simple examples in MPSGE codes of how it is possible to link single
period sub-model together over time via Capital accumulation using myopic
expectations?  The Ballard et. al model on the gams homepage only describes
the single period submodel.
```

Answer from boehringer@zew.de:

```
Find attached a simple dynamic-recursive model which Tom Rutherford
and me used for demonstration mode during a MPSGE modeling workshop
some years ago.

$TITLE  CARBON TAXES WITH EXEMPTIONS FOR ENERGY-INTENSIVE INDUSTRIES
```

```
*       GAMS-MPSGE workshop - Boehringer/Rutherford - Stuttgart, 1995

SETS    F       PRIMARY FACTORS /X,Y,E,K,L,R/
        G(F)    COMMODITIES /X,Y,E/,

        NE(G)   NON-ENERGY GOODS /X,Y/,
        MG(G)   IMPORTED COMMODITIES
        XG(G)   EXPORTED COMMODITIES
        FF(F)   FACTORS EMPLOYED IN CURRENT MOODEL;

ALIAS (S,G);

SCALAR          GAMMA           ENERGY-OTHER GOOD ELASTICITY /0.5/;

PARAMETERS      ZBAR(S)         BENCHMARK OUTPUT
                DBAR(G,S)       BENCHMARK INTERMEDIATE INPUT
                FDBAR(F,S)      BENCHMARK FACTOR INPUT
                MNET(G)         NET IMPORTS
                XNET(G)         NET EXPORTS
                MGROSS(G)       GROSS IMPORTS
                XGROSS(G)       GROSS EXPORTS
                CBAR(G)         BENCHMARK FINAL CONSUMPTION
                ENDOW(F)        FACTOR ENDOWMENTS
                T(G,S)          INPUT TAX RATE
                TC(G)           CONSUMPTION TAX
                NVK             STOCK OF NEW VINTAGE CAPITAL
                UBAR            TOTAL EXPENDITURE;

TABLE SAM(*,*)  BENCHMARK SOCIAL ACCOUNTS

           X        Y      E    EXPORT  IMPORT  DEMAND
X         20      -10             -4        2     -8
Y                 105            -20       13    -98
E         -4       -5      5                9     -5
K        -10      -40     -1                     51
L         -6      -50                            56
R                         -4                      4
FX                                24     -24       ;

*       EXTRACT THE DATA:

*       INITIALLY POINT TO FACTORS IN SAM:

FF(F) = YES;
FF(S) = NO;

ZBAR(S)    = SAM(S,S);
DBAR(G,S)  = MAX(0, -SAM(G,S));
FDBAR(FF,S) = -SAM(FF,S);
XGROSS(G)  = -SAM(G,"EXPORT");
MGROSS(G)  = SAM(G,"IMPORT");
MNET(G)    = MAX(0, MGROSS(G) - XGROSS(G));
XNET(G)    = MAX(0, XGROSS(G) - MGROSS(G));
CBAR(G)    = -SAM(G,"DEMAND");
```

```
UBAR       = (SUM(G, CBAR(G)));
ENDOW(FF)   = SAM(FF,"DEMAND");

*       MOVE IMMEDIATELY TO SECTOR-SPECIFIC CAPITAL:

FDBAR(S,S) = FDBAR("K",S);
FDBAR("K",S)  = 0;
ENDOW(S) = FDBAR(S,S);
ENDOW("K") = 0;
FF(F) = YES$ENDOW(F);

DISPLAY FDBAR, ENDOW;


*       LOAD IDENTIFIERS FOR IMPORTED GOODS, EXPORTED GOODS AND
*       ACTIVE PRIMARY FACTORS:

MG(G) = YES$MNET(G);
XG(G) = YES$XNET(G);
FF(F) = YES$ENDOW(F);

$ONTEXT

$MODEL:ENERGY

$SECTORS:
        Z(S)            ! SECTORAL OUTPUT INDEX
        INV(S)          ! INVESTMENT RATE - NEW VINTAGE
        M(G)$MG(G)      ! IMPORT INDEX
        X(G)$XG(G)      ! EXPORT INDEX
        U               ! UTILITY INDEX

$COMMODITIES:
        PFX             ! PRICE OF FOREIGN EXCHANGE
        P(G)            ! GOODS PRICE INDEX
        W(F)$FF(F)      ! FACTOR PRICE INDEX
        PNVK            ! PRICE INDEX FOR NEW VINTAGE CAPITAL
        PU              ! UTILITY PRICE

$CONSUMERS:
        RA      ! REPRESENTATIVE AGENT INCOME

$PROD:Z(S)  s:GAMMA  a:1
        O:P(S)  Q:ZBAR(S)
        I:P(G)  Q:DBAR(G,S)  A:RA  T:T(G,S)
        I:W(F)  Q:FDBAR(F,S)  a:

$PROD:INV(S)
        O:W(S)
        I:PNVK

$PROD:M(MG)
        O:P(MG) Q:MNET(MG)
        I:PFX   Q:MNET(MG)
```

```
$PROD:X(XG)
        O:PFX   Q:XNET(XG)
        I:P(XG) Q:XNET(XG)

$PROD:U  s:GAMMA  a:1
        O:PU    Q:UBAR
        I:P(G)  Q:CBAR(G)  A:RA  T:TC(G)  a:$NE(G)

$DEMAND:RA
        D:PU    Q:UBAR
        E:W(F)  Q:ENDOW(F)
        E:PNVK  Q:NVK

$OFFTEXT
$sysinclude mpsgeset energy

TC(G) = 0;
T(G,S) = 0;

NVK = 0.10 * SUM(S, ENDOW(S));
INV.L(S) = 0.1 * ENDOW(S);
ENDOW(S) = 0.9 * ENDOW(S);

ENERGY.ITERLIM = 0;
$INCLUDE energy.gen
SOLVE ENERGY USING MCP;
ABORT$(ENERGY.SOLVESTAT NE 1) "MODEL DOES NOT CALIBRATE";
ENERGY.ITERLIM = 2000;

SCALAR  SRVSHR  CAPITAL SURVIVAL SHARE,
        MPSAVE  MARGINAL PROPENSITY TO SAVE;

MPSAVE = NVK / UBAR;

SRVSHR = (SUM(S, ENDOW(S)) - NVK) / SUM(S, ENDOW(S));

SET             YR      YEARS /1990*2000/;

PARAMETER       REPORT  SUMMARY REPORT;

*       APPLY A TAX ON ENERGY USE:

T("E",S) = 0.20;
TC("E")  = 0.20;
T("E","X") = 0;

LOOP(YR,

*       DEFINE NEW CAPITAL:

        NVK = U.L * UBAR * MPSAVE;

*       DEPRECIATE EXISTING CAPITAL STOCKS:
        ENDOW(S) = SRVSHR * ENDOW(S) + INV.L(S);

$INCLUDE energy.gen
        SOLVE ENERGY USING MCP;

        REPORT(YR,"WELFARE") = 100 * (U.L - 1);
        REPORT(YR,S) = INV.L(S);
);
DISPLAY REPORT;
```

Answer from glennwh@ix.netcom.com:

```
There are many templates out there for multi-period models in MPSGE. I
would recommend, as the place to start, the paper by Lau, Rutherford and
Pahlke, available at the MobiDK Project web site
http://www.gams.com/projects/dk/mobi\_p.htm in Adobe PDF form. They list
the code in several formats (e.g., MPSGE and MCP). There are several
papers in the MobiDK Project applying these Ramsey models to policy
issues; for example, the paper by Jesper Jensen listed at
http://www.gams.com/projects/dk/conf98
```

Answer from ecscj@frost.csv.warwick.ac.uk:

```
$ontext

Here is a very simple  example of a  dynamic model in MPSGE with a
made up SAM, that I had worked out in my graduate school a couple of
years ago, with some help. I would appreciate if you have any comments or suggestions
in this model.
We can collaborate in modeling efforts, if proves helpful for any of
us. Thanks.


Suppose we have a following social accounting matrix for an economy.

Markets   X1       X2     E       M      W        Cons
---------------------------------------------------------
 p1       150           -100      50    -100
 p2                50    -25      75     -100
 pl       -100     -20                             120
 pk        -50     -30                              80
 pm1
 pm2
 pw                                        200     -200
 pfx                    125     -125
-----------------------------------------------------------
Here X1 and X2 are activities; E and M are exports and imports
W is welfare index and cons is representative consumer.

$offtext


$TITLE A SMALL TWO SECTOR DYNAMIC GENERAL EQUILIBRIUM MODEL
```

```
SET
        F  Primary factors /L, K/,
        I  Sectors    /s1, s2/;


ALIAS (I,J);

SET TP /1990*2010/, TFIRST(TP) /1990/, TLAST(TP);

TLAST(TP) = YES$(ORD(TP) EQ CARD(TP));

DISPLAY TP, TFIRST, TLAST;


TABLE ZZ(*,J) Initial data of the economy
           s1       s2
Y0       150.0      50.0
D0        50.0      70.0
I0        50.0      30.0
M0        50.0      75.0
X0       100.0      25.0
VX0      100.0      25.0
VL0      100.0      20.0
VK0       50.0      30.0
DEP0      20.0       8.0
ESUB       0.5       0.5;

PARAMETER ETRNDX(I)  / s1  8.0, s2 5.0/,
            ETRNXX(I)  / s1  10.0, s2 7.50/;
TABLE IMA(I,J) Investment  matrix
       s1    s2
s1    30.0   20.0
s2    20.0   10.0;


SCALAR    W0, INCADJ;

PARAMETER       Y0(I)
                D0(I)
                I0(I)
                INV0(J)
                M0(I)
                X0(I)
                VX0(I)
                K0(I)
                VK0(I)
                VL0(I)
                DEP0(I)
                ENDOW(F)
                EXOGFX0
                S0(I)
                PX0(I)
                PM(I)
                PM0(I)
                VM0(I)
                TX(I)
                TM(I)
                MKT(I);

Y0(I) = ZZ("Y0",I);
D0(I) = ZZ("D0",I);
I0(I) = ZZ("I0",I);
M0(I) = ZZ("M0",I);
X0(I) = ZZ("X0",I);
*K0(I) = ZZ("K0",I);
VK0(I) = ZZ("VK0",I);
VL0(I) = ZZ("VL0",I);
MKT(I) = Y0(I)+ M0(I)- D0(I)-X0(I)-I0(I);
S0(I) = Y0(I)+M0(I)-X0(I);
ENDOW("L") = SUM(I,VL0(I));
ENDOW("K") = SUM(I, VK0(I));

TX(I) = 0;
TM(I) = 0;
PM0(I) = 1+TM(I);
PX0(I) = 1+TX(I);
VX0(I) = PX0(I)*X0(I);
VM0(I) = PM0(I)*M0(I);
DISPLAY MKT, S0;

*IMA(I)=SUM(J,IMA(I,J));
*IMA(I,J) = IMA(I,J)/INV0(I);
*DISPLAY IMA;

EXOGFX0 = SUM(I,VM0(I)-VX0(I));
DISPLAY EXOGFX0;

SCALAR G      POTENTIAL GROWTH RATE /0.02/
       R       REFERENCE INTEREST RATE
       DEPR   DEPRECIATION RATE /0.07/
       ESUBT ELASTICITY OF SUBSTITUTION
                        /0.99 /;


PARAMETER       QREF(TP)        STEADY STATE QUANTITY INDEX
                PREF(TP)        PRESENT VALUE PRICE
                QLABOR(TP)      GROWTH RATE OF LABOR
                NR(I)           IMPLIED NET RATE OF INTEREST
                RK0(I)          BASEYEAR GROSS COST OF CAPITAL;


*computation of benchmark interest rate

NR(I)=(VK0(I)*(G+DEPR)-DEPR*SUM(J,IMA(I,J)))
/(SUM(J,IMA(I,J))+VK0(I)*(G+DEPR)-DEPR*SUM(J,IMA(I,J)));
RK0(I) = (NR(I)/(1-NR(I)) + DEPR);
K0(I) = VK0(I)/RK0(I);
I0(I) = K0(I)*(G+DEPR);
```

```
DISPLAY  NR, RK0, K0, I0, Y0, D0, I0, X0, M0, VK0, VL0;

PARAMETER INVRATIO;
INVRATIO(I) = SUM(J,IMA(J,I))/VK0(I);
PARAMETER IKRATIO;
IKRATIO(I) = SUM(J, IMA(I,J))/K0(I);

* WHEN INVRATIO(I) = 1  THEN THE STEADY STATE BENCHMARK RATE OF
INTEREST IS:

R = G/(1+G);

DISPLAY INVRATIO, IKRATIO, R;

* IF I0(I) = VK0(I), THEN WE NEED TO COMPUTE DIFFERENT INTEREST RATE.

*Steady state growth path quantity and price indices:
        QREF(TP) =(1+G)**(ORD(TP)-1);
        PREF(TP) =(1-R)**(ORD(TP) -1);
        QLABOR(TP) =QREF(TP);

DISPLAY QREF, PREF;


$ONTEXT
$MODEL:SMALL3

*Declaration of variables in the model. There are three blocks
* production block, demand block and agent block.

$SECTORS:
        Y(I,TP)      ! Production index
        K(I,TP)      ! Capital stock
        INV(I,TP)    ! Investment
        X(I,TP)      ! Export index
        A(I,TP)     ! Armington supply index
        U(TP)       ! Itra-period utility index
        W           ! Welfare index over period

$COMMODITIES:
        P(I,TP)     ! Price index
        PD(I,TP)    ! price index
        PX(I,TP)    ! Export price index
        PL(TP)      ! wage index
        RK(I,TP)    ! Rental price of capital
        PK(I,TP)    ! Index of rental rate
        PTK(I)      ! Terminal investment premium
        PVPFX       ! Real exchange rate index
        PU(TP)      ! Price of consumption each period
        PW          ! Welfare price index

$CONSUMERS:
        RA      ! Representative agent
```

```
$AUXILIARY:
        TK(I)     ! Terminal capital demand

* Production function is of a Cobb-Douglas type

$PROD:Y(I,TP)     t:ETRNDX(I) VA:1
        O:PD(I,TP)  Q:(Y0(I)-VX0(I))
        O:PX(I,TP)  Q:VX0(I)
        I:PL(TP)    Q:VL0(I) VA:
        I:RK(I,TP)  Q:K0(I) P:RK0(I) VA:

$PROD:K(I,TP) s:0
        O:RK(I,TP)         Q:K0(I)
        O:PK(I,TP+1)       Q:(K0(I)*(1-DEPR))
        O:PTK(I)$TLAST(TP) Q:(K0(I)*(1-DEPR))
        I:PK(I,TP)         Q:K0(I)

$PROD:INV(I,TP)   s:0
        O:PK(I,TP+1)        Q:(SUM(J,IMA(J,I)))
        O:PTK(I)$TLAST(TP)  Q:(SUM(J,IMA(J,I)))
        I:P(J,TP)           Q:IMA(J,I)

$PROD:X(I,TP)   t:ETRNXX(I)
        O:PVPFX     Q:(X0(I)*PREF(TP)) P:(PX0(I)/PREF(TP)) A:RA
        T:TX(I) I:PX(I,TP)  Q:X0(I)

$PROD:A(I,TP)       s:0
        O:P(I,TP)   Q:S0(I)
        I:PD(I,TP)  Q:(Y0(I)-X0(I))
        I:PVPFX     Q:(M0(I)*PREF(TP)) P:(PM0(I)/PREF(TP)) A:RA
        T:TM(I)

$PROD:U(TP) s:1
        O:PU(TP)   Q:(SUM(I,D0(I)))
        I:P(I,TP)  Q:D0(I)

$PROD:W  s:ESUBT
        O:PW        Q:W0
        I:PU(TP)    Q:(QREF(TP)*SUM(I,D0(I))) P:PREF(TP)

$DEMAND:RA
        E:PL(TP)           Q:(QLABOR(TP)*ENDOW("L"))
        E:PK(I,TFIRST)     Q:K0(I)
        E:PW               Q:INCADJ
        E:PTK(I)           Q:(-K0(I)) R:TK(I)
        E:PVPFX            Q:(EXOGFX0 +
        SUM((I,TP),(TX(I)*X0(I)*PREF(TP)+ TM(I)*M0(I)*PREF(TP)))) D:PW
                      Q:W0

$CONSTRAINT:TK(I)
        PTK(I)*I0(I) =E= SUM((J,TLAST),P(J,TLAST)*IMA(I,J));

$OFFTEXT
$SYSINCLUDE mpsgeset SMALL3
```

```
        P.L(I,TP)  = PREF(TP);
        PD.L(I,TP) = PREF(TP);
        PL.L(TP)   = PREF(TP);
        PX.L(I,TP)  = PREF(TP);
        PK.L(I,TP)  = PREF(TP)/(1-R);
        RK.L(I,TP)  = RK0(I)*PREF(TP);
        PU.L(TP) = PREF(TP);

        Y.L(I,TP)  = QREF(TP);
        X.L(I,TP)  = QREF(TP);
        INV.L(I,TP)= QREF(TP);
        K.L(I,TP)  = QREF(TP);
        A.L(I,TP)  = QREF(TP);
        U.L(TP) = QREF(TP);

W0 = SUM(TP,PREF(TP)*QREF(TP)*SUM(I, D0(I)));
PTK.L(I)=SUM(TLAST, PK.L(I,TLAST)*(1-R));
TK.L(I) =(1+G)*SUM(TLAST, K.L(I,TLAST));
DISPLAY W0, PTK.L, TK.L;

INCADJ = PW.L*W0
        +SUM(I, PTK.L(I)*K0(I)*TK.L(I))
        -SUM(I,TX(I)*(SUM(TP, X.L(I,TP)*PX.L(I,TP)*PVPFX.L)))
        -SUM(I,TM(I)*VM0(I)*PVPFX.L)
        -SUM(TP, QREF(TP)*PL.L(TP)*ENDOW("L"))
        -SUM((I,TFIRST),(PK.L(I,TFIRST)*K0(I)));
DISPLAY INCADJ;

*SMALL.WORKSPACE =10;
SMALL3.OPTFILE=1;

SMALL3.ITERLIM =0;
$INCLUDE SMALL3.GEN
SOLVE SMALL3 USING MCP;



PARAMETER REPORT, GDP;
         REPORT(I,TP, "Y")=Y.L(I,TP);
          GDP(tp,i) =y.l(i,tp);
  GDP(tp,i) =inv.l(i,tp);
DISPLAY REPORT;

$LIBINCLUDE GNUPLOT GDP
*$LIBINCLUDE QADPLOT REPORT TP
*Do not compute counter factual experiment until you have calibrated
the model $exit

*Counter factual experiment

TM("s2") = 0.05;
$INCLUDE SMALL3.GEN
SOLVE SMALL3 USING MCP;
```

```
TM("s2") = 0.1;
$INCLUDE SMALL3.GEN
SOLVE SMALL3 USING MCP;
```

4.23 Papers on pension system reforms

Does anyone know of MPSGE papers / templates which deal with the reform of pension systems?

Answer from ecscj@frost.csv.warwick.ac.uk:

Rutherford, Bihrubger and Palkhe had a paper in the dynamic CGE conference in Copenhagen during last summer that uses an overlapping generation model for analyzing intergenerational inequality. I do not know whether their code is public. There are some templates in http://robles.Colorado.EDU/~tomruth/olg/exchange/index.html

The NBER's aging group have several papers on the topic that I find very useful. Carlo Perroni (Warwick University) might have a few templates on pension decision in GAMS.

It is very active area and a big topic; there must be plenty of other places where one can find papers related to pension system.

Answer from glennwh@ix.netcom.com:

This is a very active area for exciting research. A couple of papers I would point to were presented at a conference this year in Denmark. Most of the papers are available by web link in PDF form at http://www.gams.com/projects/dk/conf98

I would particularly recommend the paper by Morten Lau, which endogenizes the retirment decision in a very clean model: http://www.gams.com/projects/dk/conf98/human.pdf

But there are several other papers linked there that should also be reviewed. Ronald Wendner, in Austria, has also done some nice work linking pension reform with green tax revenues. I've just lost his e-mail, but you can probably track it down (he is in Graz I think).

4.24 Implementing a quota in MPSGE

I am a Ph-D student in France, and I'm working on building a CGE model to study the use of water as a production factor, especially in agriculture. Thank's to a GAMS short course held in Germany with Chris Boehringer, I have now completed a simple multiregional SOE model in MPSGE. I would like to use this model to implement a quota on water use : in some regions, the total disposable quantity of water for production should be limited so that it equals "quota(R)*WEndow(R)", where WEndow(R) is the water endowment quantity of region R in the benchmark data and quota(R) the quota in region R.

I think this can be done by rationing, that is using an auxiliary variable, a constraint definition and a R: field but I'm not sure of how to do so. My question is : how does rationing precisely work in MPSGE ? I found some material by Tom Rutherford on the GAMS web site, and tried something (see my model below) but I am not sure to do things properly. Could anyone give me a hint, please ?

```
$TITLE a MRT SOE Model with 15 regions - Armington assumption

$ontext
Author  : Laurent Piet
          Agricultural and rural area dynamics Research Unit
          Cemagref, Bordeaux Regional Center - France
Contact : laurent.piet@cemagref.fr
(c) Cemagref, October 1998
$offtext

SCALAR BREAK /0.00001/;

SETS
        R       regions         /Z1*Z15/
        G       all goods       /AGR, IRR, IND, IAA, SER, K, L, W, T/
        F(G)    factors         /K, L, W, T/
        S(G)    sectors         /AGR, IRR, IND, IAA, SER/
        H       households      /ARC/
;

ALIAS (S,SS), (R,RR);

**-------- BENCHMARK

** Benchmark SAMs

$INCLUDE bmsam15z.dat

** Benchmark parameters

PARAMETERS
        QKBM(R,S)         Capital benchmark consumption of sector S
        QLBM(R,S)         Labor benchmark consumption of sector S
        QWBM(R,S)         Water benchmark consumption of sector S
        QTBM(R,S)         Land benchmark consumption of sector S
        QIBM(R,SS,S)      Intermediate input bechmark level of sector SS in
sector S
        QYBM(R,S)         Benchmark domestic production level in sector S
        QMBM(R,S)         Benchmark aggregate import of sector S
        QXBM(R,S)         Benchmark aggregate export of sector S
        QIMBM(R,RR,S)     Benchmark bilateral import
        QIXBM(R,RR,S)     Benchmark bilateral export
        QOMBM(R,S)        Benchmark import from RoW
        QOXBM(R,S)        Benchmark export to RoW
        QCBM(R,S)         Welfare index consumption of sector S
        DKBM(R)           Benchmark dotation in capital
        DLBM(R)           Benchmark dotation in labor
        DWBM(R)           Benchmark dotation in water
        DTBM(R)            Benchmark dotation in land
        QRBM(R)            Benchmark income of representative agent
;

**-------- GE MODEL

** Elasticities of substitution

PARAMETERS
klsub(S)        Capital Labor aggregate nest
vasub(S)        Value Added nest
;

klsub("AGR") = 1;
klsub("IRR") = 1;
klsub("IND") = 1.5;
klsub("IND") = 1.5;
klsub("SER") = 2.5;
vasub(S) = 0.5;

** Key parameters for simulation

PARAMETERS
quota(R)        quota on water use /1/;

** Model definition with MPSGE

$ONTEXT
$MODEL:MRTSOE15z

$SECTORS:
        Y(R,S)          ! Domestic production level of sector S
        A(R,S)          ! Armington aggregate on imports
        W(R)            ! Water as a production factor
        WLF(R)          ! Aggregate welfare index

$COMMODITIES:
        PY(R,S)         ! Domestic production price of S
        PA(R,S)         ! Price of Armington aggregate
        PFX             ! World price index
        PC(R)           ! Consumption price
        PL              ! Labor wage
        PK              ! Capital rental price
        PW              ! Water aggregate price
        PWAT            ! Water rental price
        PT              ! Land rental price

$CONSUMERS:
        ARC(R)          ! Income of representative agent

$AUXILIARY:
        WQUOTA(R)       ! Water quota

** Water as a factor production
```

```
$PROD:W(R)
        O:PWAT          Q:(SUM(S,QWBM(R,S)))
        I:PW            Q:DWBM(R)

** Domestic production

$PROD:Y(R,S)    s:0  VA:vasub(S)  KL(VA):klsub(S)  WT(VA):0
        O:PY(R,S)       Q:QYBM(R,S)
        I:PK            Q:QKBM(R,S)     KL:
        I:PL            Q:QLBM(R,S)     KL:
        I:PWAT          Q:QWBM(R,S)     WT:
        I:PT            Q:QTBM(R,S)     WT:
        I:PA(R,SS)      Q:QIBM(R,SS,S)

** Armington aggregation

$PROD:A(R,S)    t:8  s:4  a:8
        O:PFX           Q:QOXBM(R,S)
        O:PY(R,S)       Q:(SUM(RR,QIXBM(R,RR,S)))
        O:PA(R,S)       Q:(QCBM(R,S)+SUM(SS,QIBM(R,S,SS)))
        I:PY(R,S)       Q:QYBM(R,S)               a:
        I:PY(RR,S)      Q:QIMBM(R,RR,S)           a:
        I:PFX           Q:QOMBM(R,S)

** Aggregate welfare index

$PROD:WLF(R)    s:1
        O:PC(R)         Q:QRBM(R)
        I:PA(R,S)       Q:QCBM(R,S)

** Representative Agent income definition

$DEMAND:ARC(R)
        D:PC(R)         Q:QRBM(R)
        E:PK            Q:DKBM(R)
        E:PL            Q:DLBM(R)
        E:PW            Q:DWBM(R)       R:WQUOTA(R)
        E:PT            Q:DTBM(R)

** Water quota constraint

$CONSTRAINT:WQUOTA(R)
        WQUOTA(R) =E= quota(R)*W(R);

$OFFTEXT
$SYSINCLUDE mpsgeset MRTSOE15z

PFX.FX = 1;
WQUOTA.l(R) = 1;

**-------- BENCHMARK REPLICATION

MRTSOE15z.ITERLIM = 0;
```

```
$INCLUDE MRTSOE15z.GEN
SOLVE MRTSOE15z USING MCP;
MRTSOE15z.ITERLIM = 10000;
ABORT$(MRTSOE15z.OBJVAL GT BREAK) "WRONGLY CALIBRATED!";

**-------- COUNTERFACTUAL STUDIES

quota("Z1") = 0.7;
$INCLUDE MRTSOE15z.GEN
SOLVE MRTSOE15z USING MCP;
```

Answer from rutherford@colorado.edu:

When a veterinarian in Colorado hears that there is a four legged creature
with hoofs and a mane waiting for his attention in the waiting room, should he
presume that it is a horse or a zebra?

Similarly, if you set out to build an Arrow-Debreu equilibrium model
and need to incorporate water rationing, should you presume that this
requires a market price or an auxiliary variable with some complicated
constraint?

In the neoclassical economic model it is prices which ratio demand.
Best to stay away from the fancy stuff until you are certain that you
have the conventional market equilibrium model well under control.

See http://robles.Colorado.EDU/~tomruth/water_maquette/water.html for an
illustrative model.