10/17/03
11:51

Themes from GAMS in Computational Economics

by

David A. Kendrick[1]

On the occasion of Alexander Meeraus's 60th birthday it is appropriate to consider again some of the themes in computational economics that Alex has created or promoted via the GAMS software.[2] For many years I have consciously and unconsciously passed along to my students pieces of Alex's wisdom without ever cataloging these ideas. This paper is an opportunity to review some of those ideas and the contributions they have made and are making to the development of computational economics.

1. Set Driven Languages

From his work writing Fortran code to generate linear programming input files for economic models developed at the World Bank in the 1960's and 70's Alex realized that many of the models had variables which were indexed over sectors, plants, markets, regions, dams, seasons, years, etc. So, as he began developing GAMS to automate the model development process, he included sets to contain the indices as a major part of the language. This had the wonderful effect that large and cumbersome models with many plants, markets, commodities and time periods could be specified in only a few pages in the GAMS language. It also made the models much easier to debug and to communicate to others.

Suddenly the task of learning someone else's model changed from an arduous task of studying reams of poorly documented Fortran code to examining the set specifications and equations in a few lines of GAMS statements.

---

[1] This paper has been prepared for presentation at a conference on GAMS and computational methods in September of 2003 in Washington, D.C.

[2] See Brooke, Kendrick, Meeraus and Raman (1998) and McCarl (2003).

The set-driven nature of GAMS gave it another advantage as well. I teach my students that when they are asked to examine an unfamiliar model that has been developed in GAMS, the first thing they should do is to study the set specification section. Either the presence or the absence of certain sets will tell them much about the model in a short period of time. For example, if the model is a growth model and there is not a set for sectors they know immediately that it is a one-sector model and perhaps not very interesting for their purposes. Of if it is a global warming model a quick examination of the time period set reveals how long a time horizon is covered and therefore something about the length of time lags in the model. Also, a global warming model without a set of regions will most likely not consider the conflict between developed and developing countries on this issue.

The ability to look quickly through a number of GAMS models and judge their suitability or interest for one's own work led to a second theme from Alex.

2. Architects and Blue Prints

Alex was fond of arguing that economists needed to be more like architects. He said that architects rarely if ever designed a new building from scratch. Rather they went to the file cabinet and pulled out the blueprints for a previous building that was similar to the new one. Then they copied and modified those blueprints to convert them to the desired design for the new building. So, in Alex's view, economists should go to their computers and pull up the file for a previous model of dams in a river valley, a group of fertilizer plants or a general equilibrium description of an economy and then transform that model to develop what they wanted.

This idea took form in a library of input files for GAMS models that he called GAMSLIB. This library contained about one hundred models in its early manifestations and now, as I understand it, contains several hundred models. These models ranged from the classics such as the Chakravarty (1962) growth model to the latest world wide model of the aluminum industry – some were the product of a single scholar's endeavors for a few months and others were the result of work by teams of economists, engineers and computer scientists over several years.

Like others of the themes discussed here, this one had it antecedents. One antecedent for model libraries that I remember is the advice of my mentor, Paul Samuelson, who counseled his students to study the works of the greats and learn to stand on their shoulders. GAMSLIB is a wonderful contribution in that tradition and has eased the way for many a student and researcher as they labored how to fashion their own models.

But of course Alex would not allow just any model to be admitted to the hallowed halls of GAMSLIB. Rather it had to have style …

3. The Elements of Style

Who would have thought that the spacing of the GAMS representation of a model on a sheet of paper mattered so much? So long as the compiler would digest the model without complaint, why was it necessary to be so careful about matters of style?

Alex believed that the order of the components in a model, the spacing on the page of the elements of the model, the layout of a table, the careful and informative use of acronyms, etc. were all a crucial part of the development of a model. He insisted that this made it easier – indeed in some cases, possible - for the author to check and debug the model and for others to understand it, criticize it or emulate it. Alex was so persuasive on the subject that it led me to write a paper on style in economic modeling, viz, Kendrick (1984a).

One of the elements of style is the order of the components of the model. This gave birth to the idea of generic models.

4. Generic Models

In the early GAMS models we intermingled declaration and definition of the elements of the model. Thus in the first few lines of the model we both declared and defined the sets as well as the tables and parameters before we entered the variables and equations, viz.

```
Sets
     I    Plants     / Sicartsa, Fundidora, HYLSA /
     J    Markets    / MexicoDF, Monterrey, Guadala /  …
Equations  …
Model  …
Solve  …
```

In some later models we did the declarations at the top for the sets, tables and parameters but did not defined those elements until after the equations had been entered, viz

```
Sets
     I    Plants
     J    Markets  …
Equations …
Sets
     I     / Sicartsa, Fundidora, HYLSA /
     J    / MexicoDF, Monterrey, Guadala / …
Model …
Solve …
```

This had the effect of creating a generic model at the top of the GAMS input file that could be used for any industry in any country. Then in the bottom part of the GAMS statement the model was specialized to the particular industry and country by defining the sets, tables and parameters.

5. Separation of the Solvers from the Model Representation

Another of Alex's idea was to separate the solvers from the representation of the model. Priori to this most models were integrated with the solver. Thus one would develop a nonlinear programming model in Fortran and include both the data input and the code for the algorithm in the program. The model representation and data on the one hand and algorithmic code on the other hand were inseparable.

In contrast, Alex included in GAMS an option that permitted the user to specify that a model should be solved with different algorithms and code. For example to solve a nonlinear programming model with the Murtagh and Saunders (1987) MINOS5 code one could use

```
OPTION  NLP = MINOS5 ;
```

or to solve the same model with the Drud (1985) CONOPT code one could use

```
OPTION  NLP = CONOPT ;
```

Since some models are solved more easily with one code than with another this offered a substantial improvement. Also, it provided a modularity that permitted developers of solvers to concentrate on the algorithms and not to be concerned with the model representation.

6. Math to GAMS or GAMS to Math?

In teaching computational economics I have encouraged my students to first think through and develop their models in mathematics and then translate the math into a GAMS representation. The mathematical statement has a precision and parsimony that I believe helps the students to think clearly about their models. In fact, I believe in this so strongly that at one point I even used the Unix LEX and YACC systems to develop my own compiler which would allow student to input their models in NROFF and see the results in mathematics and also have the mathematics translated into a GAMS representation of their model, Kendrick (1984b).

When I talked with Alex about this he said that it was interesting idea but was backwards. I should instead encourage my students to develop their models in GAMS and then translate those models into mathematics for publication in the economics journals.

I have not followed this part of Alex’s advice. Rather over the years I have consistently had my students begin with the mathematical representation of models and then translate those models into GAMS. However, I must confess that if one of my students develops a substantial model in GAMS that they usually begin to do all of their model development work in the GAMS rather than in mathematics. So maybe Alex also has this one right and what we need is a GAMS to mathematics translation utility.

7. Conclusions

The era of the 1960's, 70's and 80's was an exciting time for developments in computational economics. Alan Manne had pioneered the development of computational sectoral models and had even tackled the problem of how to include economies of scale in those models, viz. Manne (1963) (1967). Hollis Chenery, who had also been an early contributor to this literature[3], had become the Vice President for Research at the World Bank. He had chosen Ardy Stoutjesdijk to head a department that was given responsibility, among other things, for the development of computational models of problems in economic development and Ardy had encouraged Alex in his work to develop the prototypes of GAMS. Also, Ardy had made arrangements with Johns Hopkins University Press to publish a series of monographs on this subject – a series that was to include, among others, books by Choksi, Meeraus and Stoutjesdijk (1980) on the fertilizer industry, by Kendrick, Meeraus and Alatorre (1984) on the steel industry and on multicountry investment analysis by Mennes and Stoutjesdijk (1985).

Jan Bisschop had joined with Alex[4] on some of the work in the early stages of the development of GAMS. Sherman Robinson was working with Kemal Dervis and Jaime DeMelo[5] in a different part of the World Bank on the development of computational general equilibrium models – a class that was to later include models in GAMS for many countries. Also, Arne Drud (1985) was a developing a large-scale nonlinear programming system in the generalized reduced gradient class that would become one of the solvers in GAMS.

---

[3] See Chenery (1952).

[4] See Bisschop and Meeraus (1982).

[5] See Dervis, de Melo and Robinson (1982).

Alex and GAMS were in many ways the hub of these developments. Alex was always engaged in helping one group or another to fashion ways to efficiently develop and solve their models and, at the same time, he was constantly adding to the capabilities of GAMS.

So hats off to Alex for many innovations and for embedding them in GAMS where they have contributed substantially to the development of computational economics in the last thirty years and will continue to do so for many years to come.

References

Bisschop, Jan and Alexander Meeraus (1982), "On the Development of a General Algebraic Modeling System in a Strategic Planning Environment", *Mathematical Programming Study*, Vol. 20, pp. 1-29.

Brooke, Anthony, David A. Kendrick, Alexander Meeraus and Ramesh Raman (1998), *GAMS: A Users Guide*, available at the GAMS Development Corporation web site at http://www.gams.com

Chakravarty, S. (1962), "Optimal Savings with a Finite Planning Horizon," *International Economic Review*, September, Vol. 3, pp. 338-355.

Chenery, Hollis (1952), Over-Capacity and the Acceleration Principle", *Econometrica*, Vol. 20, (January), pp. 1-28.

Choksi, Armeane M., Alexander Meeraus and Ardy J. Stoutjesdijk (1980), *The Planning of Investment Programs in the Fertilizer Industry*, Johns Hopkins University Press, Baltimore.

Dervis, Kemal, J. de Melo and Sherman Robinson (1982), *General Equilibrium Models for Development Policy*, Cambridge University Press, New York.

Drud, Arne (1985), "CONOPT – A GRG Code for Large Sparse Dynamic Nonlinear Optimization Problems," *Mathematical Programming*, Vol. 31, p. 153 ff.

Kendrick, David A., Alexander Meeraus and Jaime Alatorre (1984), *The Planning of Investment Programs in the Steel Industry*, Johns Hopkins University Press, Baltimore.

Kendrick, David A. (1984a), "Style in Multisectoral Modeling", in A. J. Hughes Hallet, (ed), *Applied Decision Analysis and Economic Behavior*, Martinus Nijhoff Publishers, Dordrecht, pp. 329-360.

Kendrick, David A. (1984b), "A Mathematical-Computer Language for Linear Programming Problems", in T. Basar and L. F. Pau (eds), *Dynamic Modelling and Control of National Economies*, Pergamon Press Ltd., Oxford, United Kingdom, pp. 233-240.

Manne, Alan S. (1963), *Scheduling of Petroleum Refinery Operations*, Harvard University Press, Cambridge, MA.

Manne, Alan S. (1967), *Investment for Capacity Expansion: Size, Location and Time Phasing,* M.I.T. Press, Cambridge, MA.

McCarl, Bruce A. (2003), *McCarl's GAMS User Guide 2003*, available at http://www.gams.com/dd/docs/bigdocs/gams2002.

Mennes, Loet B. M. and Ardy J. Stoutjesdijk (1985), *Multicountry Investment Analysis*, Johns Hopkins University Press, Baltimore.

Murtagh, Bruce A. and Michael A. Saunders (1987), "MINOS 5.1 User's Guide," Report SOL 83-20R, December 1983, revised January 1987, Stanford University.