PRELIMINARY RESULTS IN IMPLEMENTING A MODEL OF THE WORLD ECONOMY ON THE CYBER 205: A CASE OF LARGE SPARSE NONSYMMETRIC LINEAR EQUATIONS

DANIEL B. SZYLD

INSTITUTE FOR ECONOMIC ANALYSIS
NEW YORK UNIVERSITY

NEW YORK, NEW YORK

Preliminary Results in Implementing a Model of the World Economy on the Cyber 205: A Case of Large Sparse Nonsymmetric Linear Equations

Abstract

Daniel B. Szyld
Institute for Economic Analysis
New York University

A brief description of the Model of the World Economy implemented at the Institute for Economic Analysis is presented, together with our experience in converting the software to vector code.

For each time period, the model is reduced to a linear system of over 2000 variables. The matrix of coefficients has a bordered block diagonal structure, and we show how some of the matrix operations can be carried out on all diagonal blocks at once.

We present some other details of the algorithms and report running times.

1. Description of the Model

The first input-output model of the world economy was originally developed for the United Nations by Leontief, Carter and Petri [1977] as a tool for evaluating alternative long-term economic policies. The most recent version that has been implemented spans the period 1970-2030 in 10-year intervals. The model is dynamic in the sense that the solution for each 10-year period requires information obtained from the solution for the previous period. In this paper we focus on the solution of a single time period.

In the current version of the model, the world is divided into 16 regions (r=16) and for each of the regions the detailed economic activities are described by a set of linear algebraic equations of the form

$$A_i \underline{y}_i + S_i \underline{w} = 0 \quad (i = 1,\ldots,r). \tag{1}$$

The components of the vectors $\underline{y}_i$ correspond to levels of domestic production, imports, and exports of goods and services, and so on, for each region, and $\underline{w}$ is the vector of total world exports. In addition there are global constraints described by the equation

$$\sum_{i=1}^{r} G_i \underline{y}_i = 0 \quad , \tag{2}$$

which imposes the consistency among regional trade relations.

A more detailed description of the model can be found in Leontief, Carter and Petri [1977], Duchin and Szyld [1979], and Szyld [1981].

All the matrices involved are very sparse. For example

A_i could be 200 × 250 with 2500 nonzeros.

S_i could be 200 × 50 with 50 nonzeros.

G_i could be 50 × 250 with 100 nonzeros.

Each matrix A_i has more columns than rows and therefore some components of $\underline{y}_i$ have to be prescribed.

If $\underline{x}_i$ are the vectors of unknown components of $\underline{y}_i$ and M_i and E_i are the corresponding submatrices of A_i and G_i, the whole model for a single time period can be regarded as a linear system of equations of over 3000 variables with a nonsymmetric bordered block diagonal matrix of coefficients of the form:

$$\begin{vmatrix} M_1 & & & & S_1 \\ & M_2 & & & S_2 \\ & & \ddots & & \vdots \\ & & & M_r & S_r \\ E_1 & E_2 & \dots & E_r & O \end{vmatrix} \begin{vmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \vdots \\ \underline{x}_r \\ \underline{w} \end{vmatrix} = \begin{vmatrix} \underline{b}_1 \\ \underline{b}_2 \\ \vdots \\ \underline{b}_r \\ \underline{O} \end{vmatrix} \qquad (3)$$

where the blank blocks in the matrix are zero blocks.

When the model was first implemented, the program for the solution of (3) inverted the matrices M_i and stored the inverses. The approximate computer time to perform this task was 4 hours on a PDP-11. The (dense) inverses were saved for subsequent runs during which they were updated depending on the components of $\underline{y}_i$ prescribed and on changes in the matrices A_i. Each of these subsequent runs required 110 seconds on an IBM 370 for each time period.

The set of prescribed components of $\underline{y}_i$ and the matrices are used to determine a scenario, i.e., a set of economic assumptions. Studies carried out with the World Model compare results of different scenarios, i.e., the implications of the different assumptions. The consequences of the introduction of new technologies, different development strategies, or shifts in trade patterns are among the numerous scenarios that can be analyzed. Thus, the World Model is a flexible tool to analyze alternative policies. Several large scale empirical studies have been carried out with this model. The most recent ones are reported in Leontief and Duchin [1983], Leontief and Sohn [1982], Leontief, Koo, Nasar and Sohn [1983] and Leontief, Mariscal and Sohn [1982].

To make this tool much more flexible we needed to greatly reduce the computational resources required to run a scenario. A first step in that direction was the application of sparse matrix techniques for the solution of (3). In the present implementation the matrices A_i are stored using a sparse scheme, i.e., only the nonzero elements are stored, together with some integer arrays indicating their locations. A single array of approximate length 3200 contains all vectors $\underline{x}_i$, $i=1,...r$. Other such arrays contain the vectors $\underline{b}_i$, the nonzero values of the matrices S_i and G_i, or other data objects. Similarly, objects like the nonzeros of the matrices M_i appear in single arrays of length close to 5000.

2. Method of Solution

The algorithmic details of the solution of (3) are given in Duchin and Szyld [1979], Szyld [1981], and Furlong and Szyld [1982]. Here we enumerate the operations for the solution of (3) very schematically.

```
loop 1.  For $i=1,\ldots,r$
     1.1.  Read $A_i, G_i, S_i$, and the prescribed elements of $\underline{y}_i$
     1.2.  Produce $M_i, E_i$ and $\underline{b}_i$
     1.3.  Obtain factorization of $M_i$

loop 2.  For $i=1,\ldots,r$
     2.1.  Prepare different right hand sides with columns of $S_i$
     2.2.  Solve systems with matrix $M_i$

loop 3.  Obtain $\underline{w}$

loop 4.  For $i=1,\ldots,r$
     4.1.  Compute $\underline{b}_i - S_i\underline{w}$
     4.2.  Solve $M_i\underline{x}_i = \underline{b}_i - S_i\underline{w}$
```

The factorization of the matrices Mi (in step 1.3) and the solution of several linear systems with them (in steps 2.2 and 4.2) are performed with routines from the MA28 set developed by Duff [1977].

We report the running times for a single time period with this method of solution without any vector code in Table 1.

Table 1.

System/compiler options	CPU sec.
IBM 370/168	~38
IBM 3033	~20
Cyber 205, no options	11.46
Cyber 205, vectorization by the compiler	9.04

Architectural features combined with the sparse matrix techniques resulted in running times three to ten times faster than the 110 seconds that subsequent runs required after computation of the inverses in the first implementation of the World Model. The goal is now to obtain vector code for the Cyber 205 that will further reduce the overall running time.

3. Code vectorization

The redesign of the World Model software for its efficient use on the Cyber 205 was conceived in three phases:

I. Elementary operations over all regions

II. The MA28 package inner loops

III. New concepts for MA28

Phase I consists essentially of the vectorization of all operations except those associated with the factoring of the matrices M_i and solutions of the corresponding linear systems. Those operations correspond ~~to~~ steps 1.2, 2.1, and 4.1. Each of these steps has a different structure but they all are loops operating on vectors of length about 200, inside another loop of length 16. The basic idea was to split the outer loop and perform simultaneously the operations on all vectors of the different regions, i.e., on vectors of length of about 3200. Cyber 205 FORTRAN commands such as scatter, gather and bit operations were used throughout.

We illustrate the vectorization of step 4.1. The length of $\underline{w}$ is about 50. S_i is a rectangular matrix of about 200 rows, with only one nonzero entry per column. It is stored as a vector with an accompanying integer array indicating in which row each nonzero entry lies. The following FORTRAN statements are part of sequential code for step 4.1.

```
          DO 100 II=1,NREG
          IBEG=(II-1)*NTRADE
          IBEGB=IPNTB(II)-1
          DO 50 I=1,NTRADE
             INDEX=KTRDBG(IBEG+I)+IBEGB
             B(INDEX)=B(INDEX)-EXPSH(I+IBEG)*W(I)
     50     CONTINUE
    100 CONTINUE
```

The running time for these loops was 1008 μsec. Different vectorization options were analyzed. One of them consisted of scattering the vectors that contain the nonzero values of S_i and $\underline{w}$ to vectors of length of about 3200 and then performing the triad operation. This required 9514 clock cycles, or about 190 μsec. The version adopted performs the multiplication of the vectors containing the nonzeros of S_i and $\underline{w}$ first, a vector operation of length about 800, scatters that vector and performs the final subtraction in 7250 clock cycles or 145 μsec, a gain of a factor of 7 from the sequential code.

Similar gains have been achieved in the other portions of the code vectorized in phase I. Unfortunately only a small portion of the total running time of the World Model is spent in the code vectorized in phase I. Thus the overall gain was relatively small.

About 30% of the total running time of the World Model is spent on routines of the MA28 package in which the matrices M_i are factored (step 1.3), and solutions with many right hand sides computed (steps 2.2 and 4.2). At the present time we have completed only part of phase II, the vectorization of some of the inner loops in the MA28 set.

Due to the startup time in any vector operation, it is common practice to look into the length of the vectors involved in the operation to decide if the vectorization is really worthwhile. In codes for sparse matrices, the vector length for an operation is usually the number of nonzero elements in a particular row or column, and thus varies within the code. The technique used in this case is to assess if the vector length is above a particular value and branch the process of that particular row or column to vector or sequential code. The running time of the code incorporating these features is 7.33 CPU seconds, cf. Table 1.

Phase III, not yet implemented, consists of reconceptualizing the MA28 set. We will investigate the possibility of solving several right hand sides simultaneously, as well as other features like special treatment of right hand sides with few nonzero elements.

Acknowledgment. I would like to thank Valdimir Roytman for his assistance throughout the project and Faye Duchin for useful comments on an early draft of the manuscript.

References

Faye Duchin and Daniel B. Szyld, Application of Sparse Matrix Techniques to Inter-Regional Input-Output Analysis, Economics of Planning 15(1979) 142-167.

Iain S. Duff, MA28- a set of FORTRAN Subroutines for Sparse Unsymmetric Systems of Linear Equations, Report R.8730, A.E.R.E. Harwell, HMSO, London, 1977.

Kenneth E. Furlong and Daniel B. Szyld, World Model Solution Programs, Release 1, Institute for Economic Analysis, New York University, July 1982.

Wassily Leontief, Anne P. Carter and Peter A. Petri, The Future of the World Economy, Oxford University Press, 1977.

Wassily Leontief and Faye Duchin, Military Spending: Facts and Figures, Worldwide Implications and Future Outlook, Oxford University Press, New York, 1983.

Wassily Leontief, James Koo, Sylvia Nasar and Ira Sohn, The Future of Non-fuel Minerals in the U.S. and World Economy: Input-Output Projections from 1980-20000, Lexington-Heath, Lexington, Mass., 1983.

Wassily Leontief, Jorge Mariscal and Ira Sohn, The Prospects for the Soviet Economy to the Year 2000, Journal of Policy Modeling 5(1983) 1-18.

Wassily Leontief and Ira Sohn, Economic Growth, in Just Faaland (editor), Population and the World Economy in the 21st Century, Basil Blackwell, Oxford, 1982, pp. 98-127.

Daniel B. Szyld, Using Sparse Matrix Techniques to Solve a Model of the World Economy, in Iain S. Duff (editor), Sparse Matrices and their Uses, Academic Press, London, 1981, pp. 357-365.