

A GAMS Bug

The Variables and Marginals that Will Not Go Away

Bruce A. McCarl

Specialist in Applied Optimization
Professor of Agricultural Economics, Texas A&M
Principal, McCarl and Associates

mccarl@tamu.edu
mccarl@bihs.net
agrinet.tamu.edu/mccarl

979-693-5694
979-845-1706

The Variables and Marginals that Will Not Go Away

When solving a series of models , I have found cases where a variables and equation marginals that are supposed to be eliminated by conditionals hang around in the solution. There are two variants to this problem which are intertwined with the SOLVEOPT option.

SOLVEOPT controls whether the results of a subsequent solve replace the results of a previous solve or are merged with them. The default setting is SOLVEOPT= merge (I don't know why but you have to live with it). I always use SOLVEOPT= replace.

So what is the problem. Suppose we solve a model with land transfers allowed and then another without them. If the merge option is on the solution variable values from the land transfer variables from before are retained. The solution then looks like it still has land transfers. Similarly the information on equations that are supposed to be eliminated are still present.

Consider the example on the next page

The Variables and Marginals that Will Not Go Away

Note in here I have a model where some of the variables and equations can be removed if the data in the parameter limit and limit2 is nonzero. (Sorry for the tight typing but space is limited) – (wontleav.gms)

```
Set varname variable names /x1,x2,x3/
Parameter limit(varname) controls existence of items /x1 1, x2 1, x3 1/
      limit2(varname) controls exist of items /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;
*when limit(varname) is zero variable variablval(varname) should be gone
obj.. z=e=sum(varname,variablval(varname)$limit(varname)
*when limit2(varname)is zero variable secondvar(varname) should be gone
      +secondvar(varname)$limit2(varname));
*when limit(varname)is zero equation bound(varname) should be gone
bound(varname)$limit(varname).. variablval(varname)=l=limit(varname);
*when limit2(varname)is zero equation bound2(varname) should be gone
bound2(varname)$limit2(varname).. secondvar(varname)=l=limit2(varname);
model try /all/;      option lp=bdmlp;
parameter sol(*,*,*) reort writing array;
*solution case labeled trybefore
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);
*after doing this all cases of secondvar and bound2 should be gone
limit2(varname)=0;
*after doing this variabval(x1) should be gone
limit('x1')=0;
*solution case labeled tryaft
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);
option solveopt=replace
*now try a replace solution case labeled tryaftrep
solve try using lp maximizing z;
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);
*out of frustration zap secondval solution case labeled tryaftzap
secondvar.l(varname)=0;
solve try using lp maximizing z;
sol("z","z","tryaftzap")=z.l;
sol("variablval",varname,"tryaftzap")=variablval.l(varname);
sol("secondvar",varname,"tryaftzap")=secondvar.l(varname);
sol("margbound",varname,"tryaftzap")=bound.m(varname);
sol("margbound2",varname,"tryaftzap")=bound2.m(varname);
display sol;
```

Here I am trying to get rid of X1 cases. The resultant report writing is on the next page

The Variables and Marginals that Will Not Go Away

Now what happened in our solution.

```
----- 63 PARAMETER SOL
          trybefore          tryaft          tryaftrep          tryaftzap
z          .z          6.000          2.000          2.000          2.000
variablval.x1          1.000          1.000          gone
variablval.x2          1.000          1.000          1.000          1.000
variablval.x3          1.000          1.000          1.000          1.000
secondvar .x1          1.000          1.000          1.000          gone
secondvar .x2          1.000          1.000          1.000          gone
secondvar .x3          1.000          1.000          1.000          gone
margbound .x1          1.000          1.000          gone
margbound .x2          1.000          1.000          1.000          1.000
margbound .x3          1.000          1.000          1.000          1.000
margbound2.x1          1.000          1.000          gone
margbound2.x2          1.000          1.000          gone
margbound2.x3          1.000          1.000          gone
```

In first case (**trybefore**) all is ok. We have values for x1 in variableval, secondvar plus marginals for bound, bound2.

In our second case (**tryaft**) X1 in variableval, and bound should be gone as should levels for secondvar and marginals for bound2 **but they are not**. Why? GAMS retains them from the previous solution.

In our third case (**tryaftrep**) we use **solveopt=replace** to overwrite the old solution information not merge with it but still find the levels for **secondvar are still around**. Why? GAMS retains information for totally eliminated variables (all instances of secondvar were eliminated).

In our fourth case all is ok and the report corresponds to the model solution **but only after we manually zeroed that variable using `secondvar.l(varname)=0;`**

The Variables and Marginals that Will Not Go Away

So what should you do

1. When you are in a position where you
 - A. are **solving models in a sequence** or otherwise **specifying initial values** and or equations (through a basis or a starting point) before a solve
 - B. have some **variables or equations you eliminate conditionally** depending on model data
 - C. **Need accurate levels of the variables in the report writing** calculations (try some time explaining to the policy client why you without land transfers solution exhibits nonzero land transfers)

Then you should

- A. use **option solveopt=replace**;
 - B. **zero any variable levels** where **all cases for that variable are possibly eliminated**
2. Hope that GAMS some day fixes this

p.s. Apparently complex situations also occur when holdfixed is used and variables are removed