HANDS-ON COMPUTING TO INTRODUCE GEMPACK AND GTAP

Ken Pearson
Impact Project, Centre of Policy Studies, Monash University and
Mathematics Department, La Trobe University
11 October, 1995

In this document we give several examples of hands-on computing you can carry out to familiarise yourself with the GEMPACK software, and also various aspects of the GTAP model. In Part A, the Examples are directed at finding out about the data in a GTAP data base. In Part B, the Examples are directed at carrying out simulations. In Part C, the Examples are directed towards using GTAPVOL.TAB to calculate the volume changes (as distinct from percentage changes) produced by a simulation. In Part D, the Examples are directed at calculating equilibrium elasticities for GTAP (see chapter 5 of the GTAP book). Part E contains tips for using the programs. In Part F, we mention hands-on computing you may like to try with models other than GTAP.

This document is designed to be read in conjunction with chapter 6 "Implementing GTAP Using the GEMPACK Software" in the book *Global Trade Analysis: Modeling and Applications*, edited by Thomas W. Hertel - see Hertel (1996) in the References. [Below we refer to this book as "**the GTAP Book**".] In particular this document assumes that you have obtained the software (the **GTAP Book Version of GEMPACK**) and put it into a directory \GTAPBOOK on your computer, following the instructions in section II of that chapter of the book. As stated in that chapter, the programs, data and files can only be used on a

DOS 80386/80486/Pentium PC with a hard disk,
at least 8 Mb of RAM (memory), a numeric coprocessor,
and version 3.3 or later of DOS.

Once you have installed the software and files on your computer, you can work through the Examples here. In doing so, you will familiarise yourself with the software and, perhaps more importantly, the GTAP model. The instructions in this document are quite detailed. In particular, commands that you need to type are shown on separate lines in bold-faced text.

In this document, we assume that you have already extracted all the files from the archives CMF2-01.ZIP and AGG2-01.ZIP, as described in section II of chapter 6 of the GTAP Book. If at any stage you inadvertently delete some of these files, you can get them back by extracting them again from the relevant archive. For example, if you delete the file GTAPCHK.TAB from archive CMF2-01.ZIP, you can restore it via the command

.\pkunzip cmf2-01 gtapchk.tab

In many of the Examples here, we give specific instructions about using the DOS text editor **EDIT** to examine or edit text files. This editor is only available with versions 5.0 or later of DOS. If you have an earlier version of DOS, you will need to use a different editor. Note the comments in section II of chapter 6 of the GTAP Book about editing text files.

I am grateful to Martina Brockmeier and Padma Swaminathan for helpful feedback on an earlier version of this document.

A. USING THE SOFTWARE TO FIND OUT ABOUT THE DATA

The Examples here focus mainly on the 3x3 data set. Examples 1-4 show you how you can see the data in a GTAP data set directly. Examples 5-8 involve GTAPCHK.TAB which is used to report many useful pieces of information from a GTAP data set. Examples 9-11 require you to complete a TABLO Input file which computes other information from a GTAP data set. In Examples 12 and 13 you will calculate the sizes of various distortions (for example, import tariffs).

1. Examining the Data Directly

First change directory into your directory \GTAPBOOK (which contains the 3x3 GTAP data set) by entering the command

cd \gtapbook

There are three data files associated with each GTAP data set. For the 3x3 data, the files are

SET2-01.HAR	Set information, giving region and commodity names in aggregation
DAT2-01.HAR	The Global data set (I/O for each region, trade data etc)
PAR2-01.DAT	Parameter values

Example 1 - Using SEEHAR to view the set information

The set information tells which regions and commodities are represented in the current aggregation of the data. This set information file is given the (logical) name **GTAPSETS** in the TABLO Input file **GTAP94.TAB** which lays down the theory of GTAP. To see what information is on this set information file, type

edit gtap94.tab

(or something equivalent if you do not have version 5.0 or later of DOS). Then search for GTAPSETS. (To search in **edit**, touch the **Alt** key then touch the **s** (search) key, then touch **f** (find) and type in the word(s) you wish to search for, and finally type a carriage return. Use this technique to search for GTAPSETS.)

You will see that
(a) the names of the regions (the SET is called **REG**) are held at header "**H1**",
(b) the names of the tradeable commodities (the SET is called **TRAD_COMM**) are held at header "**H2**".
(c) there are several other SETs defined.
To exit from GTAP94.TAB, touch the **Alt** key, then the **f** (file) and **x** (exit) keys, in turn.

The set information file is a so-called **Header Array file**. Each array of data on the file has a 4-character **header**. For example, the array of 3 character strings at header "H1 " contains the names of the 3 regions.
To see the information on the file SET2-01.HAR, you need to run the GEMPACK program SEEHAR. To do this, type in

seehar

When the program starts to run, give the responses below. (The first and fifth are carriage returns. After each response, we have given a comment, which starts with an exclamation mark '!' and is printed in italics; when running the program, you should not type in the exclamation mark or the comment following it.)

Input to SEEHAR

```
<carriage-return>    ! Use the default options and finish the option selection
set2-01.har          ! Name of the file containing the set information
set2-01.see          ! Name of output file
6                    ! Number of decimal places
<carriage-return>    ! (As demanded by SEEHAR)
r                    ! Output the remaining (i.e, all) arrays
```

End of Input to SEEHAR

When this has finished you can look at the set information (which is now on the file SET2-01.SEE) via the command

edit set2-01.see

You can move around the file using the **PgDn** and **PgUp** keys, for example.
To find the names of the regions in this 3x3 data set, search for **H1** which we know to be the header where this information is to be found. (First go to the top of the file by typing **Ctrl**+**Home**, that is by holding down the **Ctrl** key and touching the **Home** key. Then search for **H1**, which you can do by touching the **Alt** key, then the **s** key, then the **f** key, then typing in **H1** and a carriage return. You can search for subsequent occurrences by hitting the **F3** function key near the top of your keyboard.) What are the names of the regions?

Repeat to find out the names of the 3 tradeable commodities; search for **H2**.
To exit from SET2-01.SEE, touch the **Alt** key, then the **f** (file) and **x** (exit) keys, in turn.

Example 2 - Using SEEHAR to view the global data

This global data file contains large amounts of data including the input-output data for each region and trade data.

This file is given the (logical) name **GTAPDATA** in the TABLO Input file **GTAP94.TAB** which lays down the theory of GTAP. To see what information is on this global data file, type

edit gtap94.tab

Then search for GTAPDATA. (Use **Alt**, then **s**, then **f**. Then use **F3** to repeat.) You will see that various pieces of data are read from this file.

One of these is the array **VIPA(i,r)** whose values are held at header "VIPA". To see this and to find out what data is stored at this header and associated with VIPA(i,r) in GTAP94.TAB, move to the start of GTAP94.TAB (use **Ctrl**+**Home**). Then search for VIPA. You will see that VIPA(i,r) is declared as a **COEFFICIENT** and that it holds the Value of Private household expenditure on Imported commodity **i** in region **r**, valued at Agents' prices. By searching again for VIPA you will see that this data is read from file GTAPDATA at header "VIPA". Then exit from GTAP94.TAB (using **Alt**, then **f**, then **x**).

To see the VIPA data in the 3x3 aggregation, you need to run the program SEEHAR, which you do by entering

seehar

Then give appropriate responses (following Example 1 above, but specifying DAT2-01.HAR as the Header Array file and DAT2-01.SEE as the output file).

When this run has finished, to look at the VIPA data, enter the command

edit dat2-01.see

and search for VIPA. The rows of the 3x3 matrix **VIPA(i,r)** correspond to the imported commodities (the first index **i** is an imported commodity) while the columns correspond to regions **r,** the second index in VIPA(i,r). What is the Value of Private household expenditure in EU (region number 2) on Food (commodity number 1), valued at Agents' prices?

To exit from DAT2-01.SEE, touch the **Alt** key, then the **f** (file) and **x** (exit) keys, in turn.

The Header Array file DAT2-01.HAR is a binary file, which means that you cannot edit or access the data in it directly. This is why you had to use the program SEEHAR to convert this data to a text file DAT2-01.SEE which you can view in an editor, or print.

Example 3 - Viewing the parameter values

Although the global data and set information are held on Header Array files, the parameters data for any GTAP data set are held on a text file. (This is to facilitate editing by those wishing to conduct sensitivity analysis.) You can see the parameter values by editing the file directly (or printing it). To see the actual parameter values used with the 3x3 data set, enter the command

edit par2-01.dat

For example, search in PAR2-01.DAT for ESUBVA (using the **Alt**, **s** and **f** keys). You will see that there are 4 values (one for each commodity in PROD_COMM). Look at these values, and then exit from PAR2-01.DAT (using the **Alt**, **f** and **x** keys). To find the significance of these values, you need to look in GTAP94.TAB (via the command "edit gtap94.tab"). Search there for ESUBVA and you will see that these four values are the elasticities of substitution between capital, labor and possibly land in the production of value added for each such commodity.

Text files can be edited directly (as above), and can also be typed. For example, enter the command

type par2-01.dat

But binary files (and hence Header Array files) cannot be edited directly or typed, as you will see if you enter the command

type dat2-01.har

All you will get is a lot of unusual characters on the screen and a few beeps.

Example 4 (optional) - Using SEEHAR for other GTAP data sets

The data files used in the MFA Liberalization application in chapter 10 of the GTAP Book are all in your \GTAPBOOK directory. They are a 10x10 aggregation of the GTAP data called aggregation number 6 (see section V of chapter 6 of the GTAP Book). First you may have to extract these files from AGG2-06.ZIP via the command

.\pkunzip -o agg2-06

The names of these files are

DAT2-06.HAR	Global data
SET2-06.HAR	Set information
PAR2-06.DAT	Parameter values

First run SEEHAR to examine the set information file. Your responses should be as in Example 1 above except that the second and third responses should be

set2-06.har
set2-06.see

Edit the resulting file SET2-06.SEE.

Then follow the method in Examples 2 and 3 to examine the global data and the parameter values for this aggregation.

2. Using GTAPCHK.TAB to report various aspects of the data

GTAPCHK.TAB contains instructions to read certain parts of the global data and to compute various useful pieces of information from it. To see this, enter

edit gtapchk.tab

and then search for the first 2 or 3 occurrences of GTAPDATA. You will see that this part of GTAPCHK.TAB looks rather similar to the corresponding part of GTAP94.TAB which you looked at earlier. You are going to look at the **GOVEXP(r)** and **CHKMKTCLTRD(i,r)** values in Example 6 below. Search for GOVEXP and CHKMKTCLTRD so that you know what information these COEFFICIENTs will hold. Then exit from GTAPCHK.TAB.

There are two steps to go through before you can see the actual GOVEXP and CHKMKTCLTRD values. These steps are in Examples 5 and 6 below.

Example 5 - Implementing GTAPCHK

Start the program TABLO running by entering the command

tablo

When the program starts to run, give the responses below. (The first and third are carriage returns. After each response, we have given a comment, which starts with an exclamation mark '!'; when running the program, you should not type in the exclamation mark or the comment following it.)

Input to TABLO

```
<carriage-return>    ! Use the default options and finish the option selection
gtapchk              ! Name of the TABLO Input file
<carriage-return>    ! Default Information file name (GTAPCHK.INF)
!     (TABLO will take a minute or two to check that the
!      formulas etc in GTAPCHK.TAB contain no errors. When this is
!      finished, a menu will appear asking what you want to do next.)
!
a                    ! Begin automatic code generation
pgs                  ! Prepare output for GEMSIM
<carriage-return>    ! Use the other default code generation options
<carriage-return>    ! Default GEMSIM Auxiliary file name (GTAPCHK.GSS)
```

End of Input to TABLO

When this is finished, you can check that the GEMSIM Auxiliary Statement and Table files GTAPCHK.GSS and GTAPCHK.GST have been created via the command

dir gtapchk.gs*

The Information file GTAPCHK.INF has also been created. You can examine it by entering

edit gtapchk.inf

Move around it using the PgDn and PgUp keys, for example. It should indicate that the TABLO Input file GTAPCHK.TAB contains no errors and there are no semantic problems. To see this, search for "errors". Then exit from GTAPCHK.INF.

Example 6 - GTAPCHK with the 3x3 data

Start the program GEMSIM running via the command

gemsim

When prompted by the program, enter the responses below.

Input to GEMSIM

```
<carriage-return>    ! Use the default options and finish the option selection
gtapchk              ! Name of the GEMSIM Auxiliary files (from Example 5)
chk2-01.dis          ! Name of display file to be created
dat2-01.har          ! Name of GTAPDATA file for the 3x3 data
set2-01.har          ! Name of GTAPSETS file for the 3x3 data
```

End of Input to GEMSIM

This should create the display file CHK2-01.DIS, which you can view by entering

edit chk2-01.dis

Find the GOVEXP and CHKMKTCLTRD results by searching for these names. To refresh you memory as to what data is in GOVEXP and CHKMKTCLTRD, you may need to look at the TABLO Input file GTAPCHK.TAB again, which you can so by touching the **Alt** key, then touching **f** (file), then **o** (open) and then typing in the name GTAP94.TAB. Once you have found what you need from GTAP94.TAB and you can reopen the file CHK2-01.DIS (again via **Alt**, **f**, **o**).

How much government expenditure does the data base show in USA?

As you can see from the comments in GTAPCHK.TAB, the CHKMKTCLTRD(i,r) values are expected to be zero. (This is one check that the data base is balanced.) To check this, look at the values of CHKMKTCLTRD in CHK2-01.DIS. You will see that they are not all zero. For example, although CHKMKTCLTRD("food",'USA") is quite small (approximately -0.0039), it is not exactly zero. The reason for this is related to the accuracy with which arithmetic is done on your DOS machine. When data is stored, or arithmetic calculations are done on a PC (using GEMPACK and most other software), only the first 6 or 7 figures are accurate. Now

CHKMKTCLTRD values are obtained by taking VOM values and subtracting other values (as you can see from the FORMULA for CHKMKTCLTRD(i,r) in GTAPCHK.TAB). But (as you can see from the VOM values as displayed in CHK2-01.DIS), VOM("food",'USA") is about 663,934.0 whereas CHKMKTCLTRD("food","USA") is about -0.0039. The 7th figure in VOM is the 0 after the decimal point. Since only 6 or 7 figures are accurate in VOM and in the values subtracted from it to give CHKMKTCLTRD values, you should not be surprised to find CHKMKTCLTRD("food","USA") nonzero in the 2nd or 3rd decimal place (the 8th or 9th figures of VOM("food","USA")). Hence the -0.0039 value of CHKMKTCLTRD("food","USA") can be considered zero to machine accuracy.

Finally, exit from CHK2-01.DIS.

Example 7 - Running GEMSIM from a GEMPACK Command file

An alternative way of running GEMSIM is to take all inputs from a so-called **Command file**. Such a file for carrying out the run of GEMSIM in Example 6 above is the file CHK2-01.CMF. First look at the contents of this file via

edit chk2-01.cmf

The statements in this should be self-explanatory. Then use this file to run GEMSIM by first starting the program running via

gemsim

and then responding (when prompted)

cmf
chk2-01.cmf

You can see GEMSIM reading this file CHK2-01.CMF. It will run to produce the same results as in Example 6 above.

When (see, for example, Examples 16 and 20 in Part B below) you use GEMSIM to carry out a simulation, you will find that these Command files are the simplest and most effective way of communicating the rather large amount of information required. You will look at Command files in more detail there.

Example 8 (optional) - GTAPCHK with other data

Here the idea is to use GTAPCHK to report the same things as above, but this time about the 10x10 aggregation used in the MFA Liberalization application (this is aggregation number 6 - see Example 4 above) instead of the 3x3 aggregation. To do this,

(i) copy CHK2-01.CMF to CHK2-06.CMF via the command

copy chk2-01.cmf chk2-06.cmf

(ii) edit CHK2-06.CMF to make sure that GEMSIM reads SET2-06.HAR and DAT2-06.HAR, and that GEMSIM produces an output display file called CHK2-06.DIS and a LOG file CHK2-06.LOG.
[If you are uncertain about your file CHK2-06.CMF, you can compare it with the file CHK206OK.CMF we have prepared and which should be in your \GTAPBOOK directory.]

(iii) Finally run GEMSIM and give the responses

cmf
chk2-06.cmf

3. Building a TABLO Input file to report other aspects of the data

Example 9 - Completing TABLO Input file GTAPCHKX.TAB

Write a TABLO Input file GTAPCHKX.TAB containing

(i) READ statements to read in selected parts of the GTAP data base,
(ii) FORMULAs to calculate
(a) total domestic sales of each tradeable commodity in each region, valued at market prices,
(b) the share of domestic production of each commodity in each region used by private households,
(c) the total value of sales of tradeables in each region, valued at market prices.

An incomplete version CHKX.TAB of such a TABLO Input file is in your \GTAPBOOK directory. A printed copy of this incomplete file is on the next page.

```
!_____________________Incomplete GTAPCHKX.TAB FILE_____________________!
!-----------------------------------------------------------------------!
!                            FILEs, SETS and SUBSET                     !
!-----------------------------------------------------------------------!
FILE GTAPSETS # File with set specification #;

SET REG # Regions in the model #
  MAXIMUM SIZE 10  READ ELEMENTS FROM FILE gtapsets HEADER "H1";
SET TRAD_COMM # TRADED COMMODITIES #
  MAXIMUM SIZE 10  READ ELEMENTS FROM FILE gtapsets HEADER "H2";
SET PROD_COMM # PRODUCED COMMODITIES #
  MAXIMUM SIZE 11 READ ELEMENTS FROM FILE gtapsets HEADER "H5";
SUBSET TRAD_COMM IS SUBSET OF PROD_COMM ;

FILE  GTAPDATA # The file containing all base data. # ;
!-----------------------------------------------------------------------!
!   base revenues and expenditures at agent's prices                    !
!-----------------------------------------------------------------------!
COEFFICIENT (all,i,TRAD_COMM)(all,j,PROD_COMM)(all,r,REG)
              VDFM(i,j,r)
    ! purchases of domestic i r for use in j in region r ! ;
COEFFICIENT (all,i,TRAD_COMM)(all,r,REG)    VDPM(i,r)
    ! private household expenditure on domestic i in r ! ;
COEFFICIENT (all,i,TRAD_COMM)(all,r,REG)    VDGM(i,r)
    ! government household expenditure on domestic i in r ! ;

!-----------------------------------------------------------------------!
!      Reading basedata.                                                !
!-----------------------------------------------------------------------!
READ (all,i,TRAD_COMM)(all,j,PROD_COMM)(all,r,REG)  VDFM(i,j,r)
              FROM FILE GTAPDATA HEADER "VDFM" ;
READ (all,i,TRAD_COMM)(all,r,REG)       VDPM(i,r)
              FROM FILE GTAPDATA HEADER "VDPM" ;
READ (all,i,TRAD_COMM)(all,r,REG)       VDGM(i,r)
              FROM FILE GTAPDATA HEADER "VDGM" ;

!-----------------------------------------------------------------------!
!      DERIVATIVES OF THE BASE DATA                                     !
!-----------------------------------------------------------------------!
 COEFFICIENT (all,i,TRAD_COM)(all,r,REG)    VDM(i,r)
   ! domestic sales of commodity i in region r
      valued at market prices (tradeables only)  ! ;

FORMULA (all,i,TRAD_COMM)(all,r,REG)
         VDM(i,r) = VDPM(i,r) + ????  + sum(j,PROD_COMM, VDFM(i,j,r)) ;

COEFFICIENT (all,i,TRAD_COMM)(all,r,REG)    SHRDPM(i,r)
! the share of domestic production used by private hhlds ! ;

FORMULA (all,i,TRAD_COMM)(all,r,REG) SHRDPM(i,r) = ????/VDM(????) ;

COEFFICIENT (all,r,REG)      TOTDVM(r)
! Total domestic sales of tradeables in region r valued at market prices ! ;

FORMULA (all,r,REG)    TOTDVM(r) = ???? ;

DISPLAY VDM ;   DISPLAY SHRDPM ;  DISPLAY TOTDVM ;
```

Suggested steps for Example 9

(1) Change directory into \GTAPBOOK, copy CHKX.TAB to GTAPCHKX.TAB and then edit the file GTAPCHKX.TAB to complete it. Use the commands

cd \gtapbook
copy chkx.tab gtapchkx.tab
edit gtapchkx.tab

and then make appropriate additions and/or changes to complete the file. Before you exit from the editor, you will want to save the changes you have made: you can do this by touching the **Alt** key then the **f** (file) and **s** (save) keys in turn. Then exit from the editor (using **Alt**, then **f**, then **x**).

(2) Run TABLO to process this file. (Give responses much as in Example 5 above, but replace GTAPCHK there by GTAPCHKX.) Look in the Information file GTAPCHKX.INF to find any errors. (Edit GTAPCHKX.INF and search for two percent signs %% as each error is marked with %%.) If you find a **"syntax error" OR a "semantic problem"**, you must make the required change in GTAPCHKX.TAB (not GTAPCHKX.INF).
Warning. We have made a deliberate error in the TABLO Input file CHKX.TAB which will cause several consequential errors. Once you identify this error and eliminate it, the consequential errors will also be eliminated. [**Hint:** If you have trouble finding our deliberate error, look closely at the spelling of the set TRAD_COMM in the declaration of the COEFFICIENT VDM(i,r), where the first error is. You will see that the SET has been declared with two M's at the end of its name but we have only put one M when using this set in the declaration of VDM. Once you add the second M, this and the consequential errors will disappear.]

(3) When the TABLO Input file is correct and you have prepared GEMSIM Auxiliary files, run GEMSIM interactively (much as in Example 6, though be careful to match your responses to the prompts) to produce the display file CHKX2-01.DIS. When running GEMSIM, attach the 3x3 data files SET2-01.HAR and DAT2-01.HAR, being careful to match file names to prompts.
Note. If you are uncertain about your TABLO Input file GTAPCHKX.TAB, you can compare it with the file CHKXOK.TAB which we have prepared and which should be in your \GTAPBOOK directory.

(4) Look at the display file CHKX2-01.DIS in your text editor to see the required values. Make sure that you have all three lots of results (a),(b),(c) as requested.

Examples 10 and 11 (optional)

Example 10. Complete the incomplete Command file GTAPCHKX.CMF below. You can then produce the same results based on the 3x3 data set by running GEMSIM and responding

cmf
gtapchkx.cmf

```
! Incomplete Command file GTAPCHKX.CMF ------------------
auxiliary files = gtapchkx ;
file gtapsets = set2-01.har ;
file gtapdata = ?? ;
display file =
! End of incomplete Command file GTAPCHKX.CMF ------------
```

[If you are uncertain about your file GTAPCHKX.CMF, you can compare it with the file CHKXOK.CMF we have prepared.]

Example 11. Rerun GEMSIM to produce the same results for the 10x10 aggregation whose data files are SET2-06.HAR and DAT2-06.HAR. (You do not need to rerun TABLO.) Produce an output display file called CHKX2-06.DIS.

4. Finding the Size of Existing Distortions

In Example 12 below, you will write a TABLO Input file SHOCKSX.TAB which can calculate some of the distortions in a GTAP data set and tell you what shocks to give to the model to remove these distortions. First we give some background about the two sorts of distortions we consider in Example 12.

4.1 Background and Notation

OUTPUT SUBSIDIES/TAXES

VOA Value of output at agents' prices - i.e., what is received by the firm
VOM Value of output at market prices - i.e., the value inclusive of subsidies/taxes

TO_L Power of the intervention

Formula: **TO_L = VOA/VOM**

If **VOA > VOM** **(**i.e. **TO_L > 1) t**here is a **subsidy.**
If **VOA < VOM** **(**i.e. **TO_L < 1)** there is a **tax.**
If **VOA = VOM** (i.e. **TO_L = 1**) there is **no distortion**.

Example
VOA = 120 VOM = 100 TO_L = 1.2 (subsidy)
To remove the subsidy, must decrease TO_L from 1.2 to 1
Percent change required to make TO_L equal to 1 is (-0.2/1.2)*100 = - 16.67%
[**Aside:** Although this is a 20% distortion, the shock to remove it is **not** 20%.]
In the model GTAP94.TAB, the VARIABLE '**to**' is the percentage change in **TO_L**.
Hence we must shock the linear variable '**to**' by
{-0.2/1.2}*100 percent.
This is the **TO_HAT** value, namely
TO_HAT = {-(TO_L -1)/TO_L} * 100

IMPORT TARIFFS/SUBSIDIES

VIMS Imports values at domestic market prices - i.e., inclusive of any tariff/subsidy
VIWS Imports values at world prices (cif)

TMS_L Power of the intervention

Formula (**you complete it)**: **TMS_L =**

If **VIMS > VIWS** (i.e. **TMS_L > 1**) there is an **import tariff**.
If **VIMS < VIWS** (i.e. **TMS_L < 1**) there is an **import subsidy**.
If **VIMS = VIWS** (i.e. **TMS_L = 1**) there is **no distortion**.

Example
VIMS = 120 VIWS = 100 TMS_L = 1.2 (tariff)
To remove the tariff, must decrease TMS_L from 1.2 to 1
Percent change required to make TMS_L equal to 1 is (-0.2/1.2)*100 = -16.67%
In the model GTAP94.TAB, the VARIABLE '**tms**' is the percentage change in **TMS_L**.
Hence we must shock the linear variable '**tms**' by
{-0.2/1.2}*100 percent.
This is the desired value of **TMS_HAT**, namely (**you fill it in**)
TMS_HAT =

4.2 A TABLO Input file to report distortions and changes to eliminate them

Example 12 - Completing TABLO Input file SHOCKSX.TAB

Write a TABLO Input file SHOCKSX.TAB containing

(i) READ statements to read in selected parts of the GTAP data base,
(ii) FORMULAs to calculate
(a) existing output distortions (that is, TO_L values),
(b) shocks to percent-change variable '**to**' to eliminate these distortions,
(c) existing import distortions (that is, TMS_L values),
(d) shocks to percent-change variable '**tms**' to eliminate these distortions,
(iii) DISPLAY statements to show the TO_L and TMS_L values,
(iv) WRITE statements to write the shocks in (ii)(b),(d) above to text files.

An incomplete version SHKX.TAB of such a TABLO Input file is in your \GTAPBOOK directory. A printed copy of this incomplete file is on the next page.

```
!_________________  Incomplete SHOCKSX.TAB file  _________________!

!  FILEs, COEFFICIENTs, READs as required (see file on disk)    !
!  FORMULAs for VOA,VOM as required (see the file on your disk) !
!  There are no errors in this part of the file.                !

!-------------------------------------------------------------!
! Calculation of Powers of distortions and                    !
!         percent changes to remove distortions               !
!-------------------------------------------------------------!

ZERODIVIDE DEFAULT 1 ;

COEFFICIENT  (all,i,NSAV_COMM)(all,r,REG)  TO_L(i,r)
! Calculate tax rates in the benchmark data !   ;

FORMULA (all,i,NSAV_COMM)(all,r,REG)
      TO_L(i,r) = VO?(i,r)/VO?(i,r) ;

COEFFICIENT  (all,i,NSAV_COMM)(all,r,REG)  TO_HAT(i,r)
! Calculate percentage rates for liberalization shocks !  ;

FORMULA  (all,i,NSAV_COMM)(all,r,REG)
     TO_HAT(i,r) = {[1-TO_L(i,r)] / TO_L(i,r)} * 100;

COEFFICIENT  (all,i,TRAD_COMM)(all,r,REG)(all,s,REG)  TMS_L(i,r,s)
! Calculate tax rates in the benchmark data !   ;

 FORMULA (all,i,TRAD_COMM)(all,r,REG)(all,s,REG)
      TMS_L(i,r,s) = ???? ;

COEFFICIENT  (all,i,TRAD_COMM)(all,r,REG)(all,s,REG)  TMS_HAT(i,r,s)
! Calculate percentage rates for liberalization shocks !  ;

FORMULA  (all,i,TRAD_COMM)(all,r,???)(all,s,REG)
     TMS_HAT(i,r,s) = ???? ;

FILE (NEW,TEXT) TOHAT # The file with shocks to obtain TO = 1 #;
FILE (NEW,TEXT) TMSHAT # The file with shocks to obtain TMS = 1 #;

WRITE TO_HAT TO FILE TOHAT  ;
WRITE ??? TO FILE ???  ;

DISPLAY TO_L ;
```

Suggested steps for Example 12

(1) Change directory into \GTAPBOOK, copy SHKX.TAB to SHOCKSX.TAB and then edit the file SHOCKSX.TAB to complete it. Use the commands

cd \gtapbook
copy shkx.tab shocksx.tab
edit shocksx.tab

and then make the required changes. [Before exiting from the editor, save your changes by touching **Alt**, then the **f** (file) and **s** (save) keys.]

2) Run TABLO (as in Example 5) to process this file. Look in the Information file SHOCKSX.INF to find any errors. (Use **edit shocksx.inf** and search for two percent signs %% since each error is marked with %%.) If you find a **syntax error OR a semantic problem**, you must make the required change in SHOCKSX.TAB (not SHOCKSX.INF).

(3) When your TABLO Input file is correct and you have prepared GEMSIM Auxiliary files, run GEMSIM interactively (as in Example 6 above) to produce the display file SHKX2-01.DIS and two output files TOHAT.SHK and TMSHAT.SHK containing the **to** and **tms** shocks respectively. When running GEMSIM, attach the 3x3 data files SET2-01.HAR and DAT2-01.HAR. The responses for GEMSIM are shown below.

Input to GEMSIM

```
<carriage-return>    ! Use the default options and finish the option selection
shocksx              ! Name of the GEMSIM Auxiliary files
shkx2-01.dis         ! Name of display file to be created
set2-01.har          ! Name of GTAPSETS file for the 3x3 data
dat2-01.har          ! Name of GTAPDATA file for the 3x3 data
tohat.shk            ! Name of output TOHAT file containing to shocks
tmshat.shk           ! Name of output TMSHAT file containing tms shocks
```

End of Input to GEMSIM

Note. If you are uncertain about your TABLO Input file SHOCKSX.TAB, you can compare it with the file SHKXOK.TAB which we have prepared and which should be in your \GTAPBOOK directory.

(4) The Display file SHKX2-01.DIS contains the TO_L and TMS_L results. Look at this display file SHKX2-01.DIS in your text editor to see these values. Make sure that your file SHKX2-01.DIS has both lots of values in it. [If not, should you have added another DISPLAY statement at the end of your TABLO Input file SHOCKSX.TAB?]

(i) Are there any output subsidies? Any output taxes? Any examples of no output distortion?

(ii) Are there any import tariffs? Any import subsidies? Any examples of no import distortion?

(5) The shocks required to remove all output subsidies/taxes are in file TOHAT.SHK. Select one commodity and one region and check that the shock shown in TOHAT.SHK is consistent with the TO_L value shown in SHKX2-01.DIS.

(6) Repeat (5) for the shocks to remove import tariffs/subsidies in file TMSHAT.SHK.

Example 13 - Compute all distortions and the shocks needed to eliminate them (using SHOCKS.TAB)

In SHOCKSX.TAB, you calculated just two of the possible distortions. The TABLO Input file SHOCKS.TAB calculates these and most other distortions, and writes the shocks required to eliminate them to files like TOHAT.SHK and TMSHAT.SHK in Example 12.

To carry out all the calculations in SHOCKS.TAB, you need to follow the two Steps as above for SHOCKSX.TAB, as described briefly below.

(a) Implement SHOCKS by running TABLO, following Example 5 above. This will produce the GEMSIM Auxiliary files SHOCKS.GSS and SHOCKS.GST.

(b) Then run GEMSIM, this time taking inputs from the Command file SHK2-01.CMF. This produces a Display file SHK2-01.DIS containing the various distortions (for example, TO_L and TMS_L) and several different .SHK files containing the values of the shocks to appropriate variables of the GTAP94 model required to eliminate the different distortions.
You should look at some of the values in SHK2-01.DIS and compare these with the relevant shock given in the appropriate .SHK output file. (For example, compare TXS_L values in SHK2-01.DIS with shocks in TXS2-01.SHK.)

You will use the output from this Example in Example 26 below where you will simulate the removal of some of the distortions.

B. SIMULATIONS WITH GTAP

1. The Model Implementation TP1010

The theory of the GTAP model as used in the GTAP Book is contained in the TABLO Input file GTAP94.TAB. This allows different aggregations with up to 10 regions and up to 10 tradeable commodities.

In order to solve the model on an 8Mb PC, a particular implementation, referred to as **TP1010**, is used. In this implementation,

(a) certain policy variables have been omitted. This means that these variables are effectively exogenous but cannot be shocked. (When you work with this implementaion, it is as if these variables had never been present in the model.) The variables omitted are:

tf tpm tpd tgm tgd tfm tfd

(b) certain variables have been selected for backsolving. This means that these variables are hard-wired as endogenous. When you run a simulation, you can obtain results for these variables. However, they cannot be set exogenous in this implementation. The variables which are backsolved for are:

pfd ppm pfm pms pfob pcif pf ppd pgm pgd qfm qfd pva qfe qva qf
pgov qg pg qgm qgd qp qpd qpm

You can look in GTAP94.TAB to find out more about these variables.

The computer version of TP1010 consists of the **executable image TP1010.EXE** and the two **Auxiliary files TP1010.AXS** and **TP1010.AXT** which should all be in your \GTAPBOOK directory. **[**These were produced by running the program TABLO taking inputs from the Stored-input file TP1010TG.STI. This file includes instructions to omit and to backsolve as indicated above. However, the GTAP Book version of TABLO cannot be used with TP1010TG.STI ; a source-code version of GEMPACK is required to produce TP1010.EXE.]

With the GTAP Book Version of GEMPACK, all simulations with TP1010 are carried out by running the program **TP1010** (and telling it to use these Auxiliary files TP1010.AXS and TP1010.AXT).

2. Simulations with TP1010

The procedure for carrying out a simulation with any model is shown in Figure 1 in chapter 6 of the GTAP Book. In carrying out simulations with TP1010, you will first run TP1010 and then run GEMPIE (as indicated in Figure 1 in chapter 6).

Examples 14-27 below all involve simulations with GTAP. Most are with the 3x3 data but some are with other GTAP data sets.

3. Numeraire simulations

In Examples 14 and 15 you will carry out these two Steps for a numeraire simulation with the 3x3 data. This is a simulation in which the usual numeraire for GTAP (namely the variable **psave** - the price of capital goods supplied to savers) is increased by 10 per cent.

Example 14 - Run TP1010 to carry out the simulation

Start the program TP1010 running via the command

tp1010

When the program starts to run, give the two responses below.

cmf
num2-01.cmf

There will be quite a lot of screen activity. The program TP1010 is reading the 3x3 data files, and applying the shocks. When this has finished, you should check that is has created the so-called **Solution file** NUM2-01.SL4 by typing

dir *.sl4

Example 15 - Run GEMPIE to convert results to human-readable form

The Solution file NUM2-01.SL4 produced in Example 14 above contains the percentage changes in all endogenous variables (for example, certain prices and quantities). However this Solution file is a binary file, so you cannot examine the results yet. First you must run GEMPIE to convert these results to human-readable form.

To do this, start the program GEMPIE running via the command

gempie

Give the responses below.

Input to GEMPIE

```
<carriage-return>    ! Use the default options and complete option selection
num2-01              ! Name of the Solution file (created in Example 14)
a                    ! Print results for ALL endogenous variables
<carriage-return>    ! Use default name NUM2-01.PI5 for the GEMPIE Print file
Numeraire sim with 3x3 data   ! Page heading
5                    ! Number of decimal places in results
```

End of Input to GEMPIE

When this has finished you can look at the results of the simulation via the command

edit num2-01.pi5

Move around the file using the **PgDn** and **PgUp** keys, for example. You should expect that prices and dollar values will increase by 10 percent and that quantities will stay unchanged. Is this what you observe? For example, look at the results for variables **qo** (a quantity) and **ps** (a price).

Example 16 - The Command file NUM2-01.CMF

The simulation itself was carried out in Example 14. There the program TP1010 took all of its inputs from the **GEMPACK Command file** NUM2-01.CMF. This file tells TP1010 all the information required to specify a simulation, including the closure, the pre-simulation data files, the shocks and the post-simulation data files (that is, updated data).

First look at this file NUM2-01.CMF via

edit num2-01.cmf

Move around the file to see the various parts of the file. Find the places where
(a) the closure is specified,
(b) the shocks are specified,
(c) TP1010 is told which pre-simulation data files to read,
(d) the name of the Solution file is specified,
(e) TP1010 is told which post-simulation data files to write.
There are other parts of the file which you will learn about later. Now exit from the file.

Now suppose that you wanted to change this file so as to give a **5 percent decrease** to the numeraire. First copy NUM2-01.CMF to NUM2-01X.CMF and then edit this new file NUM2-01X.CMF so that it carries out the same simulation except for the size of the shock. Use the commands

copy num2-01.cmf num2-01x.cmf
edit num2-01x.cmf

In making the changes, look also at the comments and the verbal description and change them to be compatible with your new shock. [If you are uncertain about your file NUM2-01X.CMF, you can compare it with the file NM201XOK.CMF we have prepared.]

You won't rerun TP1010 using NUM2-01X.CMF here, but you will do this when you do Example 19 below.

Example 17 - Look at the updated data

As well as producing a Solution file containing percentage changes to prices and volumes, the run of TP1010 in Example 14 above produced updated data files, notably an updated version of the global data for the model. This updated data reflects the state of the economy as it would be after the shock (10 per cent increase in the numeraire) has worked its way through the economy. For the simulation in Example 14 above, this updated global data is in the file called **NUM2-01.UPD**. Because NUM2-01.UPD is a binary file (a Header Array file), you must run SEEHAR to convert it to a text file.

Start the program SEEHAR running (in the usual way). When prompted, respond as shown below.

Input to SEEHAR

```
<carriage-return>    ! Use the default options and complete option selection
num2-01.upd          ! Name of updated data file (created in Example 14)
num2-01u.see         ! Name of output file ("U" for updated)
6                    ! Number of decimal places
<carriage-return>    ! (As demanded by SEEHAR)
o                    ! Output ONE array
VIPA                 ! Header for array of data to output
<carriage-return>    ! (As demanded by SEEHAR)
e                    ! Exit from SEEHAR
```

End of Input to SEEHAR

The 3x3 array of data at header "VIPA" contains the Value of Private household expenditure on each Imported commodity in each region, valued at Agents' prices. The file NUM2-01U.SEE will contain this data from the updated (that is, post-simulation) data.

To compare this with the original values of this data (in the pre-simulation file DAT2-01.HAR), recall that DAT2-01.SEE you created in Example 2 above contains all arrays in DAT2-01.HAR. Compare the VIPA values from the pre- and post-simulation data. Are the differences as you would expect? [What has happened to the price and the quantity? Hence, what should happen to the value?]

Example 18 (optional) - Run GTAPCHK on the post-simulation data

The updated global data NUM2-01.UPD (see Example 17 above) should satisfy the same balancing conditions as the original global data DAT2-01.HAR. You can check this by carrying out the instructions in GTAPCHK.TAB (see Examples 5-8 above) on NUM2-01.UPD (rather than DAT2-01.HAR).

To so this, run GEMSIM interactively as in Example 6 above, but respond NUM2-01.UPD instead of DAT2-01.HAR and call the output file say CKNU2-01.DIS instead of CHK2-01.DIS. For example, the CHKMKTCLTRD values in CKNU2-01.DIS should be zero (to machine accuracy) again.

Example 19 (optional) - Repeat for a 5 percent decrease in the numeraire

To see the effects of a 5 percent decrease in the numeraire, repeat the simulation in Examples 14 and 15 above, but instead take inputs from the Command file NUM2-01X.CMF you prepared in Example 16 above. Check that the results are as expected (namely no change in quantities and 5 percent reductions in prices and domestic dollar values).

Example 20 - Numeraire simulation with a larger data set

In Examples 14 and 15 above you carried out a numeraire simulation (increasing **psave** bt 10 percent) with the 3x3 data. In this examples you will carry out the same simulation with the 10x10 data set used in the MFA application in the GTAP book, namely GTAP aggregation number 6..

The easiest way to prepare a Command file for this simulation is to base it on NUM2-01.CMF used with the 3x3 data. So first copy NUM2-01.CMF to NUM2-06.CMF and then edit this file via the commands

copy num2-01.cmf num2-06.cmf
edit num2-06.cmf

Change **all** occurrences of "2-01" to "2-06" and change all occurrences of "3x3" to "10x10". No other changes are needed.

Then run TP1010 as in Example 14, though use the Command file NUM2-06.CMF instead of NUM2-01.CMF, and then run GEMPIE much as in Example 15. Again check the results for variables **qo** and **ps**.

4. A Simulation Reducing One Distortion

In Examples 21-25 below, you will carry out a simulation with the 3x3 data in which the import tariff on Food from the USA imported into the European Union (EU) is reduced by ten percent. (In the 3x3 data base, the power of this tariff is approximately 1.369, as you can see from the TMS_L values produced in Example 12 above.)

Example 21 - Preparing a GEMPACK Command file

Here you will prepare a GEMPACK Command file TMSEU.CMF for carrying out this simulation. Start from NUM2-01.CMF used in the numeraire simulation, copy it to TMSEU.CMF and then edit it to make the required changes. Use the commands

copy num2-01.cmf tmseu.cmf
edit tmseu.cmf

There are several changes you should make.

(1) Shock. The variable in question is **tms** (the percentage change in the power TMS_L of the import tariffs.) This variable has three arguments. The notation

tms(i,r,s)

means the percentage change in the power of the import tariff levied in region **s** on commodity **i** from region **r**. Thus to reduce the power of the import tariff on USA Food imported into the EU

by 10 percent, you must specify a shock of -10 to **tms("food","usa","eu")**. Find the shocks part of the file TMSEU.CMF and replace it by the statement

shock tms("food","usa","eu") = -10 ;

(2) Solution file name. Change the "solution file =" statement so that the Solution file is called TMSEU.SL4. Since GEMSIM automatically adds the standard Solution-file suffix .SL4, you should change this statement to

solution file = tmseu ;

(3) Verbal description. This is several lines of text describing the simulation. It is placed on the Solution file and is reported when you look at simulation results. Find the statement "verbal description = ... ;" and change it appropriately. Your new verbal description can consist of several lines of text. Make sure that you end it with a semi-colon ';'.

(4) Names of updated data files. The post-simulation global data will be different from that in the numeraire simulation, so you should change the name of the updated global data (it is referred to as file GTAPDATA). We suggest that you call it TMSEU.UPD using the root TMSEU as in the name of the Command file and the Solution file. So, find the statement "updated file gtapdata = ... ;" and change it accordingly.
[Note that, although you must specify names for the updated parameters and set-information files, the "updated" versions of these are the same as the original versions. Hence there is no need to change their names.]

(5) Log file name. Change the name of the LOG file to TMSEU.LOG (again using the same root TMSEU).

(6) Since the closure and pre-simulation data is the same as for the numeraire simulation, there are no other changes required.

(7) Change the heading at the top of the file to TMSEU.CMF. Then exit from **edit**, saving the changes to TMSEU.CMF.

If you are uncertain about any part of your file TMSEU.CMF, you can look in the file TMSEUOK.CMF which we have prepared. [If, when you use your new TMSEU.CMF, the program says there is an error which you can't detect, try using the file TMSEUOK.CMF instead of TMSEU.CMF.]

Example 22 - Run TP1010

Start TP1010 running and tell it to take its inputs from the Command file TMSEU.CMF. When it finishes, check that the Solution file TMSEU.SL4 has been created via an appropriate **dir** command.
Example 23 - Run GEMPIE

Follow the method in Example 15 above to run GEMPIE to convert the simulation results in TMSEU.SL4 to a human-readable GEMPIE Print file TMSEU.PI5. This time choose **3 decimal places**. Then look at some of the results by editing this Print file. For example, check that the output of food in the USA **qo("food","USA")** increased by 0.688 percent.

Example 24 - Changing the solution method

In Example 22 above, TP1010 was told to use **Johansen's method** to calculate simulation results. With Johansen's method, only approximate solutions are obtained to the nonlinear levels equations of the model.

GEMPACK also provides **Euler's** (pronounced "oilers") and **Gragg's solution methods**. These can provide accurate solutions of the levels equations of the model. Here you will carry out the same simulation as in Example 22 above but this time using Gragg's method (the default in GEMPACK) to calculate the results.

To do this, you should copy TMSEU.CMF to TMSEUN.CMF ("N" for nonlinear), then edit TMSEUN.CMF.

(1) Find the statement

method = johansen ;

and replace it with the statements

method = gragg ;
steps = 2 4 6 ;

[Make sure that you end every statement in your Command file with a semi-colon **;**]
These two new statements tell TP1010 to use Gragg's method and to calculate so-called 2-step, 4-step and 6-step results and then extrapolate on the basis of these to obtain the final solution. The statement "extrapolation accuracy file = yes ;", which is already in your Command file, asks TP1010 to create a so-called **Extrapolation Accuracy file** which you can look at to see if the results are sufficiently accurate.

(2) Other changes.
(a) Change the names of the Solution file, updated GTAPDATA file and of the LOG file to TMSEUN.
(b) Change the verbal description to indicate that Gragg's method is being used.

Then exit from **edit**, saving the changes to TMSEUN.CMF.

[If you are uncertain about any part of your file TMSEUN.CMF, you can look in the file TMSEUNOK.CMF which we have prepared. If, when you use your new TMSEUN.CMF, the program says there is an error which you can't detect, try using the file TMSEUNOK.CMF instead of TMSEUN.CMF.]

Now rerun the simulation by running TP1010 and taking inputs from TMSEUN.CMF.

Then rerun GEMPIE to create a GEMPIE Print file TMSEUN.PI5.

Example 25 - Comparing these results

To see how the Johansen solution and the one obtained in Example 24 above using Gragg's method differ, you can compare the results in the Print files TMSEU.PI5 and TMSEUN.PI5. For example, compare the results for variable **qo**.

To see how accurate the Gragg results are, look at the Extrapolation Accuracy file (it is a text file so you can use **edit**) TMSEUN.XAC produced in Example 24. For example, the **qo** part of this starts with a line very similar to

4 0.879331 0.884487 0.885479 0.886282 CX 3

These are respectively the 2-step, 4-step, 6-step and extrapolated results for component number 4 (Food) of **qo**. The annotation "CX 3" indicates that this result is judged to be accurate to at least 3 figures and that we can be confident about the extrapolation.

Then go to the end of this Extrapolation Accuracy file. This tells you how many of the results are judged to be accurate to 3 figures, 4 figures etc.

You might like to estimate the sizes of the errors in the Johansen solution (relative to the more accurate Gragg solution) for the different variables of the model. [Look in TMSEU.PI5 for the Johansen results and in TMSEUN.PI5 for the more accurate Gragg results.] For example, perhaps look at the relative errors for quantity variables such as **qo** and **qxs** and welfare measures such as **u** and **EV**. You will see that the relative errors are greater for the welfare measures than for the quantity variables. The largest relative error is perhaps for the Equivalent Variation **EV("eu")**, for which the accurate result is about $US61 million and the Johansen result is about $US344 million.

5. A CAP Reform Simulation Removing Several Distortions with the 3x3 data

In Example 26 below you will carry out a simulation removing all distortions on Food in the European Union. More precisely, you will simulate reform in the Common Agricultural Policy of the European Union by removing

- output subsidies on food and farm products in the European Union,
- export subsidies on food and farm products exported from the European Union, and
- import taxes on food and farm products imported into the European Union.

This will be carried out on the 3x3 data set, where the single commodity "food" includes farm products.

The levels values of the three distortions above (respectively those of TO_L, TXS_L and TMS_L) and the sizes of the shocks to variables **to**, **txs** and **tms** respectively to remove these distortions have been calculated in Example 13 above. The required shocks to remove all of these distortions are in the files TO2-01.SHK, TXS2-01.SHK and TMS2-01.SHK respectively. However, in this simulation we only wish to remove those affecting Food and the European Union; that is, we wish to give the appropriate shocks just to

to("food","eu") ! output of food in EU,
txs("food","eu","usa") and **txs("food","eu","row")** ! exports of food from EU,
tms("food","usa","eu") and **tms("food","row","eu")** ! imports of food into EU.

The first is achieved via the statement

shock to("food","eu") = select from file to2-01.shk ;

in a Command file; the "select from" tells TP1010 to select just the shocks to the appropriate component of **to** from all the shocks in the file TO2-01.SHK. Similar statements achieve the desired shocks to the relevant components of variables **txs** and **tms**.

You should look at the Command file CAPMRGE.CMF which we have prepared. You will see the shock statements and various other statements for carrying out this simulation.

Example 26 below relies on the three files TO2-01.SHK, TXS2-01.SHK and TMS2-01.SHK produced in Example 13 above. Before starting Example 26, check that these three files are present via the command

dir t*.shk

Example 26 - Carry out this CAP reform simulation

First run TP1010, taking inputs from the Command file CAPMRGE.CMF. Once this finishes running, check that the Solution file CAPMRGE.SL4 has been created. Then run GEMPIE to produce a GEMPIE Print file CAPMRGE.PI5 containing the simulation results (perhaps choose 3 decimal places for these results). Examine some of the results to see if they are what you would expect from this reform. (For example, what happens to the output of Food in USA and ROW? Why?)

This simulation used 2,4,6-step Gragg calculations to obtain a fairly accurate solution. Perhaps examine the Extrapolation accuracy file CAPMRGE.XAC to see how accurate some of the results are. Look at the summary at the end of the file.

Example 27 - Checking that the CAP reform has worked

Part of the output from the simulation is the updated global data file CAPMRGE.UPD. This contains the global data as it would be once the relevant distortions have been removed.

Examining it (how can you do this?) should enable you to confirm that these distortions have in fact been removed.

The best way to check this is to use SHOCKS.TAB again. The idea is to use the instructions in SHOCKS.TAB to calculate the sizes of the relevant distortions in the updated (that is, post-simulation) data. To do this, you will need to modify SHK2-01.CMF (which you used in Example 13 above to compute the shocks for the CAP reform simulation) to calculate the levels of the distortions in the updated data. To do this, first copy SHK2-01.CMF to a new file SHKCAP.CMF via

copy shk2-01.cmf shkcap.cmf

and then edit SHKCAP.CMF. Make sure that CAPMRGE.UPD is the global data set it reads (edit the "file gtapdata" statement) and change the names of the output display file to CAPMRGE.DIS. Also add the statement

nwr = yes ;

which will tell GEMSIM not to do any WRITEs; this will mean that new shocks files will not be written. Then save the new SHKCAP.CMF. [If you are uncertain about your file, check the file SHKCAPOK.CMF which we have prepared.]

Now run GEMSIM and take inputs from SHKCAP.CMF. Once this has run, look in the new Display file CAPMRGE.DIS. Look at the values of **TO_L("food","eu")** and **TMS("food","usa","eu")**. What values do you expect? (Remember that these are powers of the relevant taxes and so they are equal to 1 if there is no distortion.) What do you find? Repeat for the other relevant values (those changed in the simulation). [If any of these is not sufficiently close to 1, you could repeat the simulation in CAPMRGE.CMF, this time increasing the number of steps to get a more accurate solution.]

C. REPORTING VOLUME CHANGES VIA GTAPVOL.TAB

When you carry out a simulation using GTAP, the Solution file (or GEMPIE Print file) gives information about percentage changes in volumes. (For example, the **qo** results give information about changes in the outputs of different commodities in the different regions.) The TABLO Input file GTAPVOL.TAB has been designed to report the corresponding **changes** (rather than percentage changes) in some of these volumes.

For example, the variable **qo(i,r)** shows percentage changes in the output of commodity **i** in region **r**. To convert this to changes **DQO(i,r)** in these outputs, GTAPVOL uses the formula

DQO(i,r) = [qo(i,r)/100] * VOM(i,r) **(1)**

to calculate DQO(i,r). Here VOM(i,r) is the pre-simulation value of outputs at market prices, *in the initial equilibrium*. Although the global data file DAT2-0X.HAR tells us pre-simulation

dollar values of these outputs, it does not tell us pre-simulation volumes. Obtaining pre-simulation output volumes is only possible if we make some decision about the units of these volume measurements.

GTAPVOL bases its output on the idea that the number of units output initially is the same as VOM(i,r); that is, **one unit of commodity i in region r** is defined to be the amount that one dollar will buy (at pre-simulation market prices). Volume changes are then reported by GTAPVOL in these same units. Thus DQO(i,r) does not equal the change in the *values* of outputs since it does not include the price changes induced by the shock.

For example, in the 3x3 data set, VOM("food","eu") is 1168310 million dollars. GTAPVOL says that the pre-simulation output of food in the EU is 1168310 million units.

In order to apply formulas like (1) above, GTAPVOL must know pre-simulation volumes and the percentage changes (from the simulation) in these. The pre-simulation volumes are computed from the global data in the DAT2-0X.HAR file for the relevant GTAP data set. Percentage changes must be obtained from the simulation results. GTAPVOL.TAB cannot read a Solution file directly. However the GEMPACK program **SLTOHT** (**Sol**ution **to H**eader or **T**ext) can be used to convert the solutions on a Solution file to a Header Array file. GTAPVOL can then read the solution results from this Header Array file.

Therefore there are two steps in reporting volume changes.

1. Run SLTOHT to convert selected simulation results to a Header Array file.
2. Run GEMSIM, taking instructions from the GTAPVOL GEMSIM Auxiliary files GTAPVOL.GSS and GTAPVOL.GST.

Also, before step 2 above, it is necessary to run TABLO to implement GTAPVOL.TAB, producing GTAPVOL.GSS and .GST in the process. (This run of TABLO only needs to be done once, not every time you run GTAPVOL.)

Below we show how you can use this procedure to report the volume changes induced by the shock in the partial liberalization simulation carried out using Command file TMSEUN.CMF (see Examples 24-25 above).

You must complete the simulation in Example 25 above before you can carry out the steps below.

Example 28 - Implementing GTAPVOL

Run TABLO to implement GTAPVOL.TAB. This is very similar to Example 5 above. Just respond as there, except replace each response "gtapchk" by a response "gtapvol".

[This step only needs to be done once. If you use GTAPVOL on other Solution files besides TMSEUN.SL4, you don't need to repeat this step; just do Steps 1 and 2 below (in Examples 29 and 30 respectively) for the new case.]

Example 29 - Run SLTOHT to convert the simulation results to a Header Array file

Run SLTOHT by entering

sltoht

Then respond as below.

Input to SLTOHT

```
<carriage-return>       ! Use default options and finish option selection
tmseun                  ! Name of Solution file
c                       ! Take the cumulative solution
y                       ! Use a Header Array Mapping file
gtapvol.ham             ! The Header Array Mapping file to use
tmseun.sol              ! Output Header Array file containing the solutions
```

End of Input to SLTOHT

When this finishes running, check that the output Header Array file TMSEUN.SOL (containing the simulation results for selected variables) exists via

dir *.sol

GTAPVOL.TAB only needs some of the simulation results. The **Header Array Mapping file** GTAPVOL.HAM is used to put just these results on the output Header Array file TMSEUN.SOL.

[For more information about SLTOHT and Header Array Mapping files, see section 8.3 of GPD-1. SLTOHT can also be used to import solution results into spreadsheets.]

Example 30 - GTAPVOL applied to these results

Run GEMSIM accessing GTAPVOL.GSS and .GST

This can be done interactively or with a Command file. Below we show how to run it interactively. You can easily prepare a Command file to do the same run.

Start GEMSIM running (type **gemsim**) and then respond as below.

Input to GEMSIM

```
<carriage-return>   ! Take default program options and finish option selection
gtapvol             ! Name of GEMSIM Auxiliary files (from GTAPVOL.TAB)
tmseun.vol          ! Display file containing volume changes from the sim
set2-01.har         ! Name of set information file (logical name GTAPSETS)
dat2-01.har         ! Name of global data file (logical name GTAPDATA)
tmseun.sol          ! Header Array file containing sim results (from above)
```

End of Input to GEMSIM

When this finishes running, check that the output file TMSEUN.VOL has been created via

dir *.vol

For example, the pre-simulation dollar value of output of food in the EU is 1168310 million dollars at market prices (see the VOM results in the file CHK2-01.DIS produced in Example 6 above). This means that 1168310 million units (where, by GTAPVOL's definition, one unit is the amount of food that one dollar purchases, at pre-simulation market prices). The simulation result for **qo(food,eu)** in TMSEUN.SL4 is -0.354864 percent. Hence, from the formula (1) above, the **DQO(food,eu)** produced by GTAPVOL (and stored in file TMSEUN.VOL) should be

[-0.354864/100] * 1168310 = -4145.91 million units

This is the change in the volume of output of food as a result of the simulation. [You might like to find the **DQO(food,eu)** result in the file TMSEUN.VOL to check this result.]

Example 31 - GTAPVOL and other simulations

For another simulation with Solution file say SIM1.SL4, you can compute the volume changes by following Steps 1 and 2 (that is, Examples 29 and 30) above. We suggest just replacing "tmseun" in all responses shown above by "sim1". The output from Steps 1 and 2 will be SIM1.SOL and SIM1.VOL respectively. [You do not need to repeat the run of TABLO in Example 27.]

Of course, if you are using a different aggregation than the 3x3 one, you will need to give the name of the relevant global data file DAT2-0X.HAR when running GEMSIM in Step 2. The Header Array Mapping file GTAPVOL.HAM (see the run of SLTOHT above) can be used with all simulations and all different GTAP aggregations.

For a concrete example, carry out the steps to obtain the volume changes resulting from the CAP reform simulation carried out above in Example 26.

D. COMPUTING EQUILIBRIUM ELASTICITIES

The next examples enable you to compute the equilibrium elasticities (price and income elasticities of demand) discussed in section III of chapter 5 of the GTAP Book and shown in tables 5 and 6 in that chapter. In Examples 32-36 below you will calculate the price elasticities of demand for the 3x3 GTAP data set using the standard GE closure of GTAP; these are the elasticities in table 5A of chapter 5 of the GTAP Book. In Example 37 below you will calculate the price elasticities of demand for the 3x3 GTAP data set using the PE closure discussed in chapter 5; these are the elasticities in table 5B of chapter 5 of the GTAP Book. In Examples 38-39 below you will calculate the income elasticities of demand for the 3x3 GTAP data set using this PE closure; these are the elasticities in table 6 of chapter 5 of the GTAP Book.

1. Price Elasticities of Demand, GE Closure

The price elasticities of demand with respect to a specific price, say **pm(j,s)**, are obtained by carrying out a Johansen simulation shocking **to(j,s)** (the power of the output tax intervention) by 1, then dividing the simulation results for variable **qo** by the simulation result for the relevant price **pm(j,s)**. That is, in the notation used in chapter 5 of the GTAP Book,

$$\varepsilon_{(i,r),(j,s)} = qo(i,r) / pm(j,s) .$$

The matrix of elasticities is an **(i,r) x (j,s)** matrix. Each column of it is obtained from the results of a single Johansen simulation, shocking **to(j,s)**. The GEMPACK program SAGEM can carry out all the different simulations (as **j** and **s** vary) in one run. This is done in Example 32 below.

Example 32 - Price elasticities simulation, GE closure

Start the program SAGEM running via the command

sagem

When the program starts to run, give the two responses

cmf
elga2-01.cmf

This should produce a Solution file ELGA2-01.SL4, which you can check via the command

dir elga*.sl4

Example 33 - Look at the results

To look at the results of the simulations carried out in Example 32, run GEMPIE via the command

gempie

When the program begins to run, give the responses shown below.

Input to GEMPIE

```
<carriage-return>   ! Take default program options and finish option selection
elga2-01            ! Name of Solution file (from Example 32 above)
i                   ! Look at individual column results
a                   ! Print all individually-retained exogenous columns
a                   ! Print all individually-retained endogenous results
<carriage-return>   ! Take default name ELGA2-01.PI5 for Print file
Price elasticities, GE closure    ! Page heading
4                   ! Number of decimal places
```

End of Input to GEMPIE

Then look at the resulting Print file via the command

edit elga2-01.pi5

To see the simulation results, search for "PAGE 1" (use the **Alt**, **s**, **f** keys); this will bring you to the start of the simulation results. You will see three columns of results, corresponding to shocks to components 4, 5 and 6 of variable **to**; these are **to(food,USA)**, **to(mnfcs,USA)** and **to(svces,USA)** respectively. In each column are the **qo(i,r)** results and the **pm(i,r)** results.

The elasticities of the **qo(i,r)** with respect to **pm(j,s)** are obtained by dividing the **qo(i,r)** results by the **pm(j,s)** result in the column corresponding to a shock to **to(j,s)**.

Thus the elasticities of the **qo(i,r)** with respect to **pm(food,USA)** are obtained by dividing the **qo(i,r)** results in the first column by the **pm(food,USA)** result in this column. For example, the elasticity of **qo(mnfcs,EU)** with respect to **pm(food,USA)** is 0.0246/-1.3159 = -0.019, which confirms the value in the **eu_m** row and **usa_f** column in table 5A in chapter 5 of the GTAP Book.

Similarly the elasticities of the **qo(i,r)** with respect to **pm(svces,EU)** are obtained by dividing the **qo(i,r)** results in the column corresponding to the shock to **to(svces,EU)** [this is component number 13 of **to**] by the **pm(svces,EU)** result in this column. [To find this column (it is on PAGE 2 of the results part of the Print file), move down the file using the **PgDn** key; look for the heading about component number 13 of **to**.] For example, the elasticity of **qo(food,ROW)** with respect to **pm(svces,EU)** is 0.0195/-0.6365 = -0.031, which confirms the value in the **row_f** row and **eu_s** column in table 5A in chapter 5 of the GTAP Book. Now exit from ELGA2-01.PI5.

As you can see, all 81 elasticities in table 5A of chapter 5 can be calculated by dividing two results in ELGA2-01.PI5. But this is a lot of divisions to do by hand (and, for a 10x10 GTAP data set, there would be 10,000 divisions to do). The next three examples show how these divisions can be automated. In Example 34, the simulation results in the Solution file produced in Example 32 are converted to a Header Array file, which is in turn read in Example 36, where

the appropriate divisions are carried out, using the READs and FORMULAs in the TABLO Input file PRICELAS.TAB which is implemented in Example 35.

Before you go on to these, you might like to look at the Command file ELGA2-01.CMF which was used to run SAGEM in Example 32 above to carry out the 9 different Johansen simulations. [Use the command "edit elga2-01.cmf".] There you will see the statements setting up the closure and shocks; these will look familiar to you. There are also less familiar statements about "individually-retained" variables. These are the ones saying that we want all 9 columns of results (one for each component of **to** shocked) and all 18 rows of results in each column (9 for the relevant components of **qo** and 9 for the relevant components of **pm**).

Example 34 - Convert the simulation results to a Header Array file

To convert the simulation results in the Solution file ELGA2-01.SL4 produced in Example 32, run the program SLTOHT via the command

sltoht

When the program starts to run, give the two responses

sti
elgb2-01.sti

This should produce the Header Array file ELGA2-01.SOL, as you can see via the command

dir elga*.sol

Example 35 - Look at and implement PRICELAS.TAB

The TABLO Input file PRICELAS.TAB contains instructions for reading the simulation results from a Header Array file (such as that just produced in Example 34), doing the appropriate divisions, and then writing the resulting price elasticities of demand to a text file. If you look at PRICELAS.TAB (via the command "edit pricelas.tab"), you can see the various statements. For example, you can see a COEFFICIENT EPSILON(i,r,j,s) which is the TABLO notation for what is called $\varepsilon_{(i,r),(j,s)}$ in chapter 5, and, near the end of the file, you can see the FORMULA for EPSILON(i,r,j,s) dividing QO(i,r,j,s) by PM(j,s). There are also various technicalities (which you should not worry about) in this file, including the "IF" around the division formula for EPSILON(i,r,j,s) to ensure that division by zero is not attempted, and various mappings between sets.

Now exit from PRICELAS.TAB and run TABLO to implement the instructions in PRICELAS.TAB via the command

tablo

Give responses as in Example 5 above, replacing "gtapchk" by "pricelas" in each case. This will produce GEMSIM Auxiliary files PRICELAS.GSS and PRICELAS.GST as you can see via the command

dir pricelas.gs*

Example 36 - Carry out the calculations in PRICELAS.TAB

To carry out the calculations, run GEMSIM via the command

gemsim

and give the two responses

cmf
elgc2-01.cmf

This should produce the text file ELGE2-01.ELS containing the values of EPSILON(i,r,j,s), which are just the required price elasticities of demand. To see these values, look at this file via the command

edit elge2-01.els

Notice that the first 3x3 matrix of values contains the **EPSILON(i,r,food,USA)** values, which correspond to the **usa_f** column in table 5A in chapter 5. For example, find the elasticity of **qo(svces,EU)** with respect to **pm(food,USA)**; that is, find the value of **EPSILON(svces,EU,food,USA)**, and check with the relevant value in table 5A. Repeat for other elasticities such as the elasticity of **qo(svces,ROW)** with respect to **pm(food,EU)**.

2. Price Elasticities of Demand, PE Closure

The calculation of the price elasticities of demand in the PE case is very similar to the GE case. The main difference is, of course, the closure used in the simulations. The PE case is carried out in Example 37 below. This has just 3 steps,

(a) the simulations via SAGEM (similar to Example 32),
(b) the conversion of the simulation results to a Header Array file (similar to Example 34), and
(c) the calculation of the elasticities (similar to Example 36).

Example 35 (implementing PRICELAS.TAB) does not need to be repeated, and looking at the simulation results as in Example 33 above is not really required (though it may be useful to do that again here).

Example 37 - Price elasticities of demand, PE closure

(a) First carry out the simulations by running SAGEM taking all inputs from the Command file ELPA2-01.CMF (which you might like to look at, for example, to check the closure, before running SAGEM). When you run SAGEM, give responses

cmf
elpa2-01.cmf

This should produce the Solution file ELPA2-01.SL4 (which you can check). [You may wish to look at the simulation results and calculate one or two elasticities, following the method in Example 33 above, with the obvious changes; but this is not essential, since all calculations are done in (c) below.]

(b) Now run SLTOHT to convert the simulation results in ELPA2-01.SL4 to a Header Array file. When SLTOHT starts to run, give the two responses

sti
elpb2-01.sti

This should produce the Header Array file ELPA2-01.SOL (which you can check).

(c) Finally run GEMSIM to carry out the instructions in PRICELAS.TAB (to divide the **qo** results by the appropriate **pm** result). When GEMSIM starts to run, give the two responses

cmf
elpc2-01.cmf

This should produce the text file ELPE2-01.ELS containing the required price elasticities of demand. You should look in this file to see their values and should compare them with the values in table 5B in chapter 5 of the GTAP Book.

3. Income Elasticities of Demand, PE Closure

The income elasticities of demand are easier to calculate than the price elasticities since regional incomes are exogenous in the relevant PE closure (whereas the prices **pm** are endogenous). In Example 38 below you will carry out the relevant simulation (shocking each component of the income variable **y** by one percent). The required elasticities are on the Solution file from this simulation, so all that remains is to run GEMPIE in Example 39 to look at the results (and hence the elasticities).

Example 38 - Carry out the simulations for income elasticities

Carry out the simulations by running SAGEM taking all inputs from the Command file ELI2-01.CMF (which you might like to look at, for example, to check the closure and shocks, before running SAGEM). When you run SAGEM, give responses

cmf
eli2-01.cmf

This should produce the Solution file ELI2-01.SL4 (which you can check).

Example 39 - Convert the elasticities to human-readable form

Run GEMPIE, giving the responses shown below.

Input to GEMPIE

```
<carriage-return>   ! Take default program options and finish option selection
eli2-01             ! Name of Solution file (from Example 38 above)
i                   ! Look at individual column results
a                   ! Print all individually-retained exogenous columns
a                   ! Print all individually-retained endogenous results
<carriage-return>   ! Take default name ELI2-01.PI5 for Print file
Income elasticities, PE closure    ! Page heading
3                   ! Number of decimal places
```

End of Input to GEMPIE

This should produce the GEMPIE Print file ELI2-01.PI5. You can look at the simulation results (and hence the income elasticities) via the command

edit eli2-01.pi5

There are 4 columns of results (search for "PAGE 1" to get to the beginning of the results). In the first three columns, the **qo(i,r)** results are the income elasticities of demand. The fourth column (the "TOTALS" column) is the sum of the other three; this contains the elasticities with respect to changes in world income. These four columns should agree with those in table 6 of chapter 5 in the GTAP Book.

4. Calculating elasticities for other GTAP data sets

The elasticities can be calculated for larger GTAP data sets, following similar steps to those in Examples 32-39 above. To do so, you will need to make appropriate modifications to the .CMF and .STI files. If so, it would be a good idea to change the "2-01" in their names to "2-0X" where X is the number of the relevant aggregation. You will also need to follow the instructions in SETP2-01.DAT to create the appropriate SETP2-0X.DAT file.

E. TIPS FOR USING GEMPACK

Interrupting Programs

Sometimes you wil start a program running and then realise that it is not doing what you intend. You can interrupt the program and return to the DOS prompt by typing **Control-C** (that is, hold down the **Ctrl** key, which is usually on the left of your keyboard, and, while holding it down, touch the **C** key.) Sometimes you may have to type Control-C twice to achieve this.

Controlling Screen Output

Often screen output goes much too quickly for you to read. You can control it using the

Control-S **Control-Q**

keystrokes. (For Control-S, hold down the **Ctrl** key and, while holding it down, touch the **S** key.) Use Control-S to stop the screen output and Control-Q to start it again. You can repeat these as needed. However, if you get out of step, say by typing two Control-S in a row, you will lose control of the output and have to wait until the program ends; even Control-C (see above) will probably fail then. [On some machines the **Scroll Lock** key works in a similar way. It first stops screen output, then starts it, then stops it, and so on.]

If a Multi-step Simulation Does Not Converge

Gragg's method or the midpoint method converge much more quickly that Euler's method for many simulations. (That is, they produce much more accurate results for the same numbers of steps.)

It is known that Gragg's method and the midpoint method are not suitable for some simulations; see However, we do not know how to predict in advance which simulations these are. (Fortunately it seems to apply to only a relatively small fraction of simulations.) In some cases when you do three multi-step Gragg calculations (as recommended) you may find that the results oscillate rather than converging, and you may find that these oscillations become worse when you increase the numbers of steps. If this happens, take it as *prima facie* evidence that Gragg's method is not suitable, and switch to Euler's method.

F. OTHER MODELS BESIDES GTAP AND OTHER GEMPACK PROGRAMS

Other Models Besides GTAP

Several example models apart from GTAP are available in the file EXGTB.ZIP on your computer. Details about these model files are given in Appendix B of GPD-1. The models supplied are

Stylized Johansen - a pedagogical 2-sector model
Miniature ORANI - a pedagogical single-country model
DMR - Dervis, De Melo and Robinson model of KOREA

and three intertemporal (that is, dynamic) models

TREES - a stylized model of forestry
CRTS - a single-sector investment model
5SECT - a 5-sector investment model

You can extract the files relating to one or more of these models from EXGTB.ZIP. For example, the commands

```
cd  \gtapbook
.\pkunzip  exgtb  sj*.*
```

will extract all files related to the Stylized Johansen model.

Suggestions for hands-on computing with some of these models are given in Appendix E of GPD-1. If you are interested in seeing how GEMPACK works with models other than GTAP, you might like to work through the hands-on computing in Appendix E of GPD-1.

MODHAR

Another GEMPACK program besides those mentioned in section VIII of chapter 6 in the GTAP Book is available for your use, namely MODHAR. This can be used to modify the data on a GEMPACK Header Array file. This program is documented in GEMPACK document GPD-3. In principle, you could use MODHAR to modify the global data in one of the GTAP DAT2-0X.HAR files. However, it is not a simple matter to keep the data balanced etc and so this is not a task you should undertake. MODHAR can also be used to build new Header Array files.

If you wish to use MODHAR, you will first need to extract it from its archive via the commands

```
cd  \gtapbook
.\pkunzip  -o   modhar
```

REFERENCES

W. Jill Harrison and K.R. Pearson (1994), *Computing Solutions for Large General Equilibrium Models Using GEMPACK*, Centre of Policy Studies and Impact Project Preliminary Working Paper No. IP-64, Monash University, June, pp.55.

Thomas W. Hertel, ed. (1996), *Global Trade Analysis: Modeling and Applications*, Cambridge University Press (to be published in 1996).

GEMPACK Documents

GPD-1, *An Introduction to GEMPACK*, Second edition, April 1994, pp.252+15.

GPD-2, *User's Guide to TABLO, GEMSIM and TABLO-generated Programs*, Second edition, April 1994, pp.138+14.

GPD-3, *How to Create and Modify GEMPACK Header Array Files Using the Program MODHAR*, Third edition, April 1993, pp.27+4.