

HANDS-ON COMPUTING WITH RunGTAP AND WinGEM TO INTRODUCE GTAP AND GEMPACK

Ken Pearson, Mark Horridge, Erwin Corong

August 2010; revised July 2018

This document gives several examples of hands-on computing you can carry out to familiarize yourself with the RunGTAP and GEMPACK software. 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 with GTAP. Part C describes how you can make a new version (that is, aggregation, of the GTAP data base) for use with RunGTAP. Part D assumes you already know something of GEMPACK -- it tells how you can adapt and run your own economic models within RunGTAP. In Part E, we tell you how can find out more about RunGTAP and GEMPACK, and mention hands-on computing you may like to try with models other than GTAP. Appendix A (not used at the Short Course) gives detailed instructions for carrying out GTAP simulations under WinGEM rather than via RunGTAP.

This document is designed for use at the GTAP Short Course¹. It can also be used after the course by participants in conjunction with the RunGTAP and GEMPACK software they take home with them from the course. And, with certain limitations mentioned in the next paragraph, this document will also be very useful to others learning to use RunGTAP.

This document assumes that you have

- (a) a recent version of RunGTAP² and
- (b) a PC running Microsoft Windows, and
- (c) a Source-Code or Executable-Image version of GEMPACK³, together with various GTAP-related files in a directory \GTAP on your computer.

If you do **NOT** have item (c) -- GEMPACK licence and course files -- you will be able to carry out most but not all of the activities described in this document. Boxes like this in the text indicates which sections require a GEMPACK licence. If you do not have a licence, either skip over these sections, or read through them without touching the keyboard. Activities using WinGEM require a GEMPACK licence.

As you read this document, you will use **RunGTAP** and **WinGEM** to work through examples designed to familiarize you with the software and, perhaps more importantly, the GTAP model. **RunGTAP** is designed specifically to help you solve GTAP interactively. **WinGEM** is the Windows interface to the GEMPACK general-purpose software for implementing and solving general and partial equilibrium models.

¹ This document is closely related to the **Hands-on document** "Hands-on Computing to Introduce GEMPACK and GTAP" referred to in chapter 6 "Implementing GTAP Using the GEMPACK Software" in the book *Global Trade Analysis: Modeling and Applications*, edited by Thomas W. Hertel - see Hertel (1997) in the References. Below we refer to this book as "**the GTAP Book**". The GTAP book Hands-on document uses an earlier (pre-Windows) version of GEMPACK and GTAP data based on an earlier version (version 2) of the GTAP data base. Otherwise the examples there are very similar to those in this document. The major difference is that this document assumes that you are working under Windows whereas the GTAP book Hands-on document requires you to work under DOS (or in a DOS box of Windows).

² RunGTAP is a Windows program, written by Mark Horridge, specifically for solving the GTAP model. It provides a Windows interface to various GEMPACK programs. Its Help|.About command displays version information.

³ Information about GEMPACK, and a free trial version, can be obtained at: <http://www.copsmodels.com/gempack.htm>

The instructions in this document are quite detailed. Our aim is to give sufficient detail to enable a new user of this software to work through the examples relying solely on this document.

This document also assumes you are using the Standard GTAP model version 7 (Corong *et al.*, 2017) and RunGTAP version 3.7 or later.

In this document we assume that you have installed RunGTAP and WinGEM on your computer. Detailed instructions for doing this are available elsewhere in the material distributed at the Short Course.

We are grateful to many GTAP users for helpful feedback on earlier versions of this document.

Contents

A. USING THE SOFTWARE TO FIND OUT ABOUT THE DATA	5
A.1 Examining the Data Directly using RunGTAP	5
Example A1 – Viewing the global data	6
Example A2 - Viewing the set information	8
Example A3 - Viewing the parameter values	9
Example A4 (optional) - Looking at the data in other GTAP versions	9
A.2 Using GTAPVIEW to report various aspects of the data	10
Example A5 – GTAPVIEW with the 3x3 data in NCORS3X3	10
Example A6 (optional) - GTAPVIEW with other data	12
Summary: Data and Consequences of the Data	12
A.3 Building a TABLO Input file to report other aspects of the data	12
Starting WinGEM	12
Setting the working directory for WinGEM	13
Carrying Out Calculations in TAB Files	13
Example A7 - Completing TABLO Input file GVIEWX.TAB	13
Example A8 – Running TABLO to Process GVIEWX.TAB	16
Example A9 – Running GEMSIM to do the Formulas (etc) in GVIEWX.TAB	16
Example A10 (optional) – The GVIEWX results for a different version	18
A.4 Finding the Size of Existing Distortions	18
4.1 Background and Notation	18
4.2 A TABLO Input file to report distortions and changes to eliminate them	19
Example A11 - Completing TABLO Input file SHOCKSX.TAB	19
Example A12 - Compute all distortions and the shocks needed to eliminate them	22
Summary of Part A. Data and Consequences of the Data	22
B. SIMULATIONS	23
B.1 Condensing the Model	23
Example B1 (optional) - Condensing GTAP to produce output for GEMSIM	24
Simulations with GTAP via RunGTAP	24
B.2 Numeraire simulation with the 3x3 data	25
Example B2 - Carrying out the basic numeraire simulation	25
Example B3 - Look at the Results	26
Example B3A - Looking at the Results using ViewSQL or ViewHAR	28
Example B4 - Look at the updated data	28
B.3 A Simulation Reducing one Distortion	29
The Simulation	29
Example B5 - What is the pre-simulation power of this tariff?	29
Example B6 - Implementing the shock and running the simulation	29
Example B7 - Looking at the results	31
Example B8 - Look at the updated data	32
Example B9 - Saving the solution and shocks	33
Example B10 - Changing the solution method	33
Example B11 - Has the power of the tariff actually been reduced by 10% ?	34
Example B12 - Copying results into a Spreadsheet	34

Example B13 - Looking at Volume Changes	35
B.4 A Simulation Removing Several Distortions with the 3x3 data	36
Example B14 - Carry out this simulation	36
Example B15 - Using Subtotals to divide up the effects of the shocks. [optional]	37
C. MAKING A NEW VERSION FOR RUNGTAP	38
Example C1 – Making a new aggregation using GTAPAgg	38
Example C2 – Adding this aggregation as a new version in RunGTAP	40
D. CUSTOMIZING RUNGTAP	41
Stage 1: Preparatory	41
Creating a new version based on the NCOR3x3 version	41
Enter Developer Mode and use TABmate as text editor	41
Regenerate GTAP as a GEMSIM program	42
Use the GEMSIM program for this version	42
Stage 2: Changing GTAPv7.TAB	42
Adding 1 new equation and 1 new variable to GTAPv7.TAB	42
Adding the new variable to GTAPv7.MAP	43
Running the new model	43
Complications which were avoided	43
E. FINDING OUT MORE ABOUT RUNGTAP AND GEMPACK	44
Connection between RunGTAP and GEMPACK	44
Other Models Besides GTAP	45
APPENDIX A: SIMULATIONS USING WINGEM	46
AA.0 Numeraire simulations	46
Example AA0 - Condensing GTAP to produce output for GEMSIM	46
Example AN1 - Carrying out the basic numeraire simulation	47
Example AN2 - Run ViewSOL to look at the results	47
Example AN3 - The Command file NUM4P04.CMF	49
Example AN3A (optional) - A Command file for a numeraire decrease	49
Example AN4 - Look at the updated data	49
Example AN5 (optional) - Run GTAPVIEW on the post-simulation data	50
Example AN6 (optional) - Repeat for a 5 percent decrease in the numeraire	51
Example AN7 (optional) - Numeraire simulation with a larger data set	51
AA.1 A Simulation Reducing One Distortion	51
Example AA1 - Preparing a Command file	51
Example AA2 - Run GEMSIM and ViewSOL	52
Example AA3 - Changing the solution method	53
Example AA4 - Comparing these results	53
AA.2 A Simulation Removing Several Distortions with the 3x3 data	54
Example AA5 - Carry out this simulation	55
Example AA6 - Checking that the simulation has worked	55
REFERENCES	57

A. USING THE SOFTWARE TO FIND OUT ABOUT THE DATA

The Examples here focus mainly on the 3x3 data set in RunGTAP version NCORS3X3. In section A.1 (Examples A1-A4) you will use RunGTAP to look at the data in a GTAP data set directly. In section A.2 (Examples A5-A6) you will look at GTPVEWv7.TAB which is used to report many useful pieces of information from a GTAP data set. In section A.3 (Examples A7-A10) you will learn how to use WinGEM and the GEMPACK program TABLO to compute other information from a GTAP data set. In section A.4 (Examples A11-A12) you will calculate the sizes of various distortions (for example, import tariffs).

Starting RunGTAP

Double-click the **RunGTAP** icon on your desktop. You will see a screen as below:

RunGTAP: NCORS3x3/GTAP

File Copy View Version Tools Help

Title RunGTAP Version Closure Shocks Solve Results

RunGTAP uses a tabbed notebook or card index interface. The first 2 pages (**Title** and **RunGTAP**) contain general information. The 3rd page contains information about the current version: the particular data aggregation of GTAP the program is using. The remaining 4 pages relate to simulations – we postpone those until Part B below.

Choosing the NCORS3X3 version

In this document (and for the first 3 days of the GTAP Short Course) you will work with the 3-commodity, 3-region aggregation of the GTAP data referred to as version **NCORS3X3**.⁴ [Later you can use the other versions of the data available via RunGTAP.]

To make sure that you are using this version click on

Version / Change

in RunGTAP's main menu. [That is, first click on **Version** in the main menu (not on the Version page of RunGTAP) and then, in the drop-down menu that appears, click on **Change**. We use | often in this document to indicate similar choices.]

Then click on **NCORS3X3** in the "Load version" list that appears, and then click on **OK**. Read the first few lines of the information about this version on the **Version** page of RunGTAP.

A.1 Examining the Data Directly using RunGTAP

There are three data files associated with each GTAP version. These are

The Global data set (I/O for each region, trade data etc.) [GTAPDATA]

Set information, giving region and commodity names in aggregation [GTAPSETS]

Parameter values [GTAPPARM]

⁴ These come from the preliminary release of version 4 of the GTAP data. This data, which is based on ACORS3x3 for use with the classic GTAP model, was first used in the African Short Course held in January 1998 in Johannesburg, South Africa.

The names GTAPDATA etc are those used in the TABLO Input file GTAPv7.TAB which lays down the theory of the GTAP model.

Example A1 – Viewing 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 **GTAPv7.TAB** which lays down the theory of GTAP. To see what information is on this global data file, select

View / TAB files / Main Model

[Here “Main Model” is the first of several menu options which appear beside “TAB files” once you click on it. You need to slide your mouse to the right after clicking on “TAB files” and let go of the mouse button once “Main Model” is highlighted.]

A text editor will appear, with GTAPv7.TAB visible. This will probably be GEMPACK’s **TABmate** editor⁵.

Now search for GTAPDATA: select **Search / Find...** from the TABmate menu (or Ctrl-F), then type in ‘gtapdata’, and finally click on **Search forward from top**. To search for more occurrences of “GTAPDATA”, either select **Search / Search again** from the editor’s main menu or press the **F3** key (near the top of your keyboard). Use this technique to see the first 3-4 occurrences of GTAPDATA.

You will see that various pieces of data are read from this file.

One of these is the array **VDPB(c,r)** whose values are held at header "VDPB". To see this and to find out what data is stored at this header and associated with VDPB(c,r) in GTAPv7.TAB, again use **Search / Find** to **Search forward from top** for "VDPB". You will see that VDPB(c,r) is declared as a **COEFFICIENT** and that it holds the Value of Private consumption expenditure on Domestic commodity **c** in region **r**, valued at **Basic** prices. By searching again for VDPB you will see that this data is read from file GTAPDATA at header "VDPB".

You can use TABmate’s **Gloss** feature to see all the places VDPB is used. Click to place the cursor within an occurrence of “VDPB”. Then click the “G” **Gloss** icon in the toolbar at the top of TABmate⁶. A window will appear, listing each line containing VDPB. You can click on the red line numbers to jump to that spot. Right-click on the Gloss window to close it.

Now close TABmate and return to RunGTAP.

To see the VDPB data in the 3x3 aggregation, click on

View / Base Data / Core Data

in RunGTAP’s menu [again “Core Data” appears to the side of “Base Data”]. RunGTAP calls the global data file the “Core Data”.

This will start the program **ViewHAR**⁷ running. ViewHAR will open the global data file for the NCORS3X3 version and show its contents on the Contents screen.

⁵ RunGTAP may be set up to use a simpler editor/viewer which is built-in to RunGTAP. These instructions assume that you are using **TABmate**. If necessary, use RunGTAP’s **Tools.|Options** command to make **TABmate** the default editor.

⁶ Ctrl-Right-click is the quickest way to launch the Gloss window, ie, right-click on the symbol while pressing Ctrl key.

⁷ ViewHAR is a program for viewing and modifying data held in GEMPACK Header Array files.

Each of the rows corresponds to a different array of data on the file. Look at the column under the heading Name to see what these arrays are⁸.

Header	Type	Dimension	Coeff	Total	Name
1 DVER	RE		1 DVER		5 version of GTAP data
2 DPSM	RE		REG DPSM		3 Share-weighted sum of distribution parameters
(etc.)					
13 VDPB	RE	COMM*REG	VDPB	15719091	Domestic purchases, by households, at basic prices
(etc)					

You can see that array number 12, at Header "VDPB", contains the value of households' domestic purchases at basic prices. To see the actual data, click twice on the VDPB row. What is the value of EU households' domestic purchases of manufactures at basic prices? What about purchases of food and agriculture by households in ROW? And for SSA? [The units are millions of 1995 US dollars.]

Now click on *Contents* in ViewHAR's main menu to return to the list of contents. Look at the VDFB array in row 11. You can see that this is a 3-dimensional array of size **COMM x ACTS x REG**, described as "Intermediates - Firms' Domestic Purchases at Basic Prices". To see the actual data, click twice on this VDFB row. You see a 3x3 matrix of data (plus a totals row and a totals column) with the rows labeled "Food", "Mnfc" and "Svces" and the columns labeled "Food", "Mnfc" and "Svces". These must be the elements of the sets COMM (commodities) and ACTS (activities) respectively. What about the 3rd dimension REG of this data? The clue is given by the 3 drop-down lists near the top right-hand corner of the screen which say

All COMM All ACTS Sum REG

respectively. Because your computer screen is 2-dimensional, what you are seeing are the VDFB values **summed across regions (REG)**. What is the total across all regions of the value at basic prices of domestic purchases of services by the food and agriculture firms? What about of manufactures by the capital-goods firms?

How can you see the value of firms' purchases in just one region, for example in SSA? To see this, click on the **Sum REG** drop-down list box near the top right-hand side and select SSA from the options. The data will change and now you are seeing how much is purchased just by firms in the SSA. What is the value at basic prices of purchases of services by manufacturing firms in the SSA? What about by manufacturing firms in the EU?

There are lots of ways of viewing 2-dimensional slices of a 3-dimensional array. To see another, click on the **COMM** drop-down list box (the first of the three) and select

Sum COMM.

Now click on the **REG** list box and select **All REG**. What does the number in row "food" and column "ROW" indicate? What is the total value (at basic prices) of all intermediate inputs into the services sector in the EU?

Recall that you are looking at the COMMxACTSxREG data at header "VDFB" in the global data set. To check what data is read from this header according to the description of the model in GTAPv7.TAB,

⁸ The arrays may be ordered differently from the list in the text. You can set ViewHAR up to either list the headers alphabetically, or to list them in the order they were written to file. ViewHAR's **File..Options** menu command can be used to customize the display in various ways.

find RunGTAP on your screen⁹, and select *View / TAB files / Main Model* from RunGTAP's main menu to open file GTAPv7.TAB. Search in it for **VDFB** until you find the lines saying which data is read from this header (it is the values of Coefficient **VDFB(c,a,r)** where **c** is the commodity, **a** is the activity (firm) and **r** is the region). Use *Search / Find* or Ctrl-F to *Search forward from top* for **VDFB**. You will see indeed that VDFB(c,a,r) is described as "purchases of domestic c for use in activity a in region r at basic prices".

ViewHAR is able to show you some shares, as well as the actual data. To see this, return to ViewHAR and then to its Contents page (click on *Contents*) and then look again at the **VDPB** data by clicking twice on its row. Notice the two drop-down list boxes near the top left-hand corner of the ViewHAR window. The first probably says **None** at present and the second shows a figure (between 0 and 6). The first of these is able to show row, column or matrix shares for the matrix on the screen. [We refer to it as the **shares list box**.]

In the shares list box select **Row**. You will see that the numbers change to give the row shares (with the total across each row being 1). Change this to **Col** and then **Matrix** and check what you get. Now return it to **None** so that you see the actual data again.

The second drop-down list box in the top left-hand corner controls the number of figures you see after the decimal point in each number. Try changing it to see this.

You can also inspect other headers. What is the value at basic prices of domestic manufactures ("Mnfc") destined for investment in the region ROW? [Click twice on header VDIB, select "ROW" in the REG list box.]

Now close the ViewHAR window by selecting *File / Exit* from ViewHAR's (not RunGTAP's) menu.

Binary files (such as Header Array files) cannot be looked at in a text editor. To see this, select *View / Other Text File* from RunGTAP's main menu and open BASEDATA.HAR (the Core data file) by typing this name "BASEDATA.HAR" in the "File name:" box provided. RunGTAP tells you that it is unable to open this as a text file (and even suggests that you try to open it with ViewHAR).

Example A2 - Viewing 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 **GTAPv7.TAB** which lays down the theory of GTAP. To see what information is on this set information file, select *View / TAB files / Main Model* to open file GTAPv7.TAB. Then search for **GTAPSETS**. You will see that

- the names of the regions (the SET is called **REG**) are held at header "**REG**",
- the names of the commodities (the SET is called **COMM**) are held at header "**COMM**", and
- there are several other SETs defined.

To see the actual data in the sets file for this version, return to **RunGTAP** and click on *View / Sets*. Again ViewHAR displays the Contents list, shown below.

⁹ One way of doing this is to click on RunGTAP in the Windows Taskbar which is probably along the bottom of your screen.

	Header	Type	Dimension	Name
1	DVER	RE	1	version of GTAP data
2	REG	1C	3 length 12	Set REG Regions
3	COMM	1C	3 length 12	Set COMM
4	MARG	1C	1 length 12	Set MARG
5	ACTS	1C	3 length 12	Set ACTS
6	ENDW	1C	5 length 12	Set ENDW
7	ENDM	1C	3 length 12	Set ENDWM Mobile endowments
8	ENDF	1C	1 length 12	Set ENDWF Sector-specific endowment
9	ENDS	1C	1 length 12	Set ENDWS Sluggish endowments

The data type "1C" means character data (i.e., lists of names). The "3 length 12" means that there are 3 names of (maximum) length 12 held at header "REG".

To look at the names at header "REG", double-click on this row in the Contents. What are the 3 regions called?

Return to the Contents (click on *Contents*) and look at the data at header "COMM". As you know from GTAPv7.TAB (look at it again if you are not sure) these are the names of the 3 commodities.

Close ViewHAR by selecting *File / Exit* from its menu.

Example A3 - Viewing the parameter values

Like the global data and set information, the parameters data for any GTAP data set are held on a Header Array file. From RunGTAP, you can see the parameter values by clicking on

View / Parameters .

Then examine **ESUBVA** at header "ESBV". You will see that there are 9 values (one for each activity in ACTS in each region). Look at these values. To find the significance of these values, you need to look in GTAPv7.TAB. To do that, return to RunGTAP¹⁰ and then click on *View / TAB files / Main Model* . Search there for **ESUBVA** and you will see that these 9 values are the elasticities of substitution between all 4 endowment factors (capital, labor etc) in the production of value added for each such activity in each region.

Similarly, you can look at the values of the other parameters via *View / parameters* and can always check on the use of the data by looking at the TABLO file for the model.

You can edit the parameters file within ViewHAR. To test this, again click *View / parameters* and examine the **ESUBVA** values. Right-click on the value for "food", "SSA" and type in a new value. Then use **File...Save as Header Array File** to save the modified name with a new name "special.prm". Alternatively use **File...Close** *without* saving changes.

Example A4 (optional) - Looking at the data in other GTAP versions

During the course you may also work with a 5-region, 7-commodity aggregation of the GTAP data to analyse a Free Trade Agreement between Sub-Saharan Africa and the EU. This is the version named

¹⁰ You can switch between ViewHAR and RunGTAP by using the Windows Task bar at the bottom of your screen.

NSA7X5.¹¹ Change to this version (via *Version / Change*) and look at the global data, the sets data and the parameters data for this version. [You can follow the methods in Examples A1-A3 above.] In particular, which regions and commodities are recognized in this aggregation of the data?

A.2 Using GTAPVIEW to report various aspects of the data

If necessary, start RunGTAP by double-clicking on its icon. Make sure that version NCORS3X3 is selected.

Example A5 – GTAPVIEW with the 3x3 data in NCORS3X3

GTPVEWv7.TAB¹² contains instructions to read certain parts of the global data, to compute various useful pieces of information from it, and then to write this useful information to another file. To see this, select *View / TAB files / GTAPView* from RunGTAP's main menu. Then, in the editor, search for the first 2 or 3 occurrences of GTAPDATA. You will see that this part of GTPVEWv7.TAB looks rather similar to the corresponding part of GTAPv7.TAB which you looked at earlier.

The information in GTPVEWv7.TAB is classified into 4 groups:

- Macro values
- Trade, Transport and Protection
- Commodities
- Sectors or Activities

You are going to look at the GDPEXP, GDPSRC and CURRENTACCT values in this Example. Search for GDPEXP, GDPSRC and CURRENTACCT so that you know what information these COEFFICIENTs will hold.

Switch to RunGTAP and select *View / Base Data / GTAPView Output*. This opens a ViewHAR window which displays the contents of the GTAPVIEW output.

(a) GDPEXP and GDPSRC values

The long names associated with the various arrays of data produced by GTAPVIEW make it clear that the GDPEXP values are at header **AG01** (the first row on the Contents page).

To see the GDPEXP values, click twice on this **AG01** row in the Contents page. What is the value of GDP from the expenditure side in Sub-Saharan Africa? How much of this comes from household consumption, how much from exports? What about in the EU? Leave this ViewHAR window open.

To see how some of these are calculated, switch back to GTPVEWv7.TAB in TABmate¹³. First search for **GDPEXP** and find the formula for **GDPEXP(r,"cons")**; you can see that the "**cons**" values come from the values of Coefficient **PRIVEXP**. Search from the top of GTPVEWv7.TAB for **PRIVEXP** to see how these values are calculated. [You might like to repeat this for some of the other parts of **GDPEXP**.] Leave GTPVEW.TAB open in TABmate.

Switch back to the ViewHAR window to see what the value of total GDP from the expenditure side in Sub-Saharan Africa.

¹¹ This version contains the same data found in ASA7X5 for use with the classic GTAP model.

¹² Following GTAP tradition, we refer to the *GTAPVIEW* program, even though the RunGTAP file containing its source code is now named *GTPVEWv7.TAB*. Pronounce "GTPVEW" the same as "GTAPVIEW" with identifier v7 for GTAP model version 7.

¹³ If necessary, you can re-open the file via *View / TAB files / GTAPView* in RunGTAP's main menu.

Then click on **Contents** (or alternatively, double-click anywhere in the data part of the ViewHAR window) to return to the Contents list.

Next we want you to look at the values for GDP from the income side (denoted by GDPSRC). How much of this in Sub-Saharan Africa comes from factor income? How much from taxes? What is total GDP from the income side in Sub-Saharan Africa?

Does total GDP from the income side in Sub-Saharan Africa equal total GDP from the expenditure side? What about in the EU?¹⁴

(b) *CURRENTACCT* values

Recall that the formula for calculating CURRENTACCT is (look again in GTPVEWv7.TAB if necessary):

Formula (all,r,REG)(all,m,MARG)
CURRENTACCT(r,"exp",m) = sum{d,REG, VFOB(m,r,d)} + VST(m,r);
Formula (all,r,REG)(all,c,NMRG)
CURRENTACCT(r,"exp",c) = sum{d,REG, VFOB(c,r,d)};
Formula (all,r,REG)(all,c,COMM)
CURRENTACCT(r,"imp",c) = - sum{s,REG, VCIF(c,s,r)};

Thus the CURRENTACCT(r,"exp",c) values are obtained by summing the VFOB(c,r,d) values across the importing or destination regions d and then adding transport costs. [The formula is slightly different if the commodity is a margins commodity – see the first formula above.] The CURRENTACCT(r,"imp",c) values are the negative of the VCIF(c,s,r) values summed across the exporting or source regions.

To look at the CURRENTACCT values, return to the **Contents** list (in the ViewHAR window) and then double-click on the row at header **AG05**. How much do exports of all commodities contribute to the current account in Sub-Saharan Africa? By how much do imports of all commodities reduce this current account? What is the current account balance for Sub-Saharan Africa in the base data? What about for the EU?

How much do exports of food and agriculture contribute to the current account in Sub-Saharan Africa? [To see this you will need to change the Sum/All boxes near the top right-hand corner of ViewHAR. Currently they probably¹⁵ read

All REG All CURACCT Sum COMM

One way of altering these to see the desired result is to change **All CURACCT** to **exp** and then to change **Sum COMM** to **All COMM**.] What about exports of manufactures? By how much do imports of food and agriculture reduce the current account in Sub-Saharan Africa? What about in the EU?

Now exit from ViewHAR, TABmate and RunGTAP (in each case, use *File / Exit*, or just click on the **X** in the top right-hand corner).

¹⁴ In EU, total GDP is shown as 8208664.0 from the income side and 8208662.0 from the expenditure side. GEMPACK only stores data accurately to about 6 or 7 significant figures; hence the small difference in these two figures.

¹⁵ For 3-dimensional matrices, ViewHAR uses your previous choices to anticipate which slice or subtotal of the matrix you would like to see.

Example A6 (optional) - GTAPVIEW with other data

Start RunGTAP running. Change to the **NSA7X5** version (via *Version / Change*) and look at the GDPEXP, GDPSRC and CURRENTACCT values based on this 5-region, 7-commodity aggregation of the GTAP data. For example, the GDPEXP values here for EUNION should be (essentially) the same as with the 3x3 data in Example A5 for EU. But now you can see the separate values for regions SAFRICA (South Africa), RESTSAF (Rest of Southern Africa) and RESTSSH (Rest of Sub-Saharan Africa) which are combined into SSA (Sub-Saharan Africa inclusive of South Africa) in the 3x3 data (as in Example A5). Exit from ViewHAR and RunGTAP.

Summary: Data and Consequences of the Data

When you want to find out something about a particular GTAP data set (or version), you have the choice of looking at the data directly using *View / Base Data / Core Data* (as in section A.1 above) or looking at the GTAPVIEW summary of the data using *View / Base Data / GTAPView Output* (as in section A.2 above). You will look at the *Core Data* if you want to look at something held on the actual data base (for example, the VDPB values) but will need to use the GTAPVIEW summary if what you want to know has to be calculated from the actual data (for example, the value of GDP in each region).¹⁶

Sometimes you need to know about consequences of the data which are not shown in the GTAPVIEW summary. Then you may need to write your own formulas to calculate the numbers you need. This is the idea behind section A.3 below.

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

The next sections require a GEMPACK licence. If you do not have one, skip on to Section B.2, alternatively read through the next sections without attempting to follow them with your computer.

Sometimes you need to know about consequences of the data which are not shown in the GTAPVIEW summary. Then you may need to write your own formulas to calculate the numbers you need. We show you how to do this here.

You need to use GEMPACK proper to build and run your own TAB files. We show how to do this via **WinGEM**, the Windows interface to GEMPACK.

In the rest of this document, we assume that you have already extracted all the GTAP files from the course USB¹⁷ and put them into directory C:\GTAP. This includes all the files in the files **AGG4P04.ZIP**, **CMF4P04.ZIP** and **AGG4P05.ZIP**. If at any stage you inadvertently delete some of these files, you can get them back by extracting them again from the relevant ZIP archive. For example, you can restore GTAPVIEW.TAB from the zip archive **CMF4P04.ZIP**.

Starting WinGEM

Double click on the **WinGEM** icon to start GEMPACK for Windows.

This should give a menu across the top of the screen :

¹⁶ Note that ViewHAR can also show some shares (see the end of Example A2). Strictly speaking these shares are consequences of the data.

¹⁷ If not present on the CD/USB key, you can download the examples from the end of the RunGTAP patch page, at: <http://www.copsmodels.com/rgtpatch.htm>

GEMPACK for Windows

File Simulation HA files Other tasks Programs Options Window Help

Setting the working directory for WinGEM

Before starting these examples, you should have a directory containing all the GTAP files. In this document we assume that directory `\GTAP` on C: drive is such a directory. [If yours has a different name, or is on a different drive, make appropriate changes as you read this document.]

First choose

File / Change both default directories

[By this notation we mean first click on *File* in the WinGEM menu. This will produce a drop-down menu. In the drop-down menu, click on the menu item

Change both default directories]

In the file selection box that appears, choose drive **C:** and **double-click** on the subdirectory **GTAP**. [You should see **C:\GTAP** in blue.] Click on the OK button.

Carrying Out Calculations in TAB Files

You have seen in section A.2 above how formulas in the TAB file GTAPVIEW.TAB lead to values calculated and reported in the GTAPView Output available via RunGTAP. There are two steps required in going from the TAB file¹⁸ to the data files showing the numbers.

1. Run the GEMPACK program TABLO. This converts the formulas in GTAPVIEW.TAB to a computer version. [GTAPVIEW.TAB can be read by humans and by the computer. The so-called **GEMSIM Auxiliary files** produced by TABLO can only be read by the GEMSIM program.]
2. Run the GEMPACK program GEMSIM. This starts from the GEMSIM Auxiliary files created by TABLO. GEMSIM carries out the calculations in the various formulas in GTAPVIEW.TAB and assembles the results in the GTAPView Output file. When you run GEMSIM, you must tell it which formulas to carry out (that is, which TAB file the Auxiliary files come from) and which Core GTAP data to base the calculations on.

These two steps with GTAPVIEW.TAB have been carried out for you (by the GTAP staff) when you use RunGTAP. In this section you will learn how to carry out these steps for yourself, starting from a file which is similar to part of GTAPVIEW.TAB.

Example A7 - Completing TABLO Input file GVIEWX.TAB

Write a TABLO Input file GVIEWX.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 commodity in each region, valued at basic prices,
 - (b) the share of domestic production of each commodity in each region used by private households,

¹⁸ The more complete name is "TABLO Input file", which is what these files are called in the GEMPACK user documentation. Since we are not in RunGTAP now, the file is called GTAPVIEW.TAB, not GTPVEWv7.TAB.

(c) the total value of sales of commodities in each region, valued at basic prices.

(iii) WRITES to write these values to a new (Header Array) file with logical name GVIEWX.

An incomplete version VWX.TAB of such a TABLO Input file should be in directory in which you are working. A printed copy of this incomplete file appears below.

The steps for completing this example are split into two shorter examples below. These examples correspond to the two steps (run TABLO, run GEMSIM) above.

```
!_____ Incomplete GVIEWX.TAB FILE_____!
!-----!
!                               FILES, SETS and SUBSET                               !
!-----!
FILE GTAPSETS # File with set specification #;
SET REG # Regions in the model #
  READ ELEMENTS FROM FILE gtapsets HEADER "REG";
SET COMM # COMMODITIES #
  READ ELEMENTS FROM FILE gtapsets HEADER "COMM";
SET ACTS # PRODUCED COMMODITIES #
  READ ELEMENTS FROM FILE gtapsets HEADER "ACTS";
FILE GTAPDATA # The file containing all base data. # ;
!-----!
!   base revenues and expenditures at producer _ s prices           !
!-----!
COEFFICIENT (all,c,COMM)(all,a,ACTS)(all,r,REG)
  VDFB(c,a,r)
  # purchases of domestic c for use in a in region r at basic prices # ;
COEFFICIENT (all,c,COMM)(all,r,REG)  VDPB(c,r)
  # private household expenditure on domestic C in r at basic prices # ;
COEFFICIENT (all,c,COMM)(all,r,REG)  VDGB(c,r)
  # government expenditure on domestic c in r at basic prices# ;
COEFFICIENT (all,c,COMM)(all,r,REG)  VDIB(c,r)
  # investment expenditure on domestic c in r at basic prices # ;
!-----!
!   Reading basedata.                                           !
!-----!
READ (all,c,COMM)(all,a,ACTS)(all,r,REG)  VDFB(c,a,r)
  FROM FILE GTAPDATA HEADER "VDFB" ;
READ (all,c,COMM)(all,r,REG)  VDPB(c,r)
  FROM FILE GTAPDATA HEADER "VDPB" ;
READ (all,c,COMM)(all,r,REG)  VDGB(c,r)
  FROM FILE GTAPDATA HEADER "VDGB" ;
READ (all,c,COMM)(all,r,REG)  VDIB(c,r)
  FROM FILE GTAPDATA HEADER "VDIB" ;
!-----!
!   DERIVATIVES OF THE BASE DATA                               !
!-----!
COEFFICIENT (all,c,COMM)(all,r,REG)  VDB(c,r)
  # domestic sales of commodity c in region r, basic prices # ;
FORMULA (all,c,COMM)(all,r,REG)
VDB(c,r) = VDPB(c,r) + ???(c,r) + VDIB(c,r) + sum{a,ACTS, VDFB(c,a,r)} ;
COEFFICIENT (all,c,COMM)(all,r,REG)  PDCSHR(c,r)
  # the share of domestic production used by private hhlds # ;
FORMULA (all,c,COMM)(all,r,REG) PDCSHR(c,r) = ???(c,r)/VDB(??) ;
COEFFICIENT (all,r,REG)  TOTVDB(r)
# Total dom sales of commodities in region r valued at basic prices # ;
FORMULA (all,r,REG)  TOTVDB(r) = SUM( c, COMM, ???) ;
```

```
FILE (New) GVIEWX # File with headers for viewing # ;  
Write VDB    to File GVIEWX Header "VDB" ;  
Write PDCSHR to File GVIEWX Header "PDCS" ;  
! Note that headers are restricted to 4 characters !  
Write TOTVDB to File GVIEWX Header "???" ;
```

Example A8 – Running TABLO to Process GVIEWX.TAB

Choose *File / Edit file...* from WinGEM's main menu, and open VWX.TAB. Choose *File / Save as...* to save this file under the name **GVIEWX.TAB**. Then make appropriate additions and/or changes to complete the file. [Search for ? to find places where changes are needed.] When you have made all the changes, exit from the editor via *File / Exit* . Say **Yes** when asked if you want to save the changes.

You need to run TABLO to process this file. To do this, open a TABLO program window by clicking on

Simulation / TABLO Implement...

in WinGEM's main menu.

Now click on *Select* in the resulting TABLO program window and open your file **GVIEWX.TAB** which is the TABLO Input file you wish to process. Before clicking on the "Run" button, you need to ensure that TABLO generates GEMSIM (not Fortran) code. Select the GEMSIM option at top right of TABLO program window.

Then start TABLO running by clicking on *Run*.

When TABLO runs it will check the instructions in this file to see if they seem consistent. If TABLO finds inconsistencies (which it impolitely calls "errors"), it will report this after it has finished running.

When TABLO finishes running, if it finds one or more errors (as it is likely to do since we have made a deliberate error in the file we have given you), in the window which appears to indicate an error, select *Edit TABLO file*. This will put you into the program **TABmate** with your TAB file GVIEWX.TAB loaded.

In this TABmate window, errors are underlined in red. If you click on an error, the reason for the error is shown in red near the bottom of your screen. You can make changes on the screen. When you think you have removed an error, click on the **Check** button near the top middle of your screen. This causes TABLO to check your TAB file again. Keep repeating this until you have removed all errors. When you have removed all errors, exit from TABmate. [Each time you click on the Check button, TABmate saves the changes so you should not need to save them again before exiting from TABmate.]

Warning. We have made a deliberate error in the TABLO Input file GVIEWX.TAB which will cause several consequential errors. Once you identify and fix this error, the consequential errors will also be eliminated. **[Hint:** If you have trouble finding our deliberate error, look closely at the spelling of the set COMM in the declaration of the COEFFICIENT VDB(c,r), where the first error is.]¹⁹

To get back to WinGEM, click on *Re-Run* to close the "Error running TABLO" window and to rerun TABLO inside WinGEM. This time there should be no errors and so you will be offered a button *Go to GEMSIM* . Click on this and jump to Example A9 below. [This completes the first of the two steps under the heading "Carrying Out Calculations in TAB files" above.]

Example A9 – Running GEMSIM to do the Formulas (etc) in GVIEWX.TAB

When the TABLO Input file is correct and you have prepared GEMSIM Auxiliary files, you need to run GEMSIM. After you click on *Go to GEMSIM* (see the end of Example A8 above), you see a GEMSIM program window.

¹⁹ If you are uncertain about your TABLO Input file GVIEWX.TAB, you can compare it with the file GVIEWXOK.TAB which we have prepared and which should be in the directory in which you are working.

In this GEMSIM window, click on *Select* and select Command file **VWX4P04.CMF** .

Before running, click on *Edit* to look at the contents of this Command file VWX4P04.CMF. This is shown below.

```
! _____vwx4p04.cmf_____
!  
! Auxiliary files (where the formulas etc come from)  
!  
auxiliary files = gviewx ;  
!  
! Input data files  
!  
file gtapsets = set4p04.har;  
file gtapdata = dat4p04.har;  
!  
! Output File Specification  
!  
file gviewx = vwx4p04.har ;  
!  
Log File = vwx4p04.log ;  
!  
! _____End of file._____
```

You can see that this file instructs GEMSIM to

- use Auxiliary files GVIEWX (which are the ones produced by TABLO in Example A8 above), which means that GEMSIM will carry out the instructions in your file GVIEWX.TAB,
- to read GTAPDATA and GTAPSETS files called DAT4P04.HAR and SET4P04.HAR (these are the global data and set information files for the 3x3 aggregation you looked at in section A.1 above),
- to write the “useful information” which GVIEWX.TAB says is to be written to (logical) file GVIEWX to the actual file called VWX4P04.HAR, and
- to write all program output to a LOG file to be called VWX4P04.LOG.

At last it is time to actually run GEMSIM! To do this, close the file VWX4P04.CMF via TABmate’s *File /Exit* . This will return you to the open GEMSIM program window. Then click on *Run*. GEMSIM will run in a DOS box.

When GEMSIM finishes running, you will want to look at the results of the formulas (etc) in GVIEWX.TAB. To do this, click on *View Input/Output Files* and you will see three choices. Click on the last one which says

HA output GVIEWX c:\gtap\vwx4p04.har

and then click on *Ok* (or double-click on the above line). This starts ViewHAR running and puts you into the **Contents** page of the relevant file. Have a look at the VDB, PDCSHR and TOTVDB values. Make sure that you can find all three lots of results (a),(b),(c) (in Example A7 above) as requested.

You may also like to look in more detail at how GEMSIM reports the calculations it carried out. If so, return to the GEMSIM windows and click on *View log file*. Scroll through this log file (and search if necessary) to note the following things.

- Near the top of the file, see where GEMSIM has read the instructions from the Command file VWX4P04.CMF (see above).

- See where GEMSIM has read the names of the elements of various sets (for example, read the 3 elements of COMM from header "COMM").
- See where GEMSIM has read various pieces of data (for example "Reading VDFB").
- See where GEMSIM said it has written the values of various COEFFICIENTs (for example VDB). To which headers has it written the VDB and the PDCSHR values?
- See where GEMSIM has opened the actual file associated with the logical name GVIEWX. (Search for "**logical name**". What is the actual name of this file and where did this name come from?).

Exit from this LOG file which should put you back to the GEMSIM program window. Then exit from WinGEM by selecting *File / Exit* from WinGEM's main menu (not from the GEMSIM window). This will also close the open TABLO and GEMSIM windows.

Example A10 (optional) – The GVIEWX results for a different version

In the examples above, you calculated the GVIEWX results for the 3x3 aggregation known as NCORS3X3. Here you will do the same for the 5-region, 7-commodity version known as NSA7X5. The data files for this version are usually given the names **DAT4P05.HAR** (GTAPDATA file), **SET4P05.HAR** (GTAPSETS file) and **PAR4P05.DAT** (GTAPPARM file). Here the "**4P05**" is used to indicate that this is aggregation number 5 from the preliminary release of version 4 of the GTAP data.

Since the formulas in GVIEWX.TAB are the same as in the example above, you do not need to rerun TABLO. But you do need to run GEMSIM since that is the program which does the arithmetic.

Start WinGEM running. To run GEMSIM, you need to modify the Command file VWX4P04.CMF used in Example A9 above. To do this, edit this file VWX4P04.CMF via *File / Edit file* from WinGEM's main menu. First use *Save As...* to save this file as **VWX4P05.CMF** (since you will now work with the files with "4P05" in their names). Then make the appropriate changes to VWX4P05.CMF. [These involve changing every occurrence of "4P04" to "4P05". Do not change the "auxiliary files" line since the TAB file is still GVIEWX.TAB.] When you have made these changes, save the new file and exit from the editor.

Now you are ready to run GEMSIM. To do this, select *Simulation / GEMSIM Solve* from WinGEM's main menu. Then *Select* the Command file VWX4P05.CMF. Then click on *Run*. When GEMSIM finishes, look at the output file produced to see the VDB, PDCSHR and TOTVDB values for this version of the GTAP data. Exit from WinGEM when you have finished.

A.4 Finding the Size of Existing Distortions

In Example A11 below, you will write a TABLO Input file SHOCKSX.TAB which can calculate tariff distortions in a GTAP data set and calculate what shocks to give to the model in order to ensure that no ad valorem tariff exceeds 5%. First we give some background.

4.1 Background and Notation

IMPORT TARIFFS/SUBSIDIES

VMSB Imports values at domestic basic prices - i.e., inclusive of any tariff/subsidy

VCIF Imports values at world prices (cif)

TMS_L Power of the intervention

Formula (you complete it): $TMS_L =$

If $VMSB > VCIF$ (i.e. $TMS_L > 1$) there is an **import tariff**.

If $VMSB < VCIF$ (i.e. $TMS_L < 1$) there is an **import subsidy**.

If $VMSB = VCIF$ (i.e. $TMS_L = 1$) there is **no distortion**.

Example

$VMSB = 120$ $VCIF = 100$ $TMS_L = 1.2$ (tariff)

To remove the tariff, must decrease TMS_L from 1.2 to 1.

In this case we need to reduce the power from 1.2 to 1.05 (5% ad valorem rate)

General formula for percent changes: $\% = 100 * (NEW - OLD) / OLD$

Percent change required to make TMS_L equal to 1.05 is $100 * (1.05 - 1.2) / 1.2 = -12.5\%$

In the model GTAPv7.TAB, the VARIABLE '**tms**' is the percentage change in TMS_L .

Hence we must shock the linear variable '**tms**' by -12.5%

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

Example A11 - 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 import distortions (that is, TMS_L values),

(b) shocks to percent-change variable '**tms**' to ensure that no ad valorem tariff exceeds 5%.,

(iii) WRITE statements to write the tms shocks to a HAR file.

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

Suggested steps for carrying out this example are set out on the following page.

```

! _____Incomplete SHOCKSX.TAB FILE _____!
File GTAPSETS # File with set specification #;
File GTAPDATA # The file containing all base data. #;
Set
  REG      # Regions          # read elements from file GTAPSETS header "REG";
  COMM     # Traded commodities # read elements from file GTAPSETS header "COMM";
Coefficient
(all,c,COMM)(all,s,REG)(all,d,REG) VMSB(c,s,d)
  # Value of imports of c from s to d at domestic (basic) prices #;
(all,c,COMM)(all,s,REG)(all,d,REG) VCIF(c,s,d)
  # Value of imports of c from s to d at CIF prices (tradeables only) #;
Read
  VMSB from file GTAPDATA header "VMSB";
  VCIF from file GTAPDATA header "VCIF";
! Calculate Powers of tariff and percent changes to cap tariffs at 5% rate !
Coefficient
(all,c,COMM)(all,s,REG)(all,d,REG) TMS_L(c,s,d)
  # Initial source-spec. power of tax on imports of c by d from source s #;
(all,c,COMM)(all,s,REG)(all,d,REG) TMS_T(c,s,d)
  # Target power of tax on imports #;
(all,c,COMM)(all,s,REG)(all,d,REG) TMS_HAT(c,s,d)
  # Shock to remove source-spec. tax on imports of c by d from source s #;
Zerodivide default 1;
Formula
(all,c,COMM)(all,s,REG)(all,d,REG) TMS_L(c,s,d) = VMSB(c,s,d) / VCIF(c,s,d);
! ensure ad valorem rate cannot exceed 5% !
(all,c,COMM)(all,s,REG)(all,d,REG) TMS_T(c,s,d) = MIN(1.05, TMS_L(c,s,d));
! shock is % difference between target and initial power !
(all,c,COMM)(all,s,REG)(all,d,REG)
  TMS_HAT(c,s,d) = 100*[TMS_T(c,s,d) - TMS_L(c,s,d)] / TMS_L(c,s,d);
File (new) TMSHAT # The file with shocks to tms #;
Write TMS_L to file TMSHAT header "TMSL";
Write TMS_T to file TMSHAT header "TMST";
Write TMS_HAT to file TMSHAT header "TMSH";

```

Suggested steps for Example A11

You will use WinGEM, TABLO and GEMSIM for this example.

(1) Getting started.

Choose *File / Edit file...* (from WinGEM's menu) and open SHKX.TAB. Choose *File / Save as...* to save this file under the name **SHOCKSX.TAB** . Then make appropriate additions and/or changes to complete the file.

(2) Fixing the TABLO Input file.

You need to run TABLO to process this file, which you can do by opening a TABLO program window (via *Simulation / TABLO Implement...*) and selecting TABLO Input file SHOCKSX.TAB.

When running TABLO be sure to select option **PGS Generate GEMSIM Auxiliary files** as you did in Example A8 above. You do this by selecting *Options / TABLO options* from the TABLO program window.

Run TABLO as in Example A8 above. If TABLO finds one or more errors, proceed as in Example A8 to correct them and rerun TABLO until no more errors remain. Then go on to step (3) below.

[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 the directory you are working in.]

(3) Preparing the Command file.

(i) We have prepared an incomplete Command file **SHKX.CMF**. First save this under the name **SHKX4P04.CMF** using *File / Save as...*

(ii) Now Edit this file **SHKX4P04.CMF** to replace each ? in it.

(iii) SHOCKSX.TAB contains instructions to write certain coefficients to a HAR file. Note that the statement:

```
File tmshtat = tmshtat.har ;
```

specifies the name of this HAR file.

(v) Exit from the editor GEMEDIT, saving the changes.

[If you are uncertain, you can compare your file SHKX4P04.CMF with the file SHKXOK.CMF we have prepared.]

(4) Run GEMSIM to carry out the calculations in SHOCKSX.TAB.

To do this, open a GEMSIM window and *Select* the Command file SHKX4P04.CMF you prepared in (3) above.

If GEMSIM reports that there is an error in your Command file, use the split screen editing offered to find and fix the error.

When GEMSIM runs, it will produce a HAR file TMSHAT.HAR.

(5) Looking at the HAR file produced.

The output HAR file contains initial and final values for powers of tariffs, and the percent difference between final and initial values.. Choose *View Input/Output Files* to see these values.

Are there any import tariffs? Any import subsidies? Any examples of no import distortion? At shock header TMSH, are any tms values increased?

(6) Using your new shock file

Copy TMSHAT.HAR to your RunGTAP NCORS3x3 version folder [probably C:\RunGTAP\NCORS3x3]. Then start RunGTAP and check that you using the NCORS3x3 version. Move to the Shocks page and click 'Clear Shocks List'. Then carefully type into the Shocks box:

```
Shock tms = select from file TMSHAT.HAR header "TMSH";
```

Move to the Solve page and select the Gragg 3-5-7 solution method. Than 'Save Experiment' as TARF with description "cap tariffs at 5% ". Then click Solve, and study the results

If you wished to shock only tariffs imposed by the EU, you could type into the Shocks box:

```
Shock tms(COMM,REG,"EU") = select from file TMSHAT.HAR header "TMSH";
```

Example A12 - Compute all distortions and the shocks needed to eliminate them

In SHOCKSX.TAB, you calculated just one of the possible distortions. The TABLO Input file SHOCKSv7.TAB (which is a standard GTAP TABLO Input file²⁰) calculates these and most other distortions, and writes the shocks required to eliminate them to files like TOHAT.SHK and TMSHAT.SHK in Example A11.

To carry out all the calculations in SHOCKSv7.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 A11 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 SHK4P04.CMF. This produces several different .SHK files containing the values of the shocks to appropriate variables of the GTAP model required to eliminate the different distortions.

For example, look (via *View Input/Output Files*) at the shocks to remove all output subsidies or taxes. [This is the file with logical name TOHAT.] Can you tell for which commodities and which regions there is an output tax? What about output subsidy?

Summary of Part A. Data and Consequences of the Data

When you want to find out something about a particular GTAP data set (or version), you have the choice of looking at the data directly using *View / Base Data / Core Data* (as in section A.1 above) or looking at the GTAPVIEW summary of the data using *View / Base Data / GTAPView Output* (as in section A.2 above). You will look at the Core Data if you want to look at something held on the actual data base (for example, the VDPB values) but will need to use the GTAPVIEW summary if what you want to know has to be calculated from the actual data (for example, the value of GDP in each region).²¹

Sometimes you need to know about consequences of the data which are not shown in the GTAPVIEW summary. Then you may need to write your own formulas to calculate the numbers you need. If so, you can use WinGEM and the GEMPACK programs TABLO and GEMSIM as in Examples A8 and A9 above to calculate and report the numbers in question.

Of course, the GTAP staff have used WinGEM, TABLO and GEMSIM to calculate the values reported by GTPVEW.TAB when preparing the files assembled with RunGTAP for your use. They have done this for each of the different versions supplied with RunGTAP.

They have also carried out the calculations in the important GTAP TAB file SHOCKSv7.TAB (see Example A12 above) for each version supplied with RunGTAP.

²⁰ The file SHOCKS.TAB supplied in CMF4P04.ZIP is the 2018 version. The current version of SHOCKSv7.TAB used under RunGTAP in the current Short Course may be different. However the two versions are implemented and run in the same way.

²¹ Note that ViewHAR can also show some shares (see the end of Example A2). Strictly speaking these shares are consequences of the data.

B. Simulations

B.1 Condensing the Model

The theory of the GTAP model as used in this course is contained in the TABLO Input file which you can view under RunGTAP by selecting *View TAB files / Main Model* . This file is called GTAP18.TAB in CMF4P04.ZIP.

In order to solve the model on a PC with limited RAM memory, it is necessary to condense it -- this reduces the size of the equation system, which in turn reduces the amounts of memory and time required to solve the model. More details about condensation of models can be found at <http://www.copsmodels.com/gpmanual.htm#condense>

You can view the condensation file under RunGTAP by selecting *View / TAB files / Condensation* [This condensation file is very similar to the file called GTAP18GS.STI in CMF4P04.ZIP.] In the condensation in GTAP18GS.STI (which is very similar to the condensation used in the current Short Course),

(a) certain policy variables²² have been omitted. This means that these variables are effectively exogenous but cannot be shocked. (When you work with this condensation, it is as if these variables had never been present in the model.). At the beginning of this file you will find which are the variables that are being omitted as unnecessary policy variables.

tf tpm tpd tgm tgd tfm tfd tid tim atall avaall

(b) certain variables have been selected for backsolving -- they are hard-wired as endogenous. When you run a simulation, you can obtain results for these variables. However, they cannot be set exogenous once this condensation has been carried out. The variables which are backsolved for are:

pfd compvalad af ao ava afa ppm pfm pms pfob pcif pfa ppd pgm pgd qfm pva qfe qva qfa pgov qga pga qgm qpa qpm qtmfsd ptrans atmfsd

You can look in GTAP18.TAB to find out more about these variables.²³

There are two possible computer versions of this condensation of GTAP01.

- (i) The **GEMSIM Auxiliary files** version which consists of the two GEMSIM Auxiliary files **GTAP18.GSS** and **GTAP18.GST**. [These can be produced by running TABLO taking inputs from the Stored-input file GTAP18GS.STI. This file includes the same instructions to omit and backsolve as indicated above.] This is the version you will work with during the course and which you will all take away with you (see Example B1 below).
- (ii) The **TABLO-generated program** version, which consists of the **executable image GTAP18.EXE** and the two **Auxiliary files GTAP18.AXS** and **GTAP18.AXT**. This is the version you may wish to produce when you return home from the course if you have a source-code version of GEMPACK. [These can be produced by running the program TABLO taking inputs from the Stored-input file

²² Of course one day you may wish to shock one of these variables ! RunGTAP includes a second version of the GTAP model, named GTAPUv7, which omits fewer variables (but uses more memory).

²³ Some of the variables in the Welfare and Terms-of-Trade decompositions are backsolved for or substituted out in the condensation of GTAP used in this course.

GTAP18.STI and then running LTG. The file GTAP18.STI includes instructions to omit and to backsolve as indicated above.]²⁴

Example B1 (optional) - Condensing GTAP to produce output for GEMSIM

Select *Simulation / TABLO Implement...* from WinGEM's main menu. This will open a TABLO window. Next select *Options / Run from STI file* from the menu on this TABLO window. Then *Select* the Stored-input file **GTAP18GS.STI**.²⁵ [This is a so-called Stored-input file, often abbreviated to STI file. It contains all the responses to prompts from the program TABLO in order to condense the model (omissions, backsolves etc as described above) and produce output for GEMSIM.] Then click on *Run*. TABLO will take a couple of minutes to run. When it has finished you might like to look at the LOG file produced.²⁶

Simulations with GTAP via RunGTAP

Supplied with RunGTAP is the TABLO-generated program GTAPv7.EXE and Auxiliary files GTAPv7.AXS and GTAPv7.AXT [see (ii) above]. This has been produced by the GTAP staff using WinGEM and TABLO. To see which condensation actions (omissions and backsolves) were used, select *View / TAB files / Condensation* from RunGTAP's menu.

All the remaining examples in Part B below involve simulations with GTAP. Most are with the 3x3 data but some are with other GTAP data sets.

The next sections DO NOT require a GEMPACK licence.

²⁴ The program GTAPv7.EXE used to solve GTAP under RunGTAP (see later in this section) was produced by following these steps.

²⁵ Another way to select "run from STI file" more quickly is to click on the "tape recorder" button along the top row in the TABLO window. This puts you straight into a file selection box for STI files. [If you "hover" your mouse - that is, leave it still for a few seconds - over this tape recorder button you will see the hint "Select STI file (Ctrl+T)". There are other so-called "shortcut" buttons in WinGEM; you might like to hover over some of them and use ones which appeal to you.]

²⁶ If you have a source-code version of GEMPACK, you can proceed similarly except using Stored-input file GTAPv7.STI instead of GTAPGS.STI, to produce the TABLO-generated program GTAP.FOR. Then you can compile and link this (via WinGEM's *Simulation / Compile & Link...*) to produce GTAPv7.EXE. You can carry out simulations with GTAP by running GTAPv7.EXE instead of GEMSIM via WinGEM's *Simulation / Run TG program...*. RunGTAP uses GTAPv7.EXE to carry out simulations.

B.2 Numeraire simulation with the 3x3 data

This is a simulation in which the usual numeraire for GTAP (namely the variable **pfactwld** - world price index of primary factors) is increased by 10 per cent.

Example B2 - Carrying out the basic numeraire simulation

Start RunGTAP running and make sure that the NCORS3X3 version is selected. You will see a screen as below:

RunGTAP: Ncors3x3/GTAP

File Copy View Version Tools Help

Title RunGTAP Version Closure Shocks Solve Results

For simulations, the last four pages **Closure - Shocks - Solve - Results** are usually accessed in that order (from left to right).

First click on the **Closure** page of RunGTAP. You will see a list of the exogenous variables in the currently selected closure. You can choose between several different closures when running a simulation. The panel below shows which variables are exogenous in the selected closure. RunGTAP allows you to load different closures already prepared.²⁷ To see this, use the **Tools / Load Closure File** menu command to load the **book** closure file. Notice the different exogenous list. In this simulation we want you to use the standard closure for GTAP, namely the closure referred to as “Standard”. To use this one, again use the **Tools / Load Closure File**, and this time select **Standard**. Notice that **pfactwld** is one of the exogenous variables.

Now click on the **Shocks** page. Click on the **Clear Shocks List** button to remove whatever shocks are shown. Now you will specify a shock to variable **pfactwld**. To do this, click on the **down arrow** to the right of the label **Variable to shock** near the top of the Shocks page. A drop-down list of all the exogenous variables in this standard closure will appear. Click on **pfactwld**. A new edit box entitled **% Change Shock** will appear. Click in this and type in **10**. Then click on the button **Add to Shock List**. The line

Shock pfactwld = 10;

should appear in the Shocks memo which occupies the bottom half of the Shocks page. This is the only shock for the numeraire simulation.

Now click on the **Solve** page of RunGTAP.

1. First click on the topmost **Change** button (the one with **Solution method** before it). Click on **Johansen** to select Johansen’s method. [You will learn more about the different solution methods available later in this document.]
2. Make sure that **Parameter file: default** is shown to the left of the second **Change** button. [This means that you are using the default, or normal, parameter values.]²⁸
3. If it is not already there, insert the text **Numeraire simulation** in the text box labelled **Experiment Description**.

²⁷ You can also edit the closure shown on the Closure page, and can save this edited closure.

²⁸ If a different parameter file is selected, click on the Change button and select DEFAULT.

4. Next save an experiment file containing a full specification of your simulation. This will allow you to repeat the simulation later if necessary, and also avoids annoying messages reminding you to save experiment details. Click the **Save Experiment** button and type in "NUM2" as the new file name. Then click OK.
5. Now click on the **Solve** button. A "Please Wait" window will appear while the model is solved. When the solution has been obtained, RunGTAP will present you with a box telling you how long it took to obtain the solution. Just press OK.

The next natural step is to look at the results which you will do in Example B3 below.

Example B3 - Look at the Results

Click on the **Results** page of RunGTAP. You will see the **Contents** page listing many of the variables of the model.

To see the results of one of these variables listed by name, just **double-click** on the corresponding row in the Contents list. Find the **qo** row (you will have to scroll down a fair way using the right-hand scroll bar). Then double-click on this row to see the results for this variable (industry output of commodity i in region r). Select 2 decimal places (see the third drop-down list box along the top row of the **Results** page - the only one with a single figure in it). Then you should see something like the following:

qo	SSA	EU	ROW
Food	-0.00	-0.00	-0.00
Mnfcs	-0.00	0.00	0.00
Svces	0.00	-0.00	0.00

The top 5 rows are shown in red since these are exogenous. The remaining are the endogenous percentage changes which are all zero (as expected).

In the second drop-down list box near the top of the screen you should see "**1 (Sim)**". This indicates that you are seeing the linearized simulation results (the percentage changes for this variable). This list box (which we refer to as the **which solution list box**) can be used to see other solutions or other aspects of the same solution.²⁹

For example, to see the pre-simulation levels values for this variable, click on this which solution list box and select "**2 Pre**". Then you should see something like the following.

qo	SSA	EU	ROW
Food	115218.680	1306824.000	3646912.000
Mnfcs	204783.938	4725396.000	12131394.000
Svces	277344.625	9961040.000	20273808.000

These are the pre-simulation levels values. The units of quantity are what one million pre-simulation dollars buy. For example, the pre-simulation level of output of food in SSA is 115,218 units. If you change the solution list box to "**3 Post**" you can see the post-simulation levels, while if you change it to "**4 Ch/%Ch**"³⁰ you can see the changes (not percentage changes) from pre-simulation levels to the

²⁹ If the only solution available in this drop-down list is "1 (Sim)", you will need to reset RunGTAP's options to show the others. To do this, click on **Tools / Options** in RunGTAP's main menu and click on **Show levels results (if present)**. Then close RunGTAP and repeat examples B2 and B3. This time the other solutions should be available.

³⁰ Usually the first (Sim) column shows a % change and the fourth (Ch/%Ch) column shows the corresponding ordinary change. However, for the few variables defined in the model as ordinary change variables, the first (Sim)

post-simulation levels. [In this case the changes are all small since the percentage changes are close to zero.]

Now **double-click anywhere on the numbers part of the page** (or click on the **Contents** button) to return to the Contents list.

Now double-click on the **pds** row (it is higher up) to see the results for this variable (basic price of commodity c in region r). Look at the percentage change results first (select “**1 (Sim)**” in the which solution list box), then the pre-simulation levels, then the post-simulation levels and finally the changes. Are these as you expect?

Please double-click on the numbers part of the page to return to the Contents list.

For variables with just one argument, you can see all four lots of results at once. To see this, first change the which solution list box to “**2 Pre**” and then double-click on the **pgdp** row. You will see all four lots of results for this variable (GDP price index).

Now return to the Contents list (can you remember how?).

Not all variables have levels results available. To see this, first change the which solution list box to “**1 (Sim)**” and then double-click on the **qinv** row to see the results for this variable (total investment by region). Then, when you change the which solution list box to “**2 Pre**” you just see **n.a.** which means that these results are not available.

Then click double-click on the numbers part to return to the Contents list and again select the linearized results by clicking on “**1 (Sim)**” from the which solution list box.

To see the effects of another useful way of listing the variables in the **Contents** page, click on the **V** between the two drop-down list boxes near the top of the screen (just under the gap between the **Title** and **RunGTAP** pages). You should see something like:

Variable	Size	No.	Name
Macros	1	12	Scalar variables (just one element)
Vectors size: 1	MARG_COMM	2	Vector variables ranging over set MARG_COMM (size 1)
Vectors size: 3	COMM	9	Vector variables ranging over set COMM (size 3)
Vectors size: 3	REG	48	Vector variables ranging over set REG (size 3)
afa	COMM*ACTS*REG	1	Composite intermed. input c augmenting tech change ...
afe	ENDW *ACTS*REG	1	Primary factor e augmenting tech change ...

(and so on).

To see the results for all variables with a single argument ranging over the set **REG**, double-click on the third of the rows above. This time you will see the linear (percentage-change) results for the 48 such variables (including **qinv**, **pop** and **pgov**). Again you can see the other results by making a selection in the drop-down list box controlling which lot of results you are viewing (the one showing “**1 (Sim)**” at present). When you have finished looking at these results, double-click on the numbers to return to the Contents list.

You can return to the other view of the Contents list by clicking again on the **V**.

column shows an ordinary change and the fourth (**Ch/%Ch**) column shows the corresponding % change. Hence the odd “**Ch/%Ch**” column heading.

Example B3A - Looking at the Results using ViewSOL or ViewHAR

You can use the menu item *View / Results using...* to see results using ViewSOL or ViewHAR.

ViewSOL presents a view much like the results page of RunGTAP, except that:

- It shows values for some variables hidden from the RunGTAP results page. Many of the variables are prefixed "CNT" and are used to work out the welfare decomposition.
- ViewSOL allows you to view and compare two or more solutions.

With ViewHAR you see the results as a conventional HAR file (aka SOL file). The special advantages are:

- You can more easily see values for variables with 3 or more dimensions (ViewSOL will not let you see values for variables with more than 3 dimensions).
- You could write a TABLO program to read the HAR/SOL file and perform results-processing or report-generating tasks.

Example B4 - Look at the updated data

As well as producing simulation results which are percentage changes in prices and volumes, the simulation in Example B2 above produced updated data files, which reflect the state of the economy as it would be after the shock (10 per cent increase in the numeraire). To see this updated data, select:

View / Updated Data / Updated Core Data

from the RunGTAP menu. As usual, RunGTAP opens ViewHAR to show you this data.

The 3x3 array of data at header "VMPB" contains the Value of Private household expenditure on each Imported commodity in each region, valued at Basic prices. Double-click on this **VMPB** row in the Contents list to look at its data. To make this occupy just the **left** half of the screen, click (**carefully**) on the left half of this icon  near the bottom right-hand corner of the ViewHAR window.

Now, for the original (pre-simulation) version of this data, go back to RunGTAP and select *View / Base Data / Core Data*. Again double-click to select the **VMPB** data and this time click carefully on the **right** side of the  icon.

Then (via the **Taskbar** or **Alt+Tab**), bring the other ViewHAR window (the one in the left hand half of the screen) to the top.

Now that you have both the original and the updated data on your screen, look at a particular cell, say that corresponding to food in SSA. Think what has happened to prices and to quantities in this simulation, and hence deduce what must have happened to values. Check that the pre- and post-simulation VMPB("food","SSA") values are consistent with this.³¹ You might like to repeat this for some other cells in this and/or other headers. Close the two ViewHAR windows when you are finished.

Close down RunGTAP and all associated windows before proceeding.

³¹ The pre- and post-simulation values are approximately 3869.07 and 4255.98. Note that 10% of the former is 386.907. When you add this to the pre-simulation value you get 4255.98. Thus, as expected, the post-simulation value is 10% more than the pre-simulation value.

B.3 A Simulation Reducing one Distortion

If necessary, start RunGTAP and check that the **NCORS3X3** version is selected.

The Simulation

The simulation in these next examples is one in which the shock is a

10% reduction in the power of the tariff

applied by the European Union to imports of food (and agricultural) products coming from Sub-Saharan Africa.

Example B5 - What is the pre-simulation power of this tariff?

To see the **pre-simulation** value of this tariff, select **View / Base Data / GTAPView Output** from RunGTAP's main menu. This will open a ViewHAR window which shows you the GTAPVIEW output calculated using the pre-simulation data.

The import tariff information is at header **BI02**. Double-click on this row in the Contents page. Then, since you are interested in imports from SSA to EU, change the 2nd, 3rd and 4th boxes showing which parts you are looking at from

All REG	Sum REG	Sum IMPVALUE
to		
SSA	EU	All IMPVALUE

Now the **impcost** column shows the value of imports at world prices (VCIF)³² and the **mtax** column shows the value of the import tariff. For **(food,SSA,EU)** you can see that **VCIF=10133.91** and **mtax=1321.69**. What is the *ad valorem* **rate** of the tariff on imports of food from SSA to EU? What is the **power** of the import tariff on food from SSA imported into EU?³³

The 10% reduction applies to the power of the tariff (not to the *ad valorem* rate). What will the post-simulation power of this tariff be? What will the post-simulation *ad valorem* rate be?³⁴

Close this GTAPVIEW output window before continuing.

Example B6 - Implementing the shock and running the simulation

Click on **Closure**. You can choose between several different closures when running a simulation. The panel below shows which variables are exogenous in the selected closure. RunGTAP allows you to load different closures already prepared.³⁵ To see this, use **Tools...Load closure** command to select **book** and then click on **OK**. Notice the different exogenous list. In this simulation we want you to use the standard closure for GTAP, namely the closure referred to as "Standard". To use this one, click again on **Tools...Load closure**, select **Standard** and then click on **OK**.

Go to **Shocks** to introduce the 10% reduction in EU's tariffs of food products coming from SSA:

³² You can look in GTPVEWv7.TAB to check this.

³³ The *ad valorem* rate is $100 * (1321.69 / 10133.91) = 13.042$ per cent. This means that the power of the tariff (VMSB/VCIF) is 1.13042. [The power is 1 plus the *ad valorem* rate divided by 100.]

³⁴ A 10 per cent reduction in the power of 1.13042 will reduce this power by 0.113042. This will make the post-simulation power equal to $1.13042 - 0.113042 = 1.01738$. Hence the post-simulation *ad valorem* rate will be just 1.738 per cent. [The effect of this shock is thus to nearly eliminate the import tariff.]

³⁵ You can also edit the closure shown on the Closure page, and can save this edited closure.

- First delete the default shocks already showing in the shock list by clicking on the **Clear Shocks List** button.
- Now use the combo box at the top left to choose which **variable to shock**: click on the arrow at the right. This will show you a list of variables. Click on *tms* as the variable to shock and check this is the import tax in region d on good c imported from region s.
- As you have chosen a matrix variable (*tms*) a second row of combo boxes has appeared, allowing you to choose whether to shock all of the elements, or only a subset. Click on the arrow of the first box to the left which currently says **All COMM**. As you want to reduce tariffs only in the food sector, click on **Food**. Now click on the arrow of the second box which currently says **All REG** which will show you a list of the regions in this 3x3 aggregation. Since these are the regions from which the commodity in question originates, you need to select **SSA** here. Finally check the last box to the right to select the appropriate destination of the goods in question, namely **EU**.
- Now, go to the third row of boxes. You can write the value of the shock in your experiment in the first box to the left: **-10**, meaning a reduction of 10% in the selected tariffs.
- The other box *Type of Shock* offers three possible options, namely **% change rate** and **%change power** and **%target rate**. Select **% change power**.³⁶ Notice that the wonderfully helpful RunGTAP indicates that the initial *ad valorem* rate is 13.04% (as you saw in Example B5 above) and that, after this shock is applied, the final (or post-simulation) *ad valorem* rate will be 1.74%.
- Add the shock above to list of shocks by clicking on **Add to shock list**. This is the only shock you need in this experiment. [If others were needed you could select them and add them to the shock list.]

Go to the **Solve** page.

- First select the solution method to use. To do this, click on the **Change** button to the right of the **Solution method** label. [This button is the top-most button on the page.] In the **Method** part select **Johansen**. [We will discuss the alternative methods later in Example B14.] Click on **OK**.
- Next notice the **Experiment description** label and the edit box to its right. It is good practice to enter a few words summarising the simulation you are about to carry out. So first select the existing entry with your mouse and then replace it by **10 percent reduction in power of import tariff on food from SSA into EU**
- Now for the most important step. You should save all the details (shocks, closure, solution method, etc) in a file so that you can repeat the simulation later if necessary. Click the **Save Experiment** button and type in "TMSFSE" as the new file name. Edit the experiment description if necessary. Then press OK. Notice that the title bar at the top of RunGTAP reflects the new experiment name. Later on, you could use the **Load Experiment** button to recall all the details of the TMSFSE simulation.
- Now click on **Solve** to start the solution process.

³⁶ Suppose that, for particular values of (c,s,d), VMSB=120 and VCIF=100; then TMS_L=VMSB/VCIF=1.2 which means that the *ad valorem* tariff rate is 20%. If you apply a **% change power** (in RunGTAP terminology) shock of -10 to this component of variable *tms*, this will reduce the value of TMS_L by 10% namely, from 1.20 to 1.20-0.12=1.08. If however you were to apply a **% change rate** (in RunGTAP terminology) shock of -10 to this component of *tms*, this would reduce the *ad valorem* rate of 20% by one-tenth (=10%) which would move the *ad valorem* rate to 18% and hence move the value of TMS_L to 1.18. The option **%target rate** is available if you wish to carry out a simulation in which you change the *ad valorem* rate to some target value [For more details, see RunGTAP's Help file.]

RunGTAP runs the TABLO-generated program GTAPv7.EXE to solve GTAP.

Example B7 - Looking at the results

When the solution has been obtained, RunGTAP will present you with a box telling you how long it took to obtain the solution. Sometimes it will suggest that you save an experiment file containing a full specification of your simulation. That should not happen this time, since you saved the simulation details as an EXP file just prior to running the simulation. Go directly to the **Results** page to see the output of the experiment.

This page allows you to inspect the variable values computed during the **last** successful solution process. Initially a **Contents** window is displayed, listing the model variables. You can double-click on the row containing the variable you are interested in to see its results in this simulation. [You cannot double-click on the variable in the left-hand column of the Contents list but must double-click in one of the right-hand columns. Alternatively you can go to the results by clicking in this right-hand part and then pressing the space bar; and once you have clicked here you can move up and down using the arrow keys.]

As you saw in Example B3 above, you can also use the button marked “**V**” to arrange the contents list in two ways:

- variables listed individually
- variables ranging over the same set grouped together.

Experiment with this until you see something like

Variable	Size	No.	Name
Macros	1	12	Scalar variables (just one element)
afa	COMM*ACTS*REG	1	composite intermed. input c augmenting tech change
afe	ENDW_COMM*ACTS*REG	1	primary factor e augmenting tech change by act a in r
ams	COMM*REG*REG	1	commodity c augmenting tech change from s to d
ao	ACTS*REG	1	output augmenting technical change by activity a in r
au	REG	1	input-neutral shift in utility function
ava	ACTS*REG	1	value added augmenting tech change by act. a in r
cgdslack	REG	1	slack variable for qinv(r)
compvalad	ACTS*REG	1	composition of value added for good c and region r
del_tbal	REG	1	change in trade balance X - M, \$ US million
del_tbalc	COMM*REG	1	change in trade balance by c and by r, \$ US million
(etc)			

Look first at the results for the trade balances (variable **del_tbal**). What is the result for Sub-Saharan Africa? What does the number shown here mean? To return to the **Contents** list you can **double-click on the numbers part of the results page** for the variable you are currently looking at (or you can click on the **Contents** button). Do this.

Now look to see what happened to the outputs of the different commodities in SSA (this is variable **qo**). What happened to the output of food in SSA (are you surprised by the sign of this result) and what happened to the output of manufactures in the SSA? Note that results for exogenous variables are shown in red (which explains why some of the **qo** results are red). Note also that you can change the number of decimal places which are shown in the results by clicking on one of the combo boxes on the results pages. The result for **qo(“food”,“SSA”)** is 3.116 - we will refer to this result later when we discuss alternative solution methods.

You can also look at the pre- and post-simulation levels numbers for *qo* by selecting these in the which solution list box (as you did in Example B3 above).

The Contents page lists all the variables which may be viewed. You can filter the variables to show only those which contain the filter element. For example, click on the arrow of the box at the top left of the screen (which is probably currently showing **Everything**). Select **Capital**. Now you have changed the Contents window, which is showing only the variables containing a component Capital. The effect of filtering in this way is that: macro variables are not shown; each vector variable which contains Capital as one of its components turns into a scalar; each matrix turns into 1 or 2 vectors (2 if Capital is both a row and a column label); and each 3-dimensional array turns into 1, 2 or 3 matrices³⁷. Choose **Everything** again to switch filtering off.

Example B8 - Look at the updated data

After the simulation has been completed there is a new option available in RunGTAP's main menu. Click on **View / Updated Data**. The menu will show you the following options:

- Updated Core Data
- Updated GTAPView Output
- Updated Tax Rates
- Welfare Decomposition
- Volume Changes

The Updated Core data are the data in the main data file (the one with logical name **GTAPDATA** in **GTAPv7.TAB**) used by GTAP, but in this case the updated (that is, post-simulation) version. This updated data reflects the state of the economy as it would be after the shock (10% reduction in the import tariff on food coming from SSA to the EU).

GTAPView uses the updated core data to compute various subtotals and ratios useful for understanding and explaining results.

Click on **View / Updated Data / Updated Core Data**. A ViewHAR window will open. What is the post-simulation value, at FOB prices, of exports of food from SSA to the EU? Do you think this is more or less than the pre-simulation value of these exports? To check this, select **View / Base data / Core data** from RunGTAP's main menu (yes, RunGTAP makes it easy to look at the pre-simulation data) and check the value of exports of food from SSA to EU there. [You will now have two ViewHAR windows open, one with the pre-simulation core data and the other with the post-simulation core data. You may find it useful to use the  icon to make each of these take just half of the screen.]

Close the two ViewHAR windows when you have finished comparing the pre- and post-simulation data.

You can also look at the GTAPVIEW information about the post-simulation data. To do this, click on **View / Updated Data / Updated GTAPView Output**. For example, you can check the constituents of GDP from the expenditure side. You can compare these values and shares to those in the pre-simulation data (which you can obtain via **View / Base Data / GTAPView Output**).

³⁷ The Results page will not allow you to view 3-dimensional arrays -- but you can use filtering to see any 2-dimensional slice of such arrays.

Example B9 - Saving the solution and shocks

Some of the files produced when you run a simulation are saved in a folder called SAVESIMS underneath the current version folder. These include SL4/SLC solution files, a UPD file of updated core data, and a CMF file which enables the simulation to be repeated outside RunGTAP.

The *File / Save Outputs...* menu item allows you to save more outputs, either individually, or combined in a ZIP archive.

Example B10 - Changing the solution method

In Example B6 above, you told RunGTAP to solve the model using Johansen's method. With Johansen's method, only approximate solutions are obtained to the nonlinear levels equations of the model. RunGTAP (like GEMPACK) also provides multi-step solution procedures referred to as **Euler's** and **Gragg's solution methods**. Coupled with **extrapolation** (a standard numerical method used in various branches of science and engineering), these are able to provide accurate solutions of the levels equations of the model.

To change the solution method in RunGTAP go to *Solve* and click on the *Change* button to the right of the **Solution method** label. [This button is the top-most button on the page.] In the *Method* part select **Gragg** and then make sure that you have **3 solutions** and **2,4,6 steps**. [Select these if they do not become selected after you click on "Gragg".] Make sure that the "Automatic Accuracy" box is not checked and that the "Subintervals" box shows 1. Then click *Ok*. Then *Solve* the model.

Since you changed the experiment, you will be prompted to save the experiment details. Click **Yes** and save the experiment with the same name as before: TMSFSE. Ignore RunGTAP's warning that you are overwriting the record of your previous experiment. The Solve process will begin.

Once it has solved, RunGTAP displays an accuracy summary chart showing how many variables are accurate in the levels to 1, 2, 3, 4, 5, or 6 figures. A little face summarizes overall accuracy -- hopefully it is smiling. Accuracy of the updated data is summarized in the same way. Click **Help** to learn more, then **OK** to proceed.

Another message now tells how long the solution took. Click **OK** then select *View / Updated data / XAC file*. In this **Extrapolation Accuracy File** you can see the effect of the different solution method and the extrapolation. To be specific, look at the results for component number 6 of variable **qo**; these are the results for **qo("food";"SSA")**. This row of the XAC file is

6 3.83714 3.85428 3.85755 3.86020 CX 4 L5

Here the results are respectively the 2-step (3.83714), the 4-step (3.85428), the 6-step (3.85755) and the extrapolated result (3.86020). The notation **CX 4** means that you can be confident in the extrapolated result (this is the **CX**) and you can be confident that at least 4 of the figures (that is, at least the 3.860 part) in the extrapolated result are accurate. The **L5** means that at least 5 figures in the corresponding levels result (as distinct from the percentage change result) are accurate. Recall from an earlier example that the Johansen result (really a 1-step result) for this same simulation is 3.116. You can see that the Johansen result is only just an approximation to the accurate result of 3.860. In producing the Johansen result, GEMPACK solves the system of linear equations associated with the model just once (which is why Johansen is called a 1-step method) while 6 different systems of linear equations are solved to obtain the 6-step solution.

When you have finished looking at the results in this Extrapolation Accuracy file (abbreviated to XAC file by RunGTAP), close TABmate to return to RunGTAP.

The accuracy summary chart that RunGTAP displays after solving is a brief digest of the XAC file. You can select *View / Accuracy Summary* to see it again.

Example B11 - Has the power of the tariff actually been reduced by 10% ?

To check if the power of import tariff on food from SSA imported into EU has actually been reduced by 10%, you need to know the size of this power in the pre-simulation data and in the updated data. RunGTAP makes it easy for you to see these values.

You saw the pre-simulation values in Example B5 above.

To see the **post**-simulation value of this power, select *View / Updated data / Updated GTAPView Output* from RunGTAP's main menu. This will open a ViewHAR window which shows you the GTAPVIEW output calculated using the updated (that is, post-simulation) data. As in Example B5, double-click to look at the data at header **BI02**. Proceed as in Example B5 to calculate the (post-simulation) **ad valorem rate** for imports of food from SSA into the EU? What is the post-simulation **power** of the import tariff on food from SSA imported into EU? Is this power 10% lower than in the pre-simulation data?³⁸

Close the ViewHAR window.

Example B12 - Copying results into a Spreadsheet

RunGTAP makes it easy for you to copy simulation results (or data) into spreadsheets or other documents. You will find this invaluable when you are preparing reports.

As an example, below we set out the simple steps needed to copy the **qo** results from the previous simulation into Excel (or another spreadsheet).

Click on **Results** and then go to the **qo** results (the percentage change ones). You should see something like

qo	SSA	EU	ROW
Food	3.860	-0.367	-0.019
Mnfcs	-2.198	0.082	0.003
Svces	-0.366	0.004	0.001

Now click on **Copy** from RunGTAP's main menu. A message **Data copied to Clipboard** will be shown for a few seconds.

Now start Excel (or your favourite spreadsheet program) running. In a new sheet, click on the top left-hand cell and then select **Edit / Paste** from the main menu. You should see the above table appear in Excel. Note that the number of decimal places selected in RunGTAP affects the exact numbers copied.

Close Excel.

³⁸ The post-simulation rate is $100 \times (264.725 / 15231.557) = 1.738$ so the post-simulation power is 1.01738. The pre-simulation rate is 13.042 so the pre-simulation power is 1.13042. Note that 10% of 1.13042 is 0.113042; if you subtract this from 1.13042 you get 1.01738. [Note that here you have been working with a fairly accurate solution (Gragg 2,4,6) to the model. You might like to try the corresponding check based on the less accurate Johansen solution; if so, you will find that the power of the tariff only drops by approximately 7%; this emphasises the need to produce an accurate solution to see the true consequences of a shock.]

Example B13 - Looking at Volume Changes

When you carry out a simulation using RunGTAP, the results relating to volumes are mainly presented as percentage changes in volumes. (For example, the **qo** results give information about percentage changes in the outputs of different commodities in the different regions.) But you can get information about volume changes clicking **View / Updated Data / Volume Changes**. [RunGTAP uses volume changes computed in GTAPv7.TAB (click *Ctrl-F*, then type GTAPVOL). Prior to GTAPv7, the volume changes were calculated inside a program named GTAPVOL.TAB³⁹.]

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

$$\mathbf{DQO(a,r)} = [\mathbf{qo(a,r)/100}] * \mathbf{VOS(a,r)} \quad (1)$$

to calculate **DQO(a,r)**. Here **VOS(a,r)** is the pre-simulation value of outputs at suppliers' prices, *in the initial equilibrium*. Although the global data file 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 **VOS(a,r)**; that is, **one unit of output by activity a in region r** is defined to be the amount that one dollar will buy (at pre-simulation basic prices). Volume changes are then reported by GTAPVOL in these same units. Thus **DQO(a,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, **VOS("food","eu")** is 1306824 million dollars. [To see this, select **View / Base Data / GTAPView Output** and look at the **VOS** values which are at header **CM04** (in the combo box **OUTVALUE**, choose, "PRODREV").] GTAPVOL says that the pre-simulation output of food in the EU at supplier's prices is 1306824 million units. Alternatively, choose "Sum OUTVALUE" in the third combo box to see that the pre-simulation output of food in the EU at basic prices is 1293092.375.

You can look at the volume changes produced by the simulation above (Example B10) by selecting **View / Updated Data / Volume changes** in RunGTAP's main menu. To see the change in the volume of output of food in the EU, select the **DQO** row in the Contents list. You will see that the output of food decreases by about 4790 units.

Is this consistent with the simulation results and the initial volume? To check this, find the simulation result for **qo("food","EU")** and the pre-simulation volume of food produced in the EU.⁴⁰

Note that you can also see the change in the volume of **qo("food","EU")** by looking at the **Results** for variable **qo** and selecting the "**4 Chng**" in the which solution list box. RunGTAP's **View / Updated Data / Volume changes** is just an alternative way of seeing these (and other) changes.

You might like to look at other volume changes reported by RunGTAP.

³⁹ Following GTAP tradition, we refer to this as **GTAPVOL**, even though the RunGTAP file containing its source code is now incorporated in GTAPv7.TAB. Pronounce "GTPVOL" the same as "GTAPVOL".

⁴⁰ The simulation result for **qo("food","EU")** is -0.36654. The initial volume of food produced in the EU is 1306824 units. Note that 0.36654% of 1306824 is 4790.1 which checks well with the volume reduction of 4790.125 reported by RunGTAP.

B.4 A Simulation Removing Several Distortions with the 3x3 data

In the example below you will carry out a simulation removing several distortions on Food in the European Union. More precisely, you will remove

- output subsidies on food and farm products in the European Union **to("food","food","EU")**,
- export subsidies on food and farm products exported from the European Union to the other two regions **txs("food","EU","SSA")** and **txs("food","EU","ROW")**
- import taxes on food and farm products imported into the European Union from the other two regions **tms("food","SSA","EU")** and **tms("food","ROW","EU")**

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

Example B14 - Carry out this simulation

The usual procedure is repeated here. First check that RunGTAP is using the 3x3 aggregation (NCORS3X3). Click on *Closure* and check that the *Standard* closure is being used.

Go to *Shocks*.

(a) Select any shocks shown in the shocks window and delete them so that you are starting afresh.

(b) First add the shock to eliminate output subsidies on food in the EU.

- Select **"to"** as the first variable to shock
- Go to **"Elements to shock"**. Select **"Food"** in the first and second box and **"EU"** in the third box
- Now, go to the third row of boxes. You can write the value of the shock in our experiment in the first box to the left: **-100**, meaning the elimination of the production subsidies for food in the EU.
- Select **"%change rate"** as the **Type of shock**. This is because here the intention is to eliminate 100% of the subsidy.⁴¹
- Add the shock to the panel below clicking on *Add to shock list*

(c) Next add the shocks to eliminate export subsidies on food exported from the EU.

- Select **"txs"** as the second variable to shock.
- First select component **txs("food","EU","SSA")**, then select **-100** as the shock and **%change rate** as the type of shock. Add this shock to the list of shocks via *Add to shock list*.
- Repeat for **txs("food","EU","ROW")**.

(d) Next add the shocks to eliminate tariffs on food imported from other regions into the EU (variable **tms**).

Next go to the *Solve* page and choose a solution method. Select the solution method as **Gragg: 2-4-6 steps extrapolation**. Enter a suitable **Experiment description** and save the experiment details [click *Save Experiment* and type in the new file name "CUTSUBS"]. Then click on *Solve* to start the solution process.

⁴¹ If you selected **%change power** the effect would be to reduce the power of the tariff by -100%. This would reduce the power to zero in the levels; such a power would mean a 100% subsidy which is not what you want.

Another way to eliminate the subsidy is to select **%target rate** with a shock value of 0.

When the solution has been calculated, you will see the **Accuracy Summary** dialog. Use its Help button to learn what it can tell you. Then go on to examine some of the results.

Example B15 - Using Subtotals to divide up the effects of the shocks. [optional]

If your simulation involves several different shocks, GEMPACK offers you a very powerful facility known as "subtotals". This allows you to apportion the total change in any endogenous variable between the various shocks.

Try it out using the simulation from Example B14 above [if necessary, reload the CUTSUBS experiment file that you saved]. Go to the *Shocks* page. You will define 3 subtotals which decompose the results into the effects of the **to** changes, the **txs** changes and the **tms** changes.

To do this, first click on the **Define Subtotal** button.

Select variable **to**.

Click on the **Add variable to subtotal** button.

Click on the **OK** button.

Click **OK** to accept the Subtotal Description "to changes".

You will be returned to the *Shocks* page. Note that the line

Subtotal to = to changes ;

has been added to the shocks list.

Repeat for variable **txs** to get the line

Subtotal txs = txs changes ;

added, and then repeat for variable **tms** to add the line

Subtotal tms = tms changes ;

Then go to the *Solve* page and save the experiment details [click *Save Experiment* and re-use the file name "CUTSUBS"]. Then click on *Solve*.

When the solution has been calculated, examine the results for EV. You should see 4 columns of numbers. The first column shows the same EV changes that were calculated in Example B14. The next 3 columns show how much of the total change in EV may be attributed to the exogenous changes in, respectively, to, txs and tms. Click on the *Description* button above the results to verify this.

The subtotals columns show that the welfare gain in SSA comes mainly from the tms tariff cuts (last column), while the EU and ROW welfare changes come mainly from reducing the export subsidies txs (subtotal 2).

Go back to the *Shocks* page and click the *Define Subtotal* button again. As you saw above, this launches a window which can help you compose subtotal statements like the three that you added before. Use the Help button to find out how to use this window, and to find out more about the theory behind subtotals.

C. MAKING A NEW VERSION FOR RunGTAP

To follow the examples in Section C, you need the GTAPAgg program mentioned below. This is not automatically packaged with RunGTAP -- it must be purchased separately from GTAP at Purdue.

As you have seen, there are many different versions available under RunGTAP. Each such version is basically a different aggregation of the full GTAP data base. How can you add another version? You will see how in this section (and it is really quite easy).

In version 4 of the GTAP data base (this is the current⁴² version), the full data recognises 45 regions, 50 sectors (or commodities) and 5 primary factors. Aggregating this means grouping together several regions into composite regions, and doing the same for commodities and factors. For example, in the NCORS3X3 version,

- regions saf (South African Customs Union), rsa (Rest of Southern Africa) and rss (Rest of sub-Saharan Africa) have been aggregated into the region denoted SSA in NCORS3X3, various regions have been aggregated into the region denoted EU in NCORS3X3 and the remaining regions have been aggregated into ROW,
- the 50 commodities have been aggregated into 3 aggregate commodities,
- the 5 factors remain the same in NCORS3X3 as in the original data.

The easiest way to produce a new aggregation of the GTAP data is to use the **GTAPAgg** software. Once you have created a new aggregation using GTAPAgg, it is easy to incorporate this into a new version in RunGTAP. In this section you will carry out these two steps to create a new version in RunGTAP.

Example C1 – Making a new aggregation using GTAPAgg

Start GTAPAgg running by double-clicking on its icon.

GTAPAgg provides what it calls a default aggregation from which you can make other aggregations.⁴³ To start from this default aggregation, click on the button

Read aggregation scheme from file

In the open dialog box which appears, click on **Default.agg** and then click on **Open** .

Now let's work on the regional aggregation first. [You will do the sectoral and factor aggregation later.] To do this, click on the button

View/change regional aggregation

In the table at the bottom you can see that the default aggregation recognises 3 composite regions, the first called **NAFTA** comprising regions can, usa, mex, the second called **EU** comprising gbr, deu, dnk, swe, fin, reu, the third called **ROW** comprising the remaining 36 regions.

⁴² These instructions assume you have GTAPAgg for version 4 GTAP data. You might instead have GTAPAgg for version 5 GTAP data. There are more regions and sectors in the Version 5 data.

⁴³ If you have several aggregations available, you can start from any one and modify it using GTAPAgg to produce a new aggregation.

Suppose that you are about to undertake an application relating to Australia and New Zealand, and that you want to separate these two out from ROW into a new aggregated region called **ANZ**. You can do this as follows.

Right-click [that is, click the right (not the left) mouse button] on the box containing the region **ROW** in the table at the bottom of the screen. You will see a popup menu. On this menu, select the option **Insert before**. A new aggregated region called **New1** will appear above ROW and below EU.

Click on **New1** and change it to **ANZ**.

Now go to the list of regions and click on the **4 ROW** in the column titled “New region” next to **aus** (region number 1). Click on the arrow which appears in this box to reveal a drop-down menu of the possible aggregated regions. Click on **ANZ** to indicate that aus should belong to ANZ.

Repeat for **nzl** (New Zealand).

This completes the change to the region aggregation. Click on the **OK** button. This will return you to the main GTAPAgg page.

Now for the sectoral aggregation, click on the button

View/change sectoral aggregation

You will see that there are 4 aggregated commodities called Food, OthPrimary, Mnfc and Svces. You can see which of the original 50 sectors belong to each of these. You could make changes here in just the same way you made changes to the regional aggregation. For simplicity in this example, do not make any changes here. So just click on **OK** to return to the main GTAPAgg page.

Similarly you can see the factor aggregation by clicking on

View/change factor aggregation

You will see that the 5 factors remain separate and that the factors UnSkLab, SkLab and Capital have been designated as mobile factors. Again, for simplicity in this example, do not make any changes here. Click on **OK** to return to the main GTAPAgg page.

Now click on

Save aggregation scheme to file

In the save dialog box which appears, type **new4x4.agg** in the “File name:” box and then click on **Save**.

At this stage you have specified the region, sector and factor aggregation mapping but GTAPAgg has not yet carried out the arithmetic to aggregate the data as requested. To do this, click on

Create aggregated database

A Save dialog box appears: you can choose which folder your aggregation will be stored in (remember this for future reference). Accept the suggested name **new4x4.zip** by clicking on the **Save** button. If you are asked “do you want to compute aggregated time series data?”, click on **No** (again for simplicity).

It will take GTAPAgg a minute or two to do the arithmetic. While this is happening you see a message (near the bottom of the screen) “Wait..Aggregating”.

When the aggregation is complete, GTAPAgg gives you a message indicating that the aggregation is stored in the ZIP archive **new4x4.zip** as you requested. Make a note of the location of this file.⁴⁴ A button appears so that you can View Output Files straightaway - ignore this for now.

This completes the creation of the new aggregation. [Wasn't that easy?]

Close GTAPAgg by clicking on the **X** in the top right-hand corner of the screen.

Example C2 – Adding this aggregation as a new version in RunGTAP

Here you will add the aggregation created in the previous Example as a new version under RunGTAP. Start RunGTAP by double-clicking on its icon. Select

Version / New

from the main menu. A Wizard (with a very friendly face) will appear to guide you through the next steps.

Click on ***Next*** .

In the next screen, click on **New aggregation** (in the top box). Then click in the box to the right asking for the name of this new version, and change the name to **new4x4** . Then click on ***Next*** .

In the screen which appears, click on the bottom of the **Locate** buttons, the one with **ZIP archive ??** to its right. In the open dialog box which appears, you will need to locate the file **new4x4.zip** you created in Example C1 above. To do this, go to directory C:\GTAPAGG (or to the directory in which GTAPAgg is installed) in the usual Windows way, click on file **new4x4.zip** and click on **Open**. The label next to the bottom **Locate** button will change to **ZIP archive: c:\gtapagg\new4x4.zip** .

Then click on ***Next*** . Now click on ***FINISH*** in the screen shown to complete adding this new version.

RunGTAP informs you that it is about to switch to this version. Click **OK** . RunGTAP now needs to carry out various calculations to prepare the relevant files (such as GTAPView and so on) you expect to have available for any version⁴⁵. When these calculations are complete, you see a message indicating that the new solution has been generated. Click **OK**.

This completes the installation of a new version under RunGTAP. You can now carry out simulations with this version in the usual RunGTAP way.

Before closing RunGTAP, click on the ***Version*** page of RunGTAP. You will see that, to really complete the installation of this new version, you should complete the VERSION.TXT file which RunGTAP shows on its **Version** page to inform you and others about the version. We ask you to do this at a later date.

Now close RunGTAP.

⁴⁴ The file new4x4.zip will be saved by default in the directory in which GTAPAgg is installed.

⁴⁵ If you used an older copy of GTAPAgg, RunGTAP might tell you that it is converting the data files to a more modern format.

D. CUSTOMIZING RunGTAP

You need a GEMPACK licence to follow the examples in this section, which also assumes that you have had a little practice in using GEMPACK. From within RunGTAP, you can use **Help..GEMPACK licence** to verify that you have a properly installed GEMPACK licence.

Back in the 20th century, RunGTAP insisted that the same set of TABLO-generated programs (specified by TAB files) was used for all versions. From 2000 that restriction was lifted. Now you can have different variants of the GTAP model which are used with particular versions.

To see an example of this, use **Version...Change** to change to the NTP16 version and look over the information on the *Version* page. Next select the menu item **Version...Modules** : a window appears indicating that for this version the GTAP module (ie, the main model) has been replaced by the module GTAPEv7 which is stored in the version folder. Click on the Help button and explore the 2 or three help topics which explain how the "modules" -- a jargon term -- are used. Then close Help and click on the word "GTAPE" which is visible at top right of the **Version...Modules** window. A second window appears listing all the components of the GTAPE module -- a number of files with the same name but different extensions. Click **Cancel** twice to return to RunGTAP.

The NTP16 version uses its own variant of the standard GTAPv7.TAB file, GTAPEv7.TAB, which incorporates some extra equations and variables. In the next paragraphs we explain how you can create a new version which also uses its own variant of the GTAP model. We divide the process into two stages. The first stage is preparatory; it consists of creating a new version with its own copies of the files related to GTAPV7.TAB. The only difference is that the new version will implement GTAP using GEMSIM instead of using the Fortran-based GTAPv7.EXE which RunGTAP uses by default.

Stage 1: Preparatory

Creating a new version based on the NCOR3x3 version

Use **Version...Change** to switch to the ncors3x3 version. Then use **Version...New** to create a new version, called NEWVER and "based on the same aggregation" as ncors3x3, by "simply copying everything" from ncors3x3. RunGTAP will switch to the NEWVER version and run a test simulation.

Enter Developer Mode and use TABmate as text editor

Use **Tools...Options** to launch the options dialogue. Check that RunGTAP can locate the programs TABLO, GEMSIM and TABmate (most likely all in the C:\GP directory). Switch on both the options **Developer Mode** and **Use TABmate as text editor**. Close the **Options** dialog.

A new top-level menu item, **Developer**, will have appeared. Use **Developer...DOS box in version dir** to launch a command prompt in the NEWVER subdirectory. Then type into the command line:

```
copy ..\GTAPv7.*
```

This will copy a number of files from the main RunGTAP folder down into the NEWVER subdirectory. Three of these are not needed, so delete them by typing in these three commands:

```
del GTAPv7.EXE
del GTAPv7.AXS
del GTAPv7.AXT
```

Return to RunGTAP (no need to close the DOS box).

Regenerate GTAP as a GEMSIM program

Use **View...Other text file** to open the file GTAPv7.STI in the NEWVER folder. The TABmate editor should appear with that file open. Go down to the bottom of the file, past the condensation instructions, and find a line starting "wfp". Change "wfp" to "pgs". This instructs TABLO to prepare a GEMSIM program ("wfp" causes a Fortran program to be written). Use TABmate's **Save** button to save the STI file.

In TABmate, use the **Options..Code and Other** menu item to make sure that "TABLO CODE produces GEMSIM files", and to tell TABmate, if necessary, where the TABLO program is located.

Now use TABmate's **Open** button to open the file GTAPv7.TAB. Finally use the **TABLO STI** button to **Run** TABLO from the STI file. A message "busy TABLOing" appears at the bottom of TABmate. After a while a box should appear, informing you that GEMSIM files MIN, GSS and GST files have been produced. Close TABmate and return to RunGTAP.

Use the GEMSIM program for this version

Use **Version...Modules** to launch the Modules dialog. Click on the white box representing the version-specific TAB file for the main model (at top of right-most column). A second dialog appears -- choose the second option "stored in version folder". Then click on the file GTAPv7.TAB which appears in the list underneath. A list of the files making up the GTAP module should appear, together with the encouraging message "all necessary parts are present and up-to-date". Click OK twice; a warning appears saying that the main module has changed. Follow its advice and choose the **Tools...Run Test Simulation** menu item. The Wait box with spinning globe appears -- study this carefully. In the middle of the Wait box is a line which counts elapsed seconds, and just below this is a line listing the name of the program currently running. You should be able to see that the GTAP module is (a) located in the NEWVER folder, and (b) implemented using GEMSIM. If it went by too quick, click the Solve button and study the Wait box again.

Stage 2: Changing GTAPv7.TAB

Adding 1 new equation and 1 new variable to GTAPv7.TAB

Use **View...TAB files...Main model** to open the file GTAPv7.TAB in the NEWVER folder. At the bottom of that file, add the following lines:

```
Variable
  (all,a,ACTS)(all,r,REG) pmat(a,r) # price index for intermediate inputs #;
Equation E_PMAT # price index for intermediate inputs #
  (all,a,ACTS)(all,r,REG)
  [sum(c,COMM, VFP(c,a,r))] * pmat(a,r) =
  sum(c,COMM, VFP(c,a,r) * pfa(c,a,r));
```

The above lines declare a new variable, **pmat**, which is then specified to be the average price of each firm's material inputs (put the cursor on "VFP" and "pfa" and click **Gloss** to be reminded what these symbols are). When you have typed them in, click the **TABLO Check** button -- the message "no errors found" should appear at the bottom of the TABmate window.. If not, correct any typing errors you made⁴⁶ and **TABLO Check** again: repeat this until you see the message "no errors found".

⁴⁶ TABmate should locate any errors and underline them; click on the underlined parts to get an explanation of the problem.

Next use the **TABLO STI** button to **Run** TABLO from the STI file. A message "busy TABLOing" appears at the bottom of TABmate. After a while a box should appear, informing you that GEMSIM files MIN, GSS and GST files have been produced.

Adding the new variable to GTAPv7.MAP

A final complication is that RunGTAP does not normally show all variables on its Results page -- only those listed in a MAP file. You must add the new variable **pmat** to the MAP file. Each line of the file GTAPv7.MAP contains a 4-character header (contents do not matter but must be unique), followed by a space, followed by the variable name.

Use TABmate's **Open** button to open the file GTAPv7.MAP in the NEWVER folder (to see that file in the Open dialog, set **Files of type** to "all files"). At the bottom of that file add a new line:

```
0146 pmat
```

After clicking the **Save** button, close TABmate, and return to RunGTAP.

Running the new model

From RunGTAP, choose the **Tools...Run Test Simulation** menu item. After the model has solved, go to the Results page, and look for the variable **pmat**. Results from the test simulation are rather boring (it is a numeraire shock), so you might like to run a more interesting experiment and again examine the **pmat** results.

Complications which were avoided

In the example above, we added one new endogenous variable, **pmat**, together with an equation which explained it. Hence there was no need to alter any files specifying closure. If your modification involves adding exogenous variables you will have to add them to all experiment files for that version (files with EXP extension in version folder) as well as to the following files:

GTAPv7.CLG: the closure used for SAGEM GE Elasticities.

GTAP v7.CLP:(most of) the closure used for ALTERNATE runs.

If the changes you are making are substantial, you may prefer to run several simulations under WinGEM before you add your new module to RunGTAP. If so, you can find details about running simulations under WinGEM and GEMPACK in Appendix A.

E. FINDING OUT MORE ABOUT RunGTAP AND GEMPACK

During the Short Course, you will learn much more about RunGTAP. Whenever you carry out a simulation, RunGTAP prepares a vast amount of information about the simulation to assist you in analysing the simulation. Much of this information is available via the **View** menu. In particular, RunGTAP

- shows pre-simulation tax rate details, GE and PE Elasticities (see *View / Base Data*) .
- shows post-simulation tax rates, a GTAPVIEW summary of the post-simulation data, Welfare decomposition results and Volume changes (see *View / Updated Data*) .
- makes it easy for you to carry out systematic sensitivity with respect to changes in parameters or shocks (see *Tools / Sensitivity*) .
- allow you to estimate how much of the simulated changes were due to a subset of the shocks (the *Subtotals* facility).

You can find out detailed information about RunGTAP by consulting the extensive Help files which are supplied with RunGTAP. (See **Help** in RunGTAP's main menu.)

You may wonder about the syntax used in RunGTAP's Closure and Shocks pages. The syntax used here is identical to the relevant parts of so-called Command files (see the section below) required in GEMPACK for carrying out simulations. You can find more details in the GEMPACK user documentation (see the References section at the end of this document).

Connection between RunGTAP and GEMPACK

RunGTAP is a powerful Windows interface which minimizes the entry costs for working with GTAP. When you use RunGTAP to solve GTAP, RunGTAP solves the model by running the appropriate GEMPACK programs in the background (in DOS boxes). In particular, RunGTAP prepares all the files required by GEMPACK, and manages the interface between you, the user, and the GEMPACK programs in such a way that you are hardly aware of this interface. This is part of the power of RunGTAP.

For most purposes, you can use RunGTAP without being aware of GEMPACK in the background. As you have now seen, RunGTAP provides very effective ways of carrying out simulations, and of looking at pre- and post-simulation data. It also routinely calculates various other information associated with a simulation, including the welfare decomposition of results, the GTAPVIEW summary of the updated data and the volume changes.

Nonetheless, in this Short Course we introduced you to WinGEM and GEMPACK explicitly for three reasons.

- We don't want the basic modeling tasks to seem like a "black box" to you. For example, we want you to understand how the statements in TABLO files (for example, the Formulas) produce the desired calculations and output.
- As you analyse your simulation results, there may be information about the model which you need which is not packaged up by RunGTAP and the standard GTAP TABLO Input files but which you can obtain relatively easily via WinGEM. For example, you may need to calculate certain shares which are not available in GTAPVIEW: you can calculate these using the methods in sections A.3 and A.4 above.

- As you become more experienced, the modeling tasks you undertake may require you to modify the standard GTAP model (found in GTAPv7.TAB) in some way. [For example, you may need to add some behavior which is not present, or you may need to modify some existing equations.] In some cases, you will want to customise RunGTAP as in Part D above to include your new version of the model, and you may prefer to run several simulations under WinGEM before you incorporate your modifications into RunGTAP. In other cases you may not be able to use RunGTAP to solve this modified model, but you can use WinGEM.

In Appendix A, we set out the steps required to carry out using WinGEM the simulations in Part B above. Whenever you carry out a simulation using GEMSIM or a TABLO-generated program, you specify the details of the simulation (including the closure and shocks) on a GEMPACK **Command file**. Appendix A contains several examples of these. When you work with RunGTAP, it prepares the Command files for you. [For example, RunGTAP takes the lines from the Closure and Shocks pages and adds them into the Command file for the simulation.]

Other Models Besides GTAP

GEMPACK can be used to implement and solve a wide range of models besides GTAP.

Several example models apart from GTAP are available in subdirectory \GP\EXAMPLES on your computer. Details about these model files are given in the GEMPACK manual at <http://www.copsmodels.com/gpmanual.htm#models>

The models supplied include

Stylized Johansen - a pedagogical 2-sector model

Miniature ORANI - a pedagogical single-country model

ORANIG – a 23-sector version of the well-known ORANI model of the Australian economy; the theory of ORANIG has been used for models of numerous other countries including Denmark, South Africa, China, Thailand and Indonesia

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

For example, to work with the Stylized Johansen model, make a directory, change to it and copy the relevant files via the DOS commands

```
md \sj
cd \sj
copy \gp\examples\sj*.*
```

Suggestions for hands-on computing using WinGEM with some of these models are given in the GEMPACK manual at <http://www.copsmodels.com/gpmanual.htm#gpd8.2> . If you are interested in seeing how GEMPACK works with models other than GTAP, you might like to work through the examples there.

The complete GEMPACK manual is stored in the file gpmanual.htm in your GEMPACK folder (normally C:\GP).

Appendix A: Simulations using WinGEM

You need a GEMPACK licence to follow the examples in Appendix A, as well as the sample files issued to course participants.

In this Appendix, we show you how to use WinGEM to carry out simulations such as those you carried out in Part B above using RunGTAP.⁴⁷ [As before, if you modify the GTAP model, you may not be able to use RunGTAP for simulations, or you may prefer to use WinGEM until you have checked that your new model is working correctly.]

AA.0 Numeraire simulations

The GEMPACK procedure for carrying out a simulation with any model is shown in Figure 1 in chapter 6 of the GTAP Book, or alternatively in the 2 figures in the GEMPACK manual at <http://www.copsmodels.com/gpmanual.htm#gpd1.2.5.1>

In carrying out simulations with GTAP, you will first run GEMSIM (case (i) in section B.1 above) or GTAPv7.EXE (case (ii) in section B.1 above), and then run the Windows program ViewSOL.

In Examples AN1 and AN2 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 **pfactwld** - world price index of primary factors) is increased by 10 per cent.

Start WinGEM running and check (via *File / Change both default directories*) that the directory is C:\GTAP.

First it is necessary to implement a suitable version of the GTAP model. This is done by running TABLO as set out in detail in the next example.

Example AA0 - Condensing GTAP to produce output for GEMSIM

Select *Simulation / TABLO Implement...* from WinGEM's main menu. This will open a TABLO window. Next select *Options / Run from STI file* from the menu on this TABLO window. Then *Select* the Stored-input file **GTAP18GS.STI**.⁴⁸ [This is a so-called Stored-input file, often abbreviated to STI file. It contains all the responses to prompts from the program TABLO in order to condense the model (omissions, backsolves etc) and produce output GTAP.GSS and GTAP.GST for GEMSIM.] Then click on *Run*. TABLO will take a couple of minutes to run. When it has finished you might like to look at the LOG file produced.⁴⁹

⁴⁷ In the GEMPACK manual at <http://www.copsmodels.com/gpmanual.htm#gpd8.2> there is a detailed Hands-on introduction to the use of WinGEM for carrying out simulations with any model. This Appendix A provides such an introduction using the GTAP model.

⁴⁸ Another way to select "run from STI file" more quickly is to click on the "tape recorder" button along the top row in the TABLO window. This puts you straight into a file selection box for STI files. [If you "hover" your mouse - that is, leave it still for a few seconds - over this tape recorder button you will see the hint "Select STI file (Ctrl+T)". There are other so-called "shortcut" buttons in WinGEM; you might like to hover over some of them and use ones which appeal to you.]

⁴⁹ If you have a source-code version of GEMPACK, you can proceed similarly except using Stored-input file GTAPv7.STI instead of GTAP18GS.STI, to produce the TABLO-generated program GTAPv7.FOR. Then you can compile and link this (via WinGEM's *Simulation / Compile & Link...*) to produce GTAPv7.EXE. You can carry out simulations with GTAP by running GTAPv7.EXE instead of GEMSIM via WinGEM's *Simulation / Run TG program...*. RunGTAP uses GTAPv7.EXE to carry out simulations.

You are now ready to run simulations.

Example AN1 - Carrying out the basic numeraire simulation⁵⁰

Select *Simulation / GEMSIM Solve...* from WinGEM's main menu. This will open a GEMSIM window. Then select the *Command file* to be **NUM4P04.CMF**. [This Command file contains all the instructions for the simulation, including the closure and shocks. You will look at it in detail in Example AN3 below.] Then click on **Run**.

There will be quite a lot of screen activity in the resulting DOS box. The program is reading the 3x3 data files, and applying the shocks. When this has finished, the program window in control of the simulation will suggest that you **Go to ViewSOL**. Click on this and go to Example AN2 below.

Example AN2 - Run ViewSOL to look at the results

The Solution file NUM4P04.SL4 produced in the Example 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 be printed or looked at in a text editor such as GEMEDIT. The program ViewSOL lets you look at the results in a way which is very similar to the **Results** page of RunGTAP. [This example is very similar to Example B3 in Part B above.]

You will see the **Contents** page listing many of the variables of the model. ViewSOL has 3 slightly different formats for this Contents list. Select *Format...* from ViewSOL's main menu and there click on *Arrange vectors by name* (in the panel headed Vector options); then click **Ok** which will put you back to the Contents list.

To see the results of one of these variables listed by name, just **double-click** on the corresponding row in the Contents list. First double-click on the **qo** row to see the results for this variable (industry output of commodity i in region r). Select 3 decimal places (see the third drop-down list box along the top row of the current ViewSOL window - the only one with a single figure in it). Then you should see something like the following:

qo	SSA	EU	ROW
Food	-0.000	-0.000	-0.000
Mnfcs	-0.000	0.000	0.000
Svces	0.000	-0.000	0.000

The 3 rows are shown in black since these are endogenous. The results show endogenous percentage changes which are all zero (as expected).

In the second drop-down list box near the top of the screen you should see "**1 num4p04**". This indicates that you are seeing the linearized simulation results (the percentage changes for this variable). This list box (which we refer to as the **which solution list box**) can be used to see other solutions or other aspects of the same solution.

For example, to see the pre-simulation levels values for this variable, click on this which solution list box and select "**2 Pre num4p04**". Then you should see something like the following.

qo	SSA	EU	ROW
-----------	------------	-----------	------------

⁵⁰ The text of this example tells you how to run GEMSIM to carry out this simulation. If you wish to run the TABLO-generated program GTAP.EXE instead, select *Simulation / Run TG program...* from WinGEM's main menu. The rest is very similar to the text following. *Select* the Command file NUM4P04.CMF and then click on **Run**. Then **Go to ViewSOL** once the program has finished running.

Food	115218.680	1306824.000	3646912.000
Mnfcs	204783.938	4725396.000	12131394.000
Svces	277344.625	9961040.000	20273808.000

These are the pre-simulation levels values. The units of quantity are what one million pre-simulation dollars buy. For example, the pre-simulation level of output of food in SSA is 115,150 units. If you change the which solution list box to “**3 Post num4p04**” you can see the post-simulation levels, while if you change it to “**4 Chng num4p04**” you can see the changes (not percentage changes) from pre-simulation levels to the post-simulation levels. [In this case the changes are all small since the percentage changes are close to zero.]

Please click again on *Contents* to return to the Contents list.

Now double-click on the **pds** row to see the results for this variable (market price of commodity i in region r). Look at the percentage change results first (select “**1 num4p04**” in the which solution list box), then the pre-simulation levels, then the post-simulation levels and finally the changes. Are these as you expect?

Please click again on *Contents*.

For variables with just one argument, you can see all four lots of results at once. To see this, first change the which solution list box to “**2 Pre num4p04**” and then double-click on the **pgdp** row. You will see all four lots of results for this variable (GDP price index).

Again return to the Contents list.

Not all variables have levels results available. To see this, first change the which solution list box to “**1 num4p04**” and then double-click on the **qinv** row to see the results for this variable (Output of capital goods sector). Then, when you change the which solution list box to “**2 Pre num4p04**” you just see **n.a.** which means that these results are not available.

Then return to the Contents list and again select the linearized results by clicking on “**1 num4p04**” from the which solution list box.

To see the effects of another useful way of listing the variables in the *Contents* page, select *Format...* from ViewSOL's main menu and there click on *Arrange vectors by size AND set* (in the panel headed Vector options); then click *Ok* which will put you back to the Contents list. You should see something like:

Variable	Size	No.	Name
Macros	1	13	
Vectors size:1	MARG	2	
Vectors size:3	ACTS	5	
Vectors size:3	COMM	15	
Vectors size:3	REG	91	
Vectors size:5	ENDW	1	
afa	COMM*ACTS*REG	1	Composite intermed. input c augmenting tech ch. By act. a in r
pds	COMM*REG	1	price of domestically supplied commodity c in region r

(and so on).

To see the results for all variables with a single argument ranging over the set **REG**, double-click on the fifth of the rows above (the one with size “REG”). This time you will see the linear (percentage-change) results for the 91 such variables (including **qinv**, **pop** and **pgov**). Again you can see the other results by making a selection in the drop-down list box controlling which lot of results you are viewing (the one showing “**1 num4p04**” at present). When you have finished looking at these results, click on *Contents* to return to the Contents list.

When you have finished looking at the results, exit from ViewSOL.

Example AN3 - The Command file NUM4P04.CMF

The simulation itself was carried out in Example AN1. There the program GEMSIM took all of its inputs from the GEMPACK Command file NUM4P04.CMF. This file tells the program 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 NUM4P04.CMF via *File / Edit file...* from WinGEM's main menu.

Move around the file to see the various parts of the file. Lines beginning with an exclamation mark ! are comments which are ignored by the program but are vital in making this file intelligible to humans and a good record of the simulation. Find the places where

- (a) the closure is specified,
- (b) the shocks are specified,
- (c) the program is told which pre-simulation data files to read,
- (d) the name of the Solution file is specified,
- (e) the program is told which post-simulation data files to write.

There are other parts of the file which you will learn about later.

Example AN3A (optional) - A Command file for a numeraire decrease

Now suppose that you wanted to change this file so as to give a **5 percent decrease** to the numeraire. To prepare a Command file for this, use *File / Save as...* to change the name of the file you are editing to **NUM4P04X.CMF**. Then make appropriate changes so that it reduces the numeraire by 5 per cent. You will also need to make a few other changes. You should

- change the name of the Solution file to **NUM4P04X**,
- change the name of the updated file with logical name GTAPDATA to **NUM4P04X.UPD**,
- change the other occurrences of NUM4P04 to **NUM4P04X**, and
- should also look at the comments and the verbal description and change them to be compatible with your new shock.

When you have finished making the changes, exit from the editor. [If you are uncertain about your file NUM4P04X.CMF, you can compare it with the file **NM404XOK.CMF** we have prepared.]

You won't run this modified numeraire simulation using NUM4P04X.CMF here, but you will do this when you do Example AN6 below.

Example AN4 - Look at the updated data

As well as producing a Solution file containing percentage changes to prices and volumes, the simulation in Example AN1 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 AN1 above, this updated global data is in the file called **NUM4P04.UPD**. Like the pre-simulation GTAPDATA file DAT4P04.HAR, the updated GTAPDATA file **NUM4P04.UPD** is a Header Array file. This means that you need the help of a program (either ViewHAR or SeeHAR) to

look at the data in it. Below we show you how you can use ViewHAR to look at some of the updated data and compare it with the corresponding original data. [You could do much the same using SeeHAR.]

The 3x3 array of data at header "VMPB" contains the Value of Private household expenditure on each Imported commodity in each region, valued at basic prices.

Click on **View Input/Output Files** in the GEMSIM window left open from the simulation (Example AN1 above) and select **HA updated GTAPDATA C:\GTAP\NUM4P04.UPD**. This will open a ViewHAR window looking at the Contents list for this updated data. Double click on header **VMPB** to look at its data. To make this occupy just the **left** half of the screen, click (**carefully**) in the left side of the  icon near the bottom right-hand corner of the ViewHAR window.

Now, for the original data DAT4P04.HAR, go back to the GEMSIM window from which you ran the simulation and again click in **View Input/Output Files** and this time select **HA input GTAPDATA C:\GTAP\DAT4P04.HAR**. Again double-click to select the **VMPB** data and this time click carefully on the **right** side of the  icon to make this occupy the right-hand half of your screen.

Then (via the **Taskbar** or **Alt+Tab**), bring the other ViewHAR window (the one in the left hand half of the screen) to the top.

Now that you have both the original and the updated data on your screen, look at a particular cell, say that corresponding to food in SSA. Think what has happened to prices and to quantities in this simulation, and hence deduce what must have happened to values. Check that the pre- and post-simulation VMPB("food", "SSA") values are consistent with this.⁵¹ You might like to repeat this for some other cells in this and/or other headers. Close the two ViewHAR windows when you are finished.

Example AN5 (optional) - Run GTAPVIEW on the post-simulation data

The updated global data NUM4P04.UPD (see Example AN4 above) should satisfy the same balancing conditions (e.g., GDP should be the same from both sides) as the original global data DAT4P04.HAR. You can check this by carrying out the instructions in GTAPVIEW.TAB (see section A.2 above) on NUM4P04.UPD (rather than DAT4P04.HAR).

To do this, you first need to prepare a suitable Command file for running GEMSIM. Of course, the way to do this is to modify the Command file VIEW4P04.CMF used to carry out the GTAPVIEW calculations on the original data in DAT4P04.HAR. So choose **File / Edit file...** and open VIEW4P04.CMF. First **save** it **as** VWN4P04.CMF. Then edit this to change

- the name of the GTAPDATA file to read from DAT4P04.HAR to NUM4P04.UPD.
- the name of the output GTAPVIEW and LOG files to VWN4P04.HAR and VWN4P04.LOG respectively.
- the name of the output TAXRATES file to TXN4P04.HAR.
- the name of the file in the first line (a comment) to VWN4P04.CMF.

Then exit, saving the new VWN4P04.CMF.

⁵¹ The pre- and post-simulation values are approximately 4367.82 and 4804.60. Note that 10% of the former is 436.782. When you add this to the pre-simulation value you get 4804.60. Thus, as expected, the post-simulation value is 10% more than the pre-simulation value.

First run TABLO (via *Simulation / TABLO Implement...*) and *Select* GTAPVIEW.TAB and then click on **Run**. This will produce the GEMSIM Auxiliary files for GTAPVIEW.TAB.

Now run GEMSIM (via *Simulation / GEMSIM Solve...*) and *Select* this Command file VWN4P04.CMF. When GEMSIM has finished running, look at the output Header Array file (what is its name?) produced by running ViewHAR. In particular, look at the CURRENTACCT values. Did they change in the updated data?

Example AN6 (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 AN1 and AN2 above, but instead take inputs from the Command file NUM4P04X.CMF you prepared in Example AN3A above. Check that the results are as expected (namely no change in quantities and 5 percent reductions in prices and domestic dollar values).

Example AN7 (optional) - Numeraire simulation with a larger data set

In Examples AN1 and AN2 above you carried out a numeraire simulation (increasing **pfactwld** by 10 percent) with the 3x3 data. In this examples you will carry out the same simulation with the 7x5 data.

The easiest way to prepare a Command file for this simulation is to base it on NUM4P04.CMF used with the 3x3 data. The natural name for this new Command file is NUM4P05.CMF. You know how to do this by now (using *Edit file...* and *Save As...*). When you edit the new file, change **all** occurrences of "4P04" to "4P05" and change all occurrences of "3x3" to "7x5". No other changes are needed.

Then run GEMSIM as in Example AN1, though use the Command file NUM4P05.CMF instead of NUM4P04.CMF, and then run ViewSOL much as in Example AN2. Again check the results for variables **qo** and **ps**.

Close down WinGEM and all associated windows before proceeding.

AA.1 A Simulation Reducing One Distortion

In Examples AA1-AA4 below, you will carry out a simulation with the 3x3 data in which the import tariff on Food from Sub-Saharan Africa imported into the European Union (EU) is reduced by ten percent. (In the 3x3 data base, the power of this tariff is approximately 1.13, as you can see from the TMS_L values produced in Example A11 above.)

Start WinGEM running and check (via *File / Change both default directories*) that the directory is C:\GTAP.

Example AA1 - Preparing a Command file

Here you will prepare a GEMPACK Command file TMSFSE.CMF for carrying out this simulation. Start from NUM4P04.CMF used in the numeraire simulation, *Save As... TMSFSE.CMF* and then edit this to make the changes indicated below.⁵²

(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

⁵² The name **TMSFSE** is chosen as a reminder that this simulation is a **TMS** shock on **Food** from **Sub-Saharan Africa** to the **EU**.

tms(c,s,d)

means the percentage change in the power of the import tariff levied in destination region d on commodity c from source region s. Thus to reduce the power of the import tariff on Sub-Saharan Africa Food imported into the EU by 10 percent, you must specify a shock of -10 to

tms("food","SSA","eu"). Find the shocks part of the file TMSFSE.CMF and replace it by the statement

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

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

solution file = tmsfse ;

(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 ';'.
[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.]

(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 TMSFSE.UPD using the root TMSFSE 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 and Display file names.** Change the name of the LOG file to TMSFSE.LOG (again using the same root TMSFSE), and change the Display file name to TMSFSE.DIS.

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

When you have made these, exit from the editor, saving you changes. If you are uncertain about any part of your file TMSFSE.CMF, you can look in the file TMSFSEOK.CMF which we have prepared. [If, when you use your new TMSFSE.CMF, the program says there is an error which you can't detect, try using the file TMSFSEOK.CMF instead of TMSFSE.CMF.]

Example AA2 - Run GEMSIM and ViewSOL

Open a GEMSIM window as in Example AN1 above, **Select** the Command file TMSFSE.CMF and then **Run** the program. When it finishes, either run ViewSOL (as in Example AN2) to look at the results.

First look at the **qo** results. When you are looking at these in ViewSOL, notice the option on ViewSOL's tool bar of changing the number of decimal places displayed. [You can hover with your mouse over each of the items on the tool bar. Try the different ones until you find the one with hint "Set

decimal places”.] Select 3 decimal places. For example, check that the output of food in the SSA `qo("food","SSA")` increased by **3.116** percent.

Do not close ViewSOL when you have finished looking at the results.

Example AA3 - Changing the solution method

In Example AA2 above, the simulation program 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 AA2 above but this time using Gragg's method (the default in GEMPACK) to calculate the results.

To do this, you should copy (via *Save As...*) TMSFSE.CMF to TMSFSEN.CMF ("N" for nonlinear), then edit TMSFSEN.CMF as described below.

(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 GEMSIM 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 GEMSIM to create a so-called **Extrapolation Accuracy file** which you can look at to see if the results are sufficiently accurate.

(2) Change the names of the Solution file, updated GTAPDATA file and of the LOG and Display files to TMSFSEN.

Then exit from the editor, saving the changes to TMSFSEN.CMF.

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

Now rerun the simulation by running GEMSIM and taking inputs from TMSFSEN.CMF and then running ViewSOL.

Example AA4 - Comparing these results

To see how the Johansen solution and the one obtained in Example AA3 above using Gragg's method differ, you can compare the results in the Solution files TMSFSE.SL4 (Example AA2) and TMSFSEN.SL4 (Example AA3). To do this, return to ViewSOL (use the **Taskbar** or **Alt+Tab**) where you should still see the results in TMSFSE.SL4 (from Example AA2, where you left ViewSOL open).

One of the advantages of ViewSOL is that it can look at several different solutions at the same time. To see this, select *File... / Open...* and open TMSFSEN.SL4

Click on the **qo** row to see the results for variable **qo**. You will probably see from one of the boxes on the tool bar (the one with hint **Choose which solution to view**) that you are looking at the results in the TMSFSEN solution. Note the value for **qo("food","SSA")** in this. To compare with the value from the TMSFSE solution, click on the "Choose which solution to view" box on the tool bar and select the TMSFSE solution; you will immediately see the **qo** results from this solution. Note the difference between the **qo("food","SSA")** results in the two solutions.⁵³

To see how accurate the Gragg results are, look at the Extrapolation Accuracy file (it is a text file so you can use GEMEDIT to look at it) TMSFSEN.XAC produced in Example AA3. For example, the **qo** part of this starts with a line very similar to

```
6 3.83714 3.85428 3.85755 3.86020 CX 4 L5
```

These are respectively the 2-step, 4-step, 6-step and extrapolated results for component number 6 (Food in SSA) of **qo**. The annotation "CX 4" indicates that this percent change result is judged to be accurate to at least 4 figures and that we can be confident about the extrapolation. The annotation "L5" means that the corresponding result for the levels value of the output of food in SSA is accurate to at least 5 figures

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 TMSFSE.SL4 for the Johansen results and in TMSFSEN.SL4 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 \$US -619 million and the Johansen result is about \$US -207 million.

When you have finished, close ViewSOL (via *File / Exit*).

AA.2 A Simulation Removing Several Distortions with the 3x3 data

In Example AA5 below you will carry out a simulation removing all distortions on Food in the European Union. More precisely, you will remove

- 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 A12 above. The required shocks to remove all of these distortions are in the files

⁵³ In this example we want you to look at the simulation (that is, percentage change) results. Since there are also levels results (pre- and post-simulation and changes results), the simulation results for TMSFSE will be shown as "**1 tmsfse**" while those for the non-linear simulation TMSFSEN will be shown under "**5 tmsfsen**".

TO4P04.SHK, TX4P04.SHK and TMS4P04.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","food","eu") ! food commodity output of food activity in EU,  
txs("food","eu","SSA") and txs("food","eu","row") ! exports of food from EU,  
tms("food","SSA","eu") and tms("food","row","eu") ! imports of food into EU.
```

The first is achieved via the statement

```
shock to("food","food","eu") = select from file to4P04.shk ;
```

in a Command file; the "select from" tells GEMSIM to select just the shocks to the appropriate component of **to** from all the shocks in the file TO4P04.SHK. Similar statements achieve the desired shocks to the relevant components of variables **txs** and **tms**.

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

Example AA5 below relies on the three files TO4P04.SHK, TXS4P04.SHK and TMS4P04.SHK produced in Example A12 above. Before starting Example AA5, check that these three files are present by going to a DOS box (which you can do by selecting

File / Shell to DOS

in WinGEM's main menu. In this box, issue the command

```
dir t*.shk
```

Then issue the command **exit** to close the DOS box.

Example AA5 - Carry out this simulation

First run GEMSIM, taking inputs from the Command file CAPSIM.CMF. Once this finishes running, go on to ViewSOL. 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 SSA and ROW? Why?)

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

Example AA6 - Checking that the simulation has worked

Part of the output from the simulation is the updated global data file CAPSIM.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 perhaps 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 SHK4P04.CMF (which you used in Example A12 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 SHK4P04.CMF to a new file SHKCAP.CMF (use **File / Edit file...** and **Save As...**) and then edit SHKCAP.CMF. Make sure that CAPSIM.UPD is the global data set it reads (edit the "file gtapdata" statement) and change the names of the output .SHK files by replacing **4P04** by **CAP** in all their names (for example, "file TOHAT = tocap.shk ; "). 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 output files. For example, to see if output subsidies on food in the EU have been eliminated, look in the file **TOCAP.SHK** (via *View Input/Output Files*). The value here for (food,EU) is the shock required to eliminate output subsidies. This should be nearly zero (indicating that subsidies have already been eliminated); is this what you see? Repeat for the relevant values in TMSCAP.SHK and TXSCAP.SHK. [If any of these is not sufficiently close to zero, you could repeat the simulation in CAPSIM.CMF, this time increasing the number of steps to get a more accurate solution.]

REFERENCES

- Erwin L. Corong, Thomas W. Hertel, Robert A. McDougall, Marinos Tsigas, and Dominique van der Mensbrughhe (2017), 'The Standard GTAP Model, Version 7, *Journal of Global Economic Analysis*, vol 2 (2007), no. 1, pp. 1-119. [An online version is at <https://jgea.org/resources/jgea/ojs/index.php/jgea/article/view/47>]
- W. Jill Harrison and K.R. Pearson (1996), 'Computing Solutions for Large General Equilibrium Models Using GEMPACK', *Computational Economics*, vol. 9 (1996), pp. 83-127. [An online version is at <http://www.copsmodels.com/elecpr/ip-64.htm>]
- Thomas W. Hertel, ed. (1997), *Global Trade Analysis: Modeling and Applications*, Cambridge University Press, pp.403.
- W. Jill Harrison, Mark Horridge, Michael Jerie and K. R. Pearson (2015), *GEMPACK Manual*, If GEMPACK is installed, the complete GEMPACK manual is stored in the file `gpmanual.htm` in your GEMPACK folder (normally `C:\GP`). An online version is at <http://www.copsmodels.com/gpmanual.htm>