

**Installing and Using
the Source-Code Version of GEMPACK
on Windows PCs with Lahey or Intel Fortran**

GEMPACK Document No. GPD-6

**J. M. Horridge
M. Jerie
K. R. Pearson**

**Centre of Policy Studies and Impact Project
Monash University, Melbourne, Australia**

**Thirteenth edition
May 2008**

The current version of GEMPACK is Release 10.0 (May 2008).

For more information about GEMPACK, contact

The GEMPACK Business Manager
Centre of Policy Studies and Impact Project
Monash University
Clayton VIC 3800
Australia
email: info@gempack.com.au
Phone: (03)-9905-5484 or +61-3-9905-5484 from overseas
Fax: (03)-9905-2426 or +61-3-9905-2426 from overseas

Other information can be obtained from the GEMPACK Web site:

<http://www.gempack.com.au>

© Copyright 2008. CoPS/Impact and KPSOFT.

Thirteenth edition
May 2008

ISSN 1030-2514

ISBN 0-7326-1506-2

This is part of the documentation of the GEMPACK Software System for solving large economic models, developed at the Centre of Policy Studies/IMPACT Project, Monash University, Clayton VIC 3800, Australia.

Abstract

Windows PCs provide excellent platforms for doing serious general equilibrium modelling. This document describes how to install and use the source-code version of GEMPACK on a Windows PC. You need to previously install a recommended Fortran compiler:

Authors and Earlier Editions

Date	Author(s)	Comment
		[The first two editions of this were numbered GED-29.]
Feb 90	R.Walker, K.Pearson, G.Codsi	First edition (GED-29) [Release 4.1.01] [Title was "Installing and Using GEMPACK on PCs with Extended Memory".]
Sept 91	G.Codsi & K.Pearson	Second edition (GED-29) [Release 4.2.02] [Title was "Installing and Using GEMPACK on 386 DOS Machines with Extended Memory".]
Apr 93	J.Harrison & K.Pearson	Third edition (GPD-6) [Release 5.0] [Title was "Installing and Using GEMPACK on 80386 or 80486 DOS PCs with Lahey Fortran F77L-EM/32".]
May 94	J.Harrison & K.Pearson	Fourth edition (GPD-6) [Release 5.1]
Sept 96	J.Harrison & K.Pearson	Fifth edition (GPD-6) [Release 5.2] [Title now "Installing and Using GEMPACK on DOS, Windows or Windows 95 PCs with Lahey Fortran".]
Jan 97	J.Harrison & K.Pearson	Sixth edition (GPD-6) [Release 5.2] [Title now "Installing and Using the Source-Code Version of GEMPACK on DOS/Windows PCs with Lahey Fortran".]
Aug 97	J.Harrison & K.Pearson	Seventh edition (GPD-6) [Release 5.2-002]
Oct 98	J.Harrison & K.Pearson	Eighth edition (GPD-6) [Release 6.0]
Apr 99	J.Harrison & K.Pearson	Ninth edition (GPD-6) [Release 6.0-001]
Oct 00	J.Harrison & K.Pearson	Tenth edition (GPD-6) [Release 7.0]
Oct 02	J.Harrison & K.Pearson	Eleventh edition (GPD-6) [Release 8.0] [Title now "Installing and Using the Source-Code Version of GEMPACK on Windows PCs with Lahey Fortran".]
Apr 05	J.Harrison & K.Pearson	Twelfth edition (GPD-6) [Release 9.0]
May 08	J.M.Horridge, M.Jerie & K.Pearson	Thirteenth edition (GPD-6) [Release 10.0] [Title now "Installing and Using the Source-Code Version of GEMPACK on Windows PCs with Lahey or Intel Fortran".]

Table of Contents

1	THE WINDOWS PC SOURCE-CODE VERSION OF GEMPACK	1
2	PREPARING TO INSTALL GEMPACK	3
2.1	System requirements	3
2.2	Preparing your PC	3
2.2.1	Set the Environment variable TMP	3
2.2.2	If you have an earlier release of GEMPACK installed	4
2.2.3	Locating your licence file	4
2.2.4	Security and permissions	4
2.2.5	If you are installing on a network	5
3	CHOOSING AND INSTALLING YOUR FORTRAN COMPILER	7
3.1	FORTRAN compilers	7
3.1.1	Supported Lahey compilers	7
3.1.2	Supported Intel compilers	8
3.2	Installing a Lahey Fortran compiler	9
3.2.1	Testing the installation of LF95	9
3.3	Installing the Intel Fortran compiler	9
3.3.1	Installation check list	9
3.3.2	Installation steps	10
3.3.3	Editing the System PATH environment variable	12
3.3.4	Testing the installation of your Intel Fortran compiler	13
4	INSTALLING GEMPACK	15
4.1	Installing	15
4.1.1	If an error occurs	17
4.2	After a successful build, fix the PATH and GPDIR	17
4.2.1	Checking and setting PATH and GPDIR	17
4.3	If you installed in a new directory (not C:\GP)	18
4.4	GEMPACK licence	18
5	TESTING THE INSTALLATION	19
5.1	Checking the DOS Path and access to your GEMPACK licence	19
5.1.1	If you still have problems	19
5.2	Making a directory for the Stylized Johansen model	20
5.3	Simulations to test GEMPACK and WinGEM	20
6	WORKING WITH GEMPACK ON A PC	21
6.1	Suggestions for hands-on computing	21
6.1.1	Using WinGEM	21
6.1.2	Not using WinGEM	21
6.2	New model's directory location	21
6.3	For speed, use TABLO-generated programs	21
6.4	Text editor	21
6.5	If a program runs out of memory	22

6.6	Fortran compiler options	22
6.6.1	Optimization levels	22
6.6.2	Compiling TABLO-generated programs	23
6.7	File association	23
6.8	Keep and Temporary directories and GEMPACK Windows programs	24
6.9	Copying GEMPACK programs to other PCs	24
6.9.1	Windows programs	24
6.10	Accessing the CoPS web pages from the CD	25
6.11	Using GEMPACK in a teaching situation	25
7	TECHNICAL TOPICS	27
7.1	Working in a DOS box	27
7.1.1	Compiling and linking TABLO-generated programs	27
7.1.2	TABLO-generated programs	27
7.1.3	Running GEMPACK programs in a DOS box	28
7.1.4	Interrupting programs	28
7.1.5	Controlling screen output	28
7.1.6	Using stored-input files in a DOS box	29
7.1.7	Use a log file to see screen output	29
7.2	DOS batch files	29
7.2.1	Trapping for LTG errors in DOS batch files	30
7.2.2	Making executable images of GEMPACK programs in a DOS box	31
7.3	Running BuildGP	31
7.4	FLIB_DVT_BUFFER environment variable	31
7.5	Variants of LTG	32
7.5.1	Debug LTG: LTGDEB.BAT (LF95 and LF only)	32
7.5.2	Keep/Delete Fortran LTG: LTGKPFOR.BAT and LTGNOFOR.BAT (all compilers)	32
7.5.3	No Modules LTG: LTGNOMOD.BAT and LTGFNMOD.BAT (all compilers)	32
7.6	Uninstalling GEMPACK	33
7.7	Compiler options – Technical information	33
7.7.1	Compiler options for the Intel compilers	33
7.7.2	Compiler options for Lahey/Fujitsu LF95 or LF Version 7 compiler	33
7.7.3	Compiler options for Lahey compiler LF90	34
7.7.4	Changing the optimization setting (all compilers)	34
7.8	Advanced compiler use	34
7.8.1	Changing the stack size of TABLO-generated programs	34
7.8.2	Maximum amount of memory available to programs	35
8	INSTALLING GEMPACK ON A NETWORK	37
9	OLDER VERSIONS OF WINDOWS	39
9.1	Setting the TMP directory	39
9.1.1	PATH and GPDIR changes in AUTOEXEC.BAT (Windows 95, 98 or ME)	39
10	REFERENCES	41
11	GEMPACK DOCUMENTS	41
12	INDEX	43

CHAPTER 1

1 The Windows PC Source-Code Version of GEMPACK

This document GPD-6 tells you how to install the Release 10 Source-Code version of GEMPACK on a PC which is running Windows XP or Vista¹ and has a Fortran compiler installed. Suitable compilers are

- the Lahey compilers LF90, LF95 or LF, or
- the 32-bit or 64-bit Intel Visual Fortran compiler IFORT.²

This document also covers machine-specific information about working with GEMPACK on a PC.

More user documentation can be found in the other GEMPACK documents³. Start with GEMPACK document GPD-1 *Introduction to GEMPACK*. You can carry out the installation and testing of GEMPACK on a PC, as described below, without being familiar with GEMPACK. However, we recommend you quickly read chapters 1 to 3 in GPD-1 before attempting any GEMPACK modelling on your PC.

An introduction to the different GEMPACK programs can be found in section 1.7 of GPD-1, while a guide to the models supplied with GEMPACK is given in section 1.8. A guide to the full user documentation can be found in chapter 6 of GPD-1. Hands-on examples can be found in chapters 2, 3 and 4 of GPD-1 and in GEMPACK document GPD-8.

The documentation for Release 10.0 of GEMPACK is in GPD-9. Section 1.3 of GPD-9 contains a guide to the Release 10.0 documentation.

¹ For Release 10.0, Windows 98, ME, NT, 2000 are no longer officially supported – we do not test on these OS. However, you may well find that you can run GEMPACK on these OS. We provide some notes in Chapter 9.

² The Intel compilers were not supported before Release 10 of GEMPACK. You need a 64-bit version of Windows to use the 64-bit compiler.

³ References to GEMPACK documents identify the document by GEMPACK Document (GPD) number, rather than by author or date. References are always to the version of the document which is current at the date of issue of the cross-referencing document. The GEMPACK documents are listed in a separate References section near the end of this document. Comments from readers on this or any of the GEMPACK documents, either pointing out errors, inaccuracies, omissions or obscurities, or making other suggestions for improvements, will be welcomed. Please email such comments to support@gempack.com

CHAPTER 2

2 Preparing to install GEMPACK

2.1 System requirements

For the Source-Code Version of GEMPACK, you must have at least the following.

1. A PC running Windows XP or Vista.
2. A suitable Fortran compiler, such as either of the Fujitsu-Lahey compilers LF95 or LF, or even the older Lahey Fortran compiler LF90, or the 32-bit or 64-bit Intel Fortran compiler for Windows. The next chapter contains more information about choosing and installing your Fortran compiler. **Note that the compiler MUST be installed on your machine BEFORE you install GEMPACK.**
3. Hard disk, requiring perhaps 1GB free to install and test GEMPACK and Fortran. To see how much free disk space you have, right-click on the icon for your hard drive in "My Computer".
4. At least 512 MB of memory (RAM). Click on **Help | About** in the main menu of **My Computer**. This will tell you how much physical memory is available to Windows. The amount of memory you have limits the size of models you can build. Many models will use less than 128 MB of RAM. Recent versions of Windows such as XP or especially Vista themselves use up much memory. And to do useful things with GEMPACK, you'd very often need to have several GEMPACK Windows program and perhaps an MSOffice application open at the same time. Each running program uses up RAM.

2.2 Preparing your PC

2.2.1 Set the Environment variable TMP

Many programs, including GEMPACK programs and Fortran compilers, use a work directory to create temporary files. The location of this directory is determined by the TMP Environment variable.

A number of problems may arise if the TMP directory name contains spaces or non-English characters, or is very long. In particular, the two Lahey compilers LF90 and LF95 use the TMP directory to write temporary files while compiling and linking. If this directory contains a space or non-English characters, the compiler will not work and stops with an obscure message.⁴

Follow the directions below to set your Environment variable TMP to some simply-named directory (for example C:\TEMP).

- **If you are running Windows XP**

You can right-click on "My Computer" and select "Properties | Advanced | Environment Variables" and then edit the User Environment variable called TMP to be C:\TEMP.

Create a new directory C:\TEMP if you do not already have one.

- **If you are running Windows Vista**

You will need administrator access.

⁴ LF90 or LF95 produces the compiler error "ABORT – Cannot recognize command line".

Select in turn: Control panel | Classic view | System | Advanced system settings | Environment Variables

Edit the User Environment variable called TMP to be C:\TEMP.

Create a new directory C:\TEMP if you do not already have one.

2.2.2 If you have an earlier release of GEMPACK installed

You can choose where to install GEMPACK, but the best plan is to use the default directory C:\GP.

If you have an earlier release of GEMPACK installed in C:\GP, you should probably leave it on your hard disk until you have successfully installed and tested Release 10.0 (in case an unexpected problem occurs). You should rename the directory containing the previous version, so you can install Release 10.0 of GEMPACK in C:\GP.

For example, if you currently have Release 9.0 GEMPACK in C:\GP, rename that directory to, say, C:\GP90, and install Release 10.0 into C:\GP. If you use the same GEMPACK directory as before, other GEMPACK-related Windows programs such as RunDynam and RunGTAP will automatically use the latest version of GEMPACK.

2.2.3 Locating your licence file

The installer will ask for the location of your Release 10 Source-code licence file, suffixed GEM [a Release 9 licence will not work]. Probably you were emailed such a file (perhaps enclosed in a zip attachment). If so, find and extract it now.

2.2.4 Security and permissions

Recent Windows versions such as Windows XP and Vista have introduced different User Accounts and different grades of Users. The simplest way of operating your computer is when you are the sole user of your computer and have full Administrator/Power User rights when using it to run GEMPACK.

To install GEMPACK, you may need Administrator or Power User privileges. Otherwise, the installer may be unable to change the PATH variable as part of the install procedure. Under Windows Vista, Administrator access will be needed to start the installer. In a work environment, you may need IT support personnel to supply [the password for] such access.

2.2.4.1 Notes for IT support:

You can install GEMPACK as Administrator, and run as Limited or Standard user, as long as permissions are set to allow Limited or Standard users to access needed files. Users of GEMPACK need to be able to read and execute the programs in the GEMPACK directory. In their working directories, Limited Users need full rights to read, write, execute and delete files.

The software requires that users be able to open and use a command prompt (cmd.exe) window.

Under Windows XP or Vista, you may find that, if one user has copied files to a directory for use by another user, the files "belong" to the first user and can not be deleted by the second user. If the second user tries to write to a file of the same name as one belonging to the first user, the write fails because the GEMPACK program is unable to delete the old file before creating a new version with the same name.

Users may sometimes need to alter or create files in the GEMPACK directory. Hence, we **do not recommend** that you install GEMPACK under C:\Program Files. Under Vista and some XP configurations, write/modify access is prohibited there.

Please assist the user by switching off “hide file extensions” (From Explorer, Tools...Folder Options...View).

2.2.5 If you are installing on a network

If you are installing GEMPACK on a network please read chapter 8 before proceeding.

CHAPTER 3

3 Choosing and installing your Fortran compiler

3.1 FORTRAN compilers

For Release 10, we support Lahey (see section 3.1.1) and Intel compilers (section 3.1.2). Both products offer both academic and multi-user pricing and both are distributed by leading Fortran vendor Polyhedron or in Australia by Devin Trussell of Computer Transition Systems, see below for contact details. Alternatively you may also obtain either product through the publishers website.

When choosing a Fortran compiler to use with Release 10 Source-code GEMPACK it is natural to ask which one is better. For Release 10 Source-code GEMPACK we recommend the Intel Visual Fortran compiler as the best all round choice. The speed of a GEMPACK program varies with the application and Fortran compiler used. The speed of the Intel compiler compares favourably with the Lahey alternatives. In addition, the Intel compiler offers more flexibility in developing applications which require access to large amounts of memory, now more commonly available on modern computing hardware. For more details on access to large memory see the discussion of the 64-bit Intel compiler in chapter 2 of GPD-9. Furthermore, the Intel compiler is being actively developed and maintained, whereas the Lahey Fortran products although maintained are not being actively developed for the Windows platform. Hence

**if you are buying a new Fortran compiler for use with GEMAPCK,
we recommend Intel Visual Fortran.**

Australian and international distributors of Fortran compilers are

(Australian)

- Computer Transition Systems
<http://www.cts.com.au>
info@cts.com.au

(International)

- Polyhedron Software Ltd
<http://www.polyhedron.com/fortranmain>
sales@polyhedron.com
- Lahey Computer Systems Inc.
<http://www.lahey.com>
- Intel®
<http://www.intel.com/software/products>
(Hint: choose Compilers to locate the Fortran pages.)

3.1.1 Supported Lahey compilers

We support two Lahey Fortran compilers for Windows⁵:

LF90. We support only the final version 4.50i. [If you have an earlier maintenance update of version 4.50 – for example, version 4.50c – you should go to the Lahey web site <http://www.lahey.com> and

⁵ With versions of GEMPACK prior to Release 8.0, it was possible to use Lahey Fortran 77 compiler F77L-EM/32 compiler that is called F77L3. However, since Release 8.0 (or later) this F77L3 compiler is **no longer supported**. F77L3 does not meet modern Fortran standards; does not support allocatable arrays or filenames, and does not work with Windows XP or later.

download (there is no cost) the final maintenance update for LF90 which is version 4.50i.]⁶

From Release 10 Source-code GEMPACK, LF90 is deprecated⁷ and will not be supported by Release 11 Source-code GEMPACK.

We recommend that LF90 users upgrade as soon as possible to the Intel Fortran compiler (see section 3.1.2). See section 7.1 of GPD-9 for more details.

LF95. We support only versions 5.60 and later. On the Lahey web site, versions 5.60 and 5.70 are referred as **Lahey/Fujitsu Fortran 95** while versions 7.0 and later are referred to as **Lahey/Fujitsu Fortran** (95 is not added) or just **LF Fortran**.

- The final maintenance updates for versions 5.60 and 5.70 are versions 5.60h and 5.70f respectively. [If you have an earlier maintenance update of version 5.60 or 5.70 – for example, version 5.60d – you should go to the Lahey web site <http://www.lahey.com> and download (there is no cost) the final maintenance update for your version.]
- Any of the versions of LF (the current ones are versions 7.0 and 7.1) are suitable. However, you should go to the Lahey web site to download and use the latest maintenance update for your version.

We sometimes use LF95 in this document to refer to both LF95 and the more recent LF.

Various editions of Fujitsu/Lahey LF95 and LF compilers are available. If you are just using LF95 or LF for GEMPACK, you can use the least expensive one called **LF95 Express** or **LF Express**. If you are developing programs in Fortran as well and need to debug programs, you may wish to buy Standard LF95 or LF95 Pro.

3.1.2 Supported Intel compilers

We support the following two versions of the Intel Visual Fortran Compiler IFORT:

- Intel 32-bit compiler for IA-32 systems. This version of the compiler runs on 32-bit Windows XP or Vista. It produces 32-bit EXEs. Below we refer to this compiler as **IF32**.
- Intel 64-bit compiler for EM64T-based applications. This version of the compiler requires 64-bit Windows XP or Vista.⁸ Below we refer to this compiler as **IF64**.

We have tested several versions of these compilers. We only support version 10.0 or later versions. At the time of writing the most recent version available is 10.1. We strongly recommend version 10.1 of the compiler over previous ones as it allows a greatly simplified installation.⁹

⁶ Note that you cannot use Essential Lahey Fortran 90 (ELF 90); you must use the full LF90 compiler.

⁷ Binary files (such as GEMPACK Header Array files – see section 3.1 of GPD-4) produced by LF90 programs are not fully compatible with those produced by more recent Intel or Lahey compilers. For example, an LF95 program needs to first convert (by creating a temporary copy) a file created by an LF90 program. This happens automatically, but there is a small speed penalty. See chapter 15 of GPD-4 for more details.

⁸ There is also an Intel 64-bit compiler for PCs which have the Intel Itanium chip. Few users will have Itanium chips – they do not emulate standard 32-bit applications (for example, Microsoft Office) as well as the EM64T/AMD64 chips. Consequently we offer only marginal support for the Itanium environment.

⁹ Versions of Intel Visual Fortran prior to 10.1 have an installation prerequisite requirement. The prerequisite varies according to the host operating system, and your choice of compiler. Installations under Windows XP 32-bit or Vista 32-bit require a suitable version of Microsoft Visual Studio (or equivalent). Installations under Windows XP x64 or Vista 64-bit require the Microsoft platform Software Development Kit (SDK) which is

Intel Visual Fortran 10.1 does not require any installation prerequisites; it is a package which includes Microsoft Visual Studio 2005 Premier Partner Edition (MSVSPPE). The Intel Visual Fortran installer will install MSVSPPE for you if needed. MSVSPPE has a service pack available from Microsoft which is compulsory for installations under Vista and recommended for installations under XP. More details on obtaining the required service packs are given in section 3.3.1.

3.2 Installing a Lahey Fortran compiler

Before installing GEMPACK, install your Fortran compiler LF90, LF95 or LF: select a *typical installation*.

If you change your compiler, for example from LF90 to LF95, you need to reinstall the Source-code files again. Different files are installed for the different compilers.

3.2.1 Testing the installation of LF95

Edit and save (using a text editor) the following simple Fortran program HELLO.F90. It has only 3 lines.

```
write(*,*) 'Hello world'  
stop  
end
```

Open a DOS box in the directory in which you saved HELLO.F90.

Enter the command

```
lf95 hello.f90
```

This should produce the file HELLO.EXE.

Then run this HELLO.EXE by typing in the command

```
hello.exe
```

You should see the words "Hello world" appear.

Now go to chapter 4 to begin installation of GEMPACK.

3.3 Installing the Intel Fortran compiler

In the sections below we give instructions for installation of Intel Visual Fortran 10.1. However the product is constantly evolving and we ask that you visit our web page at

<http://www.monash.edu.au/policy/gpifort.htm>

for the latest installation instructions before proceeding.¹⁰

Instructions for installing earlier supported versions of the compiler (9.0, 9.1 and 10.0) are also provided at the above address.

The instructions below relate to Version 10.1 (or later) only.

3.3.1 Installation check list

For the installation you should have the following:

free from Microsoft. For more details on installation requirements and suitable versions of Microsoft Visual Studio please consult the Intel Visual Fortran Installation Guide that accompanies your product.

¹⁰ On that web page will be clear instructions as to whether or not you should follow the detailed instructions in this document. If modifications are required, details of those will be given on the web site.

- The Intel Visual Fortran install executable from Intel: **w_fc_p_10.1.xxx.exe** (approx. 300 MB) which you either downloaded or have on CD. [Here “xxx” is the subversion number – for example “014” which means version 10.1.014.]
- Your Intel Fortran licence file or serial number.
- The Intel Visual Fortran for Windows Installation Guide.
- (Recommended for XP installations.)
(Required for Vista installations.)
Microsoft® Visual Studio® 2005 Team Suite Service Pack 1
VS80sp1-KB926601-X86-ENU.exe (approx. 432 MB)
Download from <http://www.microsoft.com/downloads> by copying the above title into the download search field.
- (Required for Vista installations only.)
Visual Studio 2005 Service Pack 1 Update for Windows Vista
VS80sp1-KB932232-X86-ENU.exe (approx. 29 MB)
Download from <http://www.microsoft.com/downloads> by copying the above title into the download search field.

Your installation procedure will depend on your version of Windows and choice of Intel Fortran compiler. The installation steps given in the next section clearly indicate the appropriate choices for each combination.

3.3.2 Installation steps

Please print out and read this section before beginning the installation procedure. This procedure is for

- installing Intel Visual Fortran 10.1 for IA-32 applications with MSVSPPE 2005 under Windows XP or Windows Vista, or,
- installing Intel Visual Fortran 10.1 for Intel(R) 64 (sometimes referred to as EM64T) applications Windows XP x64 or Windows Vista 64-bit.

The installation procedure we describe here is straight forward and varies only slightly from the defaults suggested by the Intel install program.

1. Start the install by running the **w_fc_p_10.1.xxx.exe** install program.
2. Click through the installation procedure accepting the default settings until you reach the "Setup Type" dialogue where the choice you are presented with is Complete or Custom. Select Custom and click Next.
3. On the following "Choose Destination Location" dialogue click Next.

4. For 32-bit installations complete step 4.1, for 64-bit installations complete step 4.2.
 - 4.1. **32-bit Installations.** On the "Select Features" dialogue alter the default selections as follows. Expand the item "Intel(R) Visual Fortran Compiler" by clicking on the adjacent + symbol, and clear the box beside "Compile applications for Intel(R) 64". Also clear the box adjacent to "Intel(R) Debugger". Your selection should look like **Figure 1**. Go to step 5.

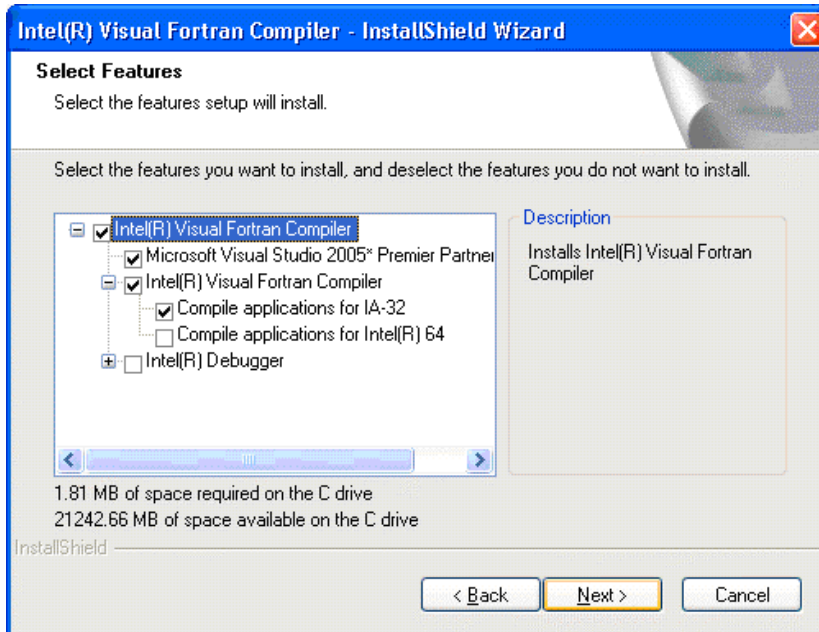


Figure 1: Component selection for Intel Fortran targeting IA-32

- 4.2. **64-bit Installations.** On the "Select Features" dialogue alter the default selections as follows. Expand the item "Intel(R) Visual Fortran Compiler" by clicking on the adjacent + symbol, and clear the box beside "Compile applications for IA-32". Also clear the box adjacent to "Intel(R) Debugger". Your selection should look like **Figure 2**. Go to step 5.

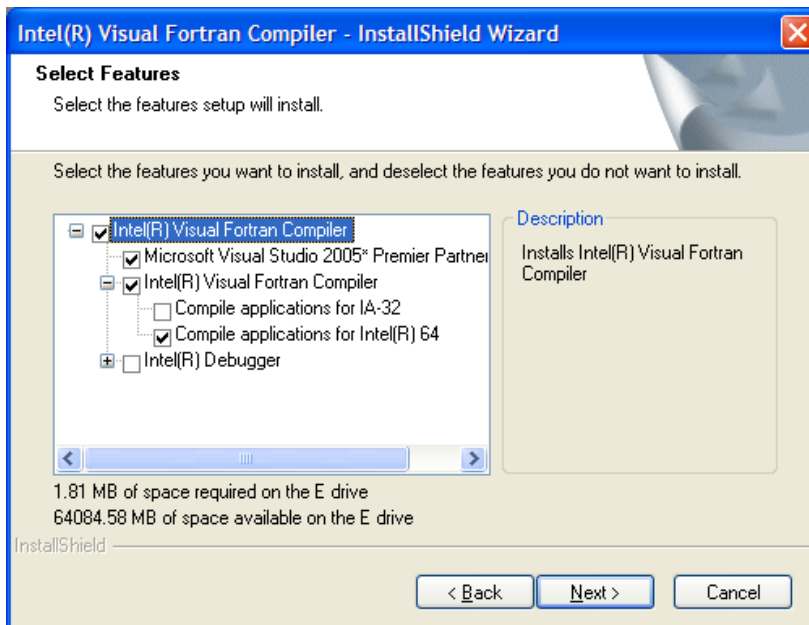


Figure 2: Component selection for Intel Fortran targeting Intel® 64

5. Confirm your selections and click Next.
6. At the "Ready to Install" dialogue, click Install.

7. A sequence of dialogues now follows. You will need to accept the MSVSPPE licence agreement, click through the subsequent dialogues accepting the defaults. If the install program detects that your Windows installation does not already have "Microsoft .NET Framework 2.0" installed you will be prompted through the .NET installation. Several progress dialogues follow indicating that the Intel Fortran compiler components are being installed.
8. **Vista installations only.** [Go on to point 9 if not using Vista.] If you are installing under Vista you will see during the installation step installing "Integrations in Microsoft Visual Studio" the warning message in **Figure 3** below

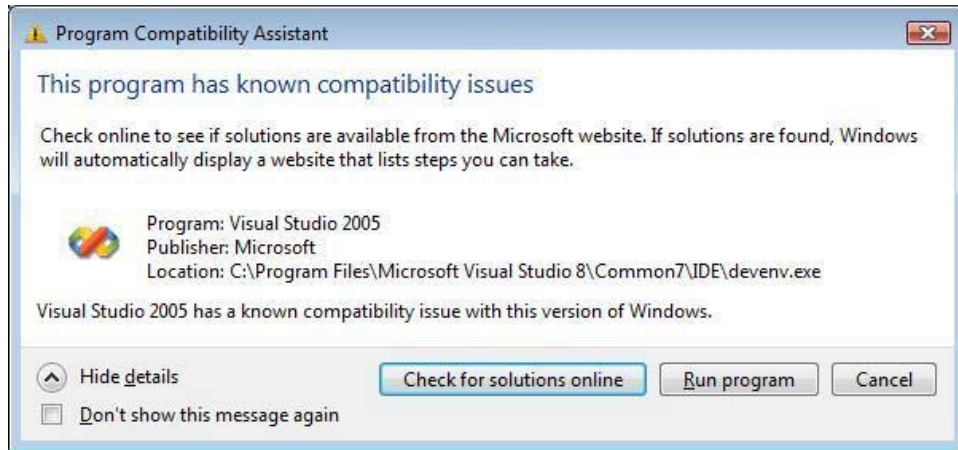


Figure 3: Vista installation warning message

Click Run program and complete the installation. This issue will be taken care of by installing the two service packs that you have already downloaded in the subsequent steps.

9. Eventually you will see "Installation Complete". Click Finish.
10. Installation of the Visual Studio service pack is compulsory for installations on Windows Vista, and is recommended for Windows XP. To install this service pack, run the installation file **VS80sp1-KB926601-X86-ENU.exe** which is your downloaded copy of Microsoft® Visual Studio® 2005 Team Suite Service Pack 1. The service pack is quite large and may take a few minutes to install.
11. **Vista installations only.** Installation of a second Visual Studio service pack is compulsory on Windows Vista. [Skip this step if you are installing on Windows XP.] To install this service pack on Vista, run the installation file **VS80sp1-KB932232-X86-ENU.exe** which is your downloaded copy of Visual Studio 2005 Service Pack 1 Update for Windows Vista.

When all installations are finished go to section 3.3.3 below to edit the System Path variable.

If you change your compiler, for example from IF32 to IF64, you need to reinstall the Source-code files again. Different GEMPACK source code files are installed for the different compilers.

3.3.3 Editing the System PATH environment variable

This section gives detailed instructions for editing the Windows system path variable. You must complete this section in order for Release 10 Source-code GEMPACK to work with Intel Visual Fortran.

Open the System Properties dialogue as follows:

- **If you are running Windows XP**

You can right-click on "My Computer" and select "Properties | Advanced | Environment

Variables"

and then select the **Path** System Environment variable, click Edit, and follow the instructions below to change its value.

- **If you are running Windows Vista**
You will need administrator access.

Select in turn: Control panel | Classic view | System | Advanced system settings | Environment Variables

and then select the **Path** System Environment variable, click Edit, and follow the instructions below to change its value.

The change you need to make to the Path variable value depends on which version of the compiler you have installed.

- **32-bit installations.** Users who installed the 32-bit (IA-32) compiler will find the following string in the variable value field (perhaps at the end):

```
;%IFORT_COMPILER10%\IA32\Lib
```

Using the mouse select and copy this string (including the semi-colon), and then paste a copy after the Lib. On the new copy change **Lib** to become **Bin** (only edit the new copy of Lib). Your path field should then contain this string

```
;%IFORT_COMPILER10%\IA32\Lib;%IFORT_COMPILER10%\IA32\Bin
```

- **64-bit installations.** Users who installed the 64 bit (Intel(R) 64 also known as EM64T) compiler will find the following string in the variable value field (perhaps at the end):

```
;%IFORT_COMPILER10%\em64t\Lib
```

Using the mouse select and copy this string, and then paste a copy after the Lib, then on the new copy change **Lib** to become **Bin** (only edit the new copy of Lib). Your path field should then end like this

```
;%IFORT_COMPILER10%\em64t\Lib;%IFORT_COMPILER10%\em64t\Bin
```

Click Ok to accept the changes, and Ok again to close the Environment Variables and System Properties dialogues.

3.3.4 Testing the installation of your Intel Fortran compiler

Edit and save (using a text editor) the following simple Fortran program HELLO.F90. It has only 3 lines.

```
write(*,*) 'Hello world'  
stop  
end
```

Open a command prompt (DOS box) in the directory in which you saved HELLO.F90. Enter the commands¹¹:

```
ifortvars
```

```
ifort hello.f90
```

This should produce the file HELLO.EXE.

Then run this HELLO.EXE by typing in the command

```
hello.exe
```

You should see the words "Hello world" appear.

¹¹ The ifortvars command is needed to temporarily set environment variables which tell the Intel compiler where to find certain library files. You normally do not need to run ifortvars since it is run automatically from within LTG. But you might need to run it if (as in this test) you are using the Intel compiler outside of GEMPACK.

After installing Intel Fortran, you may see a DOS box icon on your desktop called "Intel Build Environment". This is just a command prompt which automatically runs ifortvars on startup.

CHAPTER 4

4 Installing GEMPACK

All components of Source-code GEMPACK are contained in a single large install program

GPSC-10.0-000-install.exe

[After the initial release, more recent install programs will be labelled GPSC-10.0-00X-install.exe where X is 1,2,3 etc.] The package includes:

- the Source-code version of GEMPACK.
The install procedure puts Source-code (Fortran) files into your GEMPACK directory, and builds libraries and executable images of the various Fortran-based GEMPACK programs.
- Windows GEMPACK programs:
 - WinGEM the Windows interface to GEMPACK.
 - ViewHAR used to view and modify data on GEMPACK Header Array files.
 - ViewSOL used for viewing Solution files (results of simulations).
 - TABmate an editor for modifying and debugging TABLO Input files.
 - RunGEM used for carrying out simulations with models and doing sensitivity analysis.
 - AnalyseGE used for analysing simulation results.
- many examples of models built using GEMPACK.
- electronic versions of the GEMPACK user documentation (PDF files).

Install instructions are given below. But before you install GEMPACK, you must install your Fortran compiler, as explained previously.

4.1 Installing

Exit from all Windows programs before beginning this install procedure.

1. Source-Code GEMPACK is usually supplied on a CD. If you have a CD insert the CD¹² into your CD drive, locate the install program¹³
GPSC-10.0-000-install.exe
on the CD, and double-click it. If you received GEMPACK by some other method, for example by download, locate the install program you downloaded and double-click it. The installation program will begin. The following steps take you through the install program dialogue windows.
2. **Welcome and Copyright warning.** To agree to the copyright conditions click Next.
3. **Destination Location.** Here you choose where GEMPACK will be installed; we refer to this directory as the **GEMPACK directory**. We recommend that you accept the suggested **C:\GP** directory. You may choose another existing or new directory by clicking the Browse button. If you do so, when you return to the original screen check carefully that the directory name is what you want. Sometimes the install program adds \GP to the end of the name you have selected. In

¹² If your PC has no CD drive, you will need to first copy the install files to another drive, eg, a USB drive or a network drive.

¹³ After the initial release, more recent install programs will be labelled GPSC-10.0-00X-install.exe where X is 1,2,3 etc.

choosing a directory name, bear in mind that:

- GEMPACK programs **do not handle non-English characters** (see sections 5.9 and 5.9.3 in GEMPACK document GPD-1 for details).
 - Recent GEMPACK programs can handle long file or directory names, and names containing spaces. However, very long (or deeply-nested) directory names, or names containing characters outside [A-Z, 0-9] can occasionally cause problems, especially for older GEMPACK programs. We advise a simple directory name, such as C:\PROGS\GEMPACK
 - **Under Vista do not install in the Program Files directory.** Vista prevents you changing files there.
4. **Changes to your PATH and Environment.** The Install program can make changes to your PATH and the Environment variable called GPDIR. We recommend that you agree to these changes. Alternatively, you may make these changes yourself: see section 4.2 about how to change your PATH and set GPDIR.
 5. **Selecting a GEMPACK licence file (optional).** To run Release 10 Source-code GEMPACK you need a Release 10 Source-code GEMPACK licence file LICEN.GEM located in the GEMPACK directory (the destination location chosen at step 3 above is the GEMPACK directory). If the Installer does not find an existing LICEN.GEM in the GEMPACK directory you may click Browse to select your licence file for installation. GEMPACK licence files have a .gem file extension. Possible licence file locations are:
 - in a GEMPACK directory from a previous install of Release 10 Source-code GEMPACK (note a Release 9 or earlier licence will not work),
 - on the Install CD,
 - the location you saved your email licence file attachment to. If you recently purchased GEMPACK, we usually e-mail you a licence file inside a zip archive e-mail attachment. First extract your licence file from the zip archive before selecting it.

The installer puts a copy of your licence file, if necessary renaming it to LICEN.GEM, in your GEMPACK directory.

If you do not have a licence file right now, you can easily put it into your GEMPACK directory later. Simply rename it to LICEN.GEM and copy it into the GEMPACK directory.

If the Install program finds an existing LICEN.GEM in your chosen GEMPACK directory, it will not overwrite it, and the Browse button will be disabled. If the existing licence file is an old or wrong licence it is **your responsibility** to later place the correct Release 10 Source-code licence file into your GEMPACK directory, renamed if necessary to LICEN.GEM.

See section 1.2 of GEMPACK document GPD-9 for more details about GEMPACK licences.

6. **Select Fortran Compiler.** Indicate which Fortran compiler you will use. This compiler should be **installed and working**, as described in chapter 3 above.
7. **Ready to begin installation.** You may review your settings, click Back to make changes. Click Next to begin the installation. The Install program copies many files to your GEMPACK directory. Leave the CD in the CD drive until the installation is complete.
8. **Continue with compiling libraries and executable images.** After file copying is finished you are prompted to launch BuildGP by clicking **Next**. When you click on Next, the install program will exit and the BuildGP program will be launched. BuildGP builds the GEMPACK libraries and executable images of the Fortran-based GEMPACK programs. This will take between 5 and 20 minutes, depending on the speed of your machine.

If all goes well, you will eventually see a message saying that the libraries and images have been built successfully. In that case, just click **OK** and go on to section 4.2.

4.1.1 If an error occurs

If an error occurs during this build process, you will be told the name of the procedure when the error occurred. We suggest that you note this name on paper, then exit from BuildGP.

Usually an Error log is shown. This should give you some idea what is happening.

Possible Checks and Actions to try:

1. Check that there is plenty of room on your hard disk.
[If not, you will need to delete some files to create space.]
2. Check that Fortran is installed and on your path: go to a DOS prompt and type in LF90 or LF95 or IFORT (whichever is your compiler) and see if you get some output from your compiler about Usage.
3. Check that the TMP environment variable is set properly as described in section 2.2.1.
4. Check that the name of the GEMPACK directory does not contain spaces or Chinese or other non-English characters. See sections 5.9 and 5.9.3 in GEMPACK document GPD-1 for details.
5. Then reboot your computer.
6. Install again from the CD.
Or, you could try re-running just the final, BuildGP, stage. To do this, open a DOS box in your GEMPACK directory and type "BuildGP". Check the GEMPACK directory and compiler choices, then click "Start Build".
7. If an error reappears, please notify support@gempack.com. If an Error log has been created, please send it with details of what happened.

BuildGP checks that there is enough free disk space on the drive you chose for the GEMPACK directory. You can override these checks if you disagree or you can exit from BuildGP, clear some more disk space and then rerun BuildGP. You can read more about BuildGP in section 7.1.

4.2 After a successful build, fix the PATH and GPDIR

For GEMPACK to run, you must make sure that the GEMPACK directory (usually C:\GP) is on your PATH, and that the Environment variable GPDIR is set to the GEMPACK directory.

If you allowed the installer to make changes to your PATH and the Environment variable GPDIR (see section 4.1 above), this has already been done, so go on to section 4.3.

If you did not allow the installer or BuildGP (as appropriate) to make changes to your PATH and GPDIR (see section 4.1 above), you must make the required changes yourself, as described next.

4.2.1 Checking and setting PATH and GPDIR

The usual method to change the PATH and set the Environment variable GPDIR is by editing the System Properties. The important changes are that

- the directory into which you installed GEMPACK must be on your DOS path.
- if this GEMPACK directory is not C:\GP, the DOS environment variable GPDIR must be set equal to the GEMPACK directory.

You will need administrator access to make these changes. Follow steps 1 to 5 below to check that the installer made these changes, or to make them yourself:

1. For XP, right click on *My Computer* and select *Properties | Advanced | Environment Variables*. For Vista, select in turn: *Control panel | Classic view | System | Advanced system settings | Environment Variables*.

2. Select the system *Path* variable and add the GEMPACK directory to it (separated by a semicolon ;). You can add the GEMPACK directory to the beginning or end of the current PATH as you prefer. [If you have a previous GEMPACK directory on your Path, replace it with the new GEMPACK directory.] Click on the *Set* button to accept this change. Check that the directory containing the Fortran compiler is also on the PATH. If not, add this directory (for example, C:\LF9557\BIN for the LF90 compiler.)
3. Set a new *system* Environment variable named GPDIR and give it a value C:\GP10 (or whatever your GEMPACK directory is called.) Select the *Set* button to accept this change. Note that if you have installed GEMPACK in the default directory C:\GP, it is not necessary to set the Environment variable GPDIR. The GEMPACK programs will assume that C:\GP is the GEMPACK directory, which contains your GEMPACK licence and the GEMPACK libraries and programs, if GPDIR is not set.
4. Set a new *user* Environment variable named GPDIR, in the same way.
5. Click on the *Ok* button to accept these changes to the Environment.

These changes will take effect when you next start Windows programs or open a new DOS box.

Test these changes by opening a new DOS box and entering "SET". This should show the altered path and the new Environment variable GPDIR.

4.3 If you installed in a new directory (not C:\GP)

If you did NOT install GEMPACK in C:\GP, you may need to help some windows programs to find and use GEMPACK programs. We give the main examples below.

TABmate. You will need to tell TABmate where to find TABLO.EXE. To do this, go to the **Code and other** menu item under the **Options** menu. Then click on the **Change** button in the **Location of TABLO.EXE** part of the form and find the latest TABLO.EXE.

RunMONASH and RunDynam. Click on the **Options** menu and use menu items such as **Which ViewSOL to use** to tell the program which versions of ViewSOL, ViewHAR and AnalyseGE to use.

RunGTAP. Click on **Tools..Options** and then use the various **Change** buttons to tell RunGTAP where to find Gemsim, ViewHAR, ViewSOL, TABmate, AnalyseGE, Tablo etc.

4.4 GEMPACK licence

Your GEMPACK licence must be called LICEN.GEM and it must be placed in your GEMPACK directory (that is, the directory in which you installed GEMPACK). The installer may have already done this: if so, skip this section.

If you already have a Release 10 licence on your computer in another directory, please copy the file LICEN.GEM to your current GEMPACK directory.

Please see section 1.2 of GEMPACK document GPD-9 for details about GEMPACK licences.

CHAPTER 5

5 Testing the Installation

In this chapter we suggest that you test the main features of the installation by carrying out a simulation with the Stylized Johansen model.

If this simulation does not work, you will need to go back to some of the steps in sections 4.1 and 4.2 above.

5.1 *Checking the DOS Path and access to your GEMPACK licence*

Before carrying out the test simulations, you should check that your PATH has been set correctly and that GEMPACK programs are able to access your GEMPACK licence.

To do this, open a command prompt (DOS box) by going

Start..Run and type in "cmd" and hit OK.

This will start a DOS box running. In that DOS box, type in the commands

cd \temp (or some directory where you are able to write files)
tablo

If your PATH is set correctly, the GEMPACK program TABLO will start to run and it will find your GEMPACK licence. In this case you will be offered lots of options for the program TABLO.

If TABLO does not start running, your PATH is not as required. Please check again the steps in section 4.2 and then repeat this part of the testing.

If TABLO starts running, hit "Enter" (carriage-return).

If TABLO reports that it could not access your GEMPACK licence or that the licence is invalid, the error message will tell you where TABLO looked for the licence file. Please check that your Release 10 licence file (it must be called LICEN.GEM) is in your GEMPACK directory. If TABLO looks for LICEN.GEM in a directory which is different from the one in which you installed GEMPACK, check the parts of section 4.2 which relate to the Environment variable GPDIR. Repeat this testing once you have remedied any problems.

Stop TABLO running by typing **Control-C** (that is, hold down the Control key, which is usually on the left of your keyboard and may be labelled "Ctrl", and, while holding it down, touch the C key). TABLO should stop running (though you may need to type Control-C twice). Now type

exit

which should close the DOS box. You are now ready to carry out the test simulations, so please skip to the next section.

5.1.1 **If you still have problems**

If you still have problems, please run ViewHAR and select

Help / About ViewHAR/Diagnostics / Diagnostics

Save this information in a file, and email it to *support@gempack.com*.

5.2 Making a directory for the Stylized Johansen model

Start WinGEM running, following the procedure described in section 2.4.1 of GPD-1. Then prepare a directory for the model SJ (Stylized Johansen) and set the working directory, as described in sections 2.4.2 and 2.4.3 of GPD-1.

If you are an experienced user, you might like to follow the brief notes in section 2.4 of GPD-8 instead.

5.3 Simulations to test GEMPACK and WinGEM

To test that GEMPACK and WinGEM are working correctly, we suggest that you carry out the simulations with Stylized Johansen in section 2.4 of GPD-1. These examples carry out the simulation using a TABLO-generated program in section 2.4.6 of GPD-1. You might also like to carry out this same simulation using GEMSIM, as described in section 2.4.7 of GPD-1. In either case, check that the results of the simulation are as expected (see, for example, section 2.7 of GPD-1).

If you are a new user of GEMPACK, we recommend you work through the Stylized Johansen example in Chapter 2 of GPD-1 rather than using the brief notes on Stylized Johansen in section 2.4 of GPD-8.

If any of these tests does not work, re-check the steps in the installation in section 4.2 above.

CHAPTER 6

6 Working with GEMPACK on a PC

The following sections contain other information relevant to working with GEMPACK on your PC.

6.1 *Suggestions for hands-on computing*

The suggestions below will help you to become familiar with many important features of the use of GEMPACK on the PC. They are based on the models supplied with GEMPACK (see chapter 1 of GPD-8), especially the Stylized Johansen and Miniature ORANI models. The example model files are all in the EXAMPLES subdirectory of your GEMPACK directory (usually C:\GP\EXAMPLES).

As described below, you may follow the Windows way of working, or you may prefer to work at the DOS prompt.

6.1.1 Using WinGEM

See chapter 2 "Getting Started with GEMPACK via WinGEM" of GPD-8. This begins with examples based on the Stylized Johansen and goes on to examples based on Miniature ORANI, GTAP, ORANIG and ORANIF.

6.1.2 Not using WinGEM

See chapter 3 "Unix/Command Prompt: Hands-on Computing" of GPD-8. This begins with examples based on the Stylized Johansen and goes on to examples based on Miniature ORANI.

6.2 *New model's directory location*

We suggest that you put each new model you build in a separate directory on the hard disk, outside of the GEMPACK directory (usually C:\GP). Note that your PATH setting will ensure that the GEMPACK programs are found correctly. Conversely if your PATH and GPDIR are not set correctly, the GEMPACK programs will not run.

When you use WinGEM with any model, make sure that you set WinGEM's working directory to point to the directory containing the files for this model (as spelled out in section 2.4.3 of GPD-1).

6.3 *For speed, use TABLO-generated programs*

You have installed a source-code version of GEMPACK on your PC and have a suitable Fortran compiler, so you will probably prefer to use TABLO-generated programs instead of GEMSIM for simulations. For small models such as Stylized Johansen and Miniature ORANI there is not much difference in speed. But TABLO-generated programs will run faster with large models. [Some CPU times are reported in chapter 4 of GPD-8.]

6.4 *Text editor*

When installing and using GEMPACK, you will need to be able to edit text files. This is best done using a text editor (that is, an editor designed for handling text files exclusively). The text editor Notepad is supplied with Windows— but many others are available.

We recommend that you use the text editor **TABmate**. TABmate has syntax highlighting which is excellent if you are writing or debugging TABLO Input files. TABmate can also be used for other text

files, can open several files at the same time and has various Tools which are useful for GEMPACK development.

You can also use the older text editor GemEdit. If you use a word processor (such as Microsoft Word) to edit text files, be careful to save the resulting file as a text file.

6.5 If a program runs out of memory

GEMPACK programs may require more memory than is available on your computer. If so you will receive a message saying that the program is stopping because it is

unable to allocate sufficient memory.

If this happens free up more memory by closing down any other running applications; then try to rerun the task. Otherwise you need to find some other way of carrying out the task. You could also buy more memory, but 32-bit Windows cannot support more than 4 GB, and will only allocate 2 GB to a program. If that is insufficient you would need to consider the 64-bit Intel compiler, see section 3.1.2.

[If you are running GEMSIM or a TABLO-generated program, sometimes this error occurs if you have forgotten to condense your model.]

6.6 Fortran compiler options

The compiler options used in the standard installation usually compile relatively quickly and produce executable images which run as fast as possible. Except for the options which affect TABLO-generated programs (see section 6.6.2), we recommend that you do not change these compiler options without consulting us.

Some users may wish to change the optimization level when compiling TABLO-generated programs. That is why below we

- briefly introduce the idea of optimization levels, and
- discuss how and why you might want to change the default for TABLO-generated programs.

6.6.1 Optimization levels

There are several different options available with the different compilers supported by GEMPACK. For the GEMPACK user, the only ones you may change are the options relating to level of optimization. With the compilers supported by GEMPACK we recommend either no optimization or O1 level of optimization.

- Using no optimization results in quicker compilation and linking but possibly slower running.¹⁴
- O1 level of optimization performs certain standard optimizations of object code. Compilation will take longer than if no optimization is done but usually results in faster run-times. However, for TABLO-generated programs (which often have several thousand lines of code), this option can cause long compile/link times.¹⁵

See section 7.7 for more technical information about compiler options.

¹⁴ No optimization is denoted by “-O0” for the Lahey compilers LF90, LF95 and LF while it is denoted by “/Od” for the Intel compilers.

¹⁵ O1 level of optimization is denoted by “-O1” for the Lahey compilers LF90, LF95 and LF while it is denoted by “/O1” for the Intel compilers. See your compiler manuals for details.

6.6.2 Compiling TABLO-generated programs

For TABLO-generated programs, what level of optimization you might like depends on how often you will run the program after compilation.

- If you are going to run a TABLO-generated program many times it is worth waiting longer to compile it if the run time is reduced. For example, if you are preparing an executable image of a TABLO-generated program to run under RunDynam (or one of its variants), the program will be run many times (once for each year, for each of Base, Base Rerun and Policy).
- But if you are in development mode where you are constantly changing the TAB file and hence re-running LTG frequently, you don't want long compile times.

As described in more detail in section 7.7 below, the default is to compile TABLO-generated programs with no optimization. That is fine for the second case above but may not be what you want in the first case above (where you will run the executable image many times). If you wish to change the way in which TABLO-generated programs are compiled, and if you are using an Intel compiler or LF or LF95 (but not with LF90), proceed as follows.

- To change to O1 optimization (longer compile time, faster run time), run the BAT file LTGFIGO1.BAT in the main GEMPACK directory. This will set O1 options for compiling TABLO-generated programs. After you have run the LTGFIGO1.BAT file, you will need to recompile and link your model (for example, using WinGEM or using LTG at the DOS prompt). [However, you may find compile times are unacceptably long. If so, see the next point.]
- To change back to the no optimization option (shorter compile time, longer run time), run the BAT file LTGFIGO0.BAT.

Optimization O1 will not help much if LU decomposition (see section 12.1 of GPD-3) takes most of the run time. Using O1 can only reduce the time for the Formulas, Submatrices, Backsolves and Updates (see section 6.2 of GPD-3). [The LU decomposition routines are not compiled when you compile a TABLO-generated program – they are compiled with O1 (see section 7.7) when you build the GEMPACK libraries.]

6.7 File association

"File association" is the Windows mechanism due to which (for example):

- the ViewHAR icon is displayed beside HAR files in Explorer
- if you double-click a HAR file, it opens in ViewHAR

These happen because files suffixed HAR are "associated" with ViewHAR.exe. Usually the "association" is set up at install time. Only one program can be associated with each file type – so programs might compete to possess more popular suffixes. To stop such contests, Vista prevents any program from changing an existing association. This may mean, for example, that the GEMPACK installer cannot associate TABmate with TAB files (because Microsoft wants the TAB suffix for Visual Studio). In such cases, you must set up the association manually. In both XP and Vista you can change the HAR file association as follows:

- in Windows Explorer, right-click on a HAR file and choose **Open with...**
- check box **Always use the selected program** should be ticked.
- Use the Browse button to select the latest ViewHAR.exe.

6.8 Keep and Temporary directories and GEMPACK Windows programs

Older versions of the GEMPACK Windows programs used to write INI files and temporary files in subdirectories of the Windows directory. This caused problems in Windows XP/Vista where a "standard" user may not be allowed to write or delete files in the Windows directory or any subdirectory of it. Now, however:

- The position of the Keep Directory where the INI files are stored has been changed to a subdirectory of the user's "My Documents" directory.
- The default place where GEMPACK Windows Programs write temporary files has been moved from under the Windows directory to a subdirectory of the (user's) TMP or TEMP Environment variable.¹⁶ You can reset the TMP variable as described in section 2.2.1.
- In RunGEM, WinGEM, AnalyseGE and RunMONASH there are Menu Options within the programs which allow you to set your Temporary directory to a directory of your choosing. The program remembers this Temporary directory setting. When you start the programs for the first time, the default temporary directory is set from the value of the TMP or TEMP environment variable.
- For very unusual cases, there are ways to avoid problems with the Keep and the default Temporary directories by setting environment variables called **GPKEEP** and **GPTEMP**. Set a new environment variable called GPKEEP if you want to change the usual Keep Directory. Set a new environment variable called GPTEMP if you want to change the default temporary directory without changing the environment variable TMP.

If you are having problems with these features of one of the GEMPACK Windows programs, consult the relevant Help file for details and advice.

6.9 Copying GEMPACK programs to other PCs

If you have a GEMPACK site licence, you are entitled to copy GEMPACK programs to PCs covered by your site licence, for example, within the same department or institute.

You should note that the GEMPACK programs TABLO and GEMSIM require a GEMPACK licence. Accordingly, under the terms of your GEMPACK licence, you must not copy (or send copies of) executable images of these to machines outside the site which is covered by your GEMPACK licence.

You are allowed to send copies of the other GEMPACK programs, including TABLO-generated programs, outside of the site covered by your GEMPACK licence. However, you should note that

- TABLO-generated programs may require an Introductory licence if they are used with a large model - see section 1.9.5 of GPD-1.
- SAGEM.EXE requires at least an Introductory GEMPACK licence to solve larger than medium-sized models (see section 6.3.3 of GPD-7).

Of course you must not send a copy of your GEMPACK licence outside the site covered by your licence.

6.9.1 Windows programs

Anyone (with or without a GEMPACK licence) is entitled to download and install the GEMPACK Windows programs from the GEMPACK web site (see section 1.3 of GPD-1). However a

¹⁶ For example if your Environment variable TMP is set to C:\TEMP, the default temporary directory for WINGEM would be C:\TEMP\GPTEMP\WINGEM.

GEMPACK licence may be required to work with larger than medium-sized models. See the Help files for the various programs for details.

The versions of these programs may be updated between GEMPACK Releases. You can check

<http://www.monash.edu.au/policy/gpwingem.htm>

on the GEMPACK web site to see if updates are available.

6.10 Accessing the CoPS web pages from the CD

The GEMPACK CD may contain a recent snapshot of the Centre of Policy Studies (CoPS) Web site at Monash University. The snapshot is created as a convenience for those who do not have easy or rapid Internet access. Please be aware that the real Web site may well have been updated since this CD was created. The real web address is *<http://www.monash.edu.au/policy>*

To access the CD snapshot of the web site, you do **not** have to be connected to the Internet. However, you do need a Web Browser, such as Firefox or Internet Explorer. Enter the location:

e:/welcome.htm where "e" is the drive letter of your CD drive.

Included in the CoPS Web pages are digital versions of many CoPS/Impact Project Working papers. The **Resources** web page links to the CoPS Archive, containing numerous examples of GEMPACK applications.

Links pointing to sites outside the Centre of Policy Studies, and *mailto* links have been preserved, but will only function if you are connected to the Internet.

6.11 Using GEMPACK in a teaching situation

You may be interested in using GEMPACK for teaching GE modelling in a computer laboratory.

If so, the Source-Code Version is not ideal since it requires a Fortran compiler which may be costly to license for laboratory (multi-user) use.

We use the Limited Executable-Image Version of GEMPACK in our courses (for example, the Practical GE Modelling course – see <http://www.monash.edu.au/policy/pgemc.htm>). This does not require a Fortran compiler. We have found it ideal for teaching.

If you have a multi-user Source-Code GEMPACK licence for your university (etc), we will be happy to supply you with a Limited Executable-Image CD which you can use in your teaching.

CHAPTER 7

7 Technical Topics

In this chapter we discuss various technical topics. We expect that most GEMPACK users can happily ignore this chapter.

7.1 Working in a DOS box

Many GEMPACK users prefer to work in a Windows way via WinGEM (or perhaps via RunGEM and TABmate – see section 5.4 of GPD-8). If so, **you can skip this section** and go to section 6.5.

If you prefer to work at the DOS prompt, you will find useful information in this section.

7.1.1 Compiling and linking TABLO-generated programs

Wherever your TABLO-generated program is located on the disk, you can compile and link it using the command LTG. For example, to compile and link the program SJ.FOR, the command is

LTG SJ

[Don't add the suffix .FOR.]

Occasionally, you may need to increase the stack size – see section 7.8.1 below.

7.1.2 TABLO-generated programs

When you run TABLO to produce a TABLO-generated program, you will notice that TABLO produces several extra files with suffix .FOR as well as the usual files with suffixes .FOR, .AXS and .AXT.

For example, when you run TABLO on SJ.TAB, TABLO produces

- SJ.FOR [the TABLO-generated program]
- SJ.AXS and SJ.AXT [the Auxiliary files – see section 3.2 of GPD-3]
- SJ.MIN (the Model Information file – see section 3.4.1 of GPD-3), and
- various Fortran **module files**, namely
SJ_C0.FOR, SJ_C1.FOR and SJ_C3.FOR to SJ_C7.FOR [not SJ_C2.FOR]
SJ_U0.FOR to SJ_U4.FOR,
SJ_V0.FOR, SJ_V1.FOR and SJ_V3.FOR to SJ_V6.FOR [not SJ_V2.FOR],
and SJ_M0.FOR.

These module files are required for compiling and linking the TABLO-generated program (similarly to the .FOR file).

For very large models, there may be more module files than those listed above. Large modules are split into several .FOR files, to help the compiler. [Some compilers (including LF95 and LF) do not handle very large files well.]

No Ux.FOR or Vx.FOR module files are produced if the TAB file for the model has no update statements or levels variables. For example, there are no SJCHK_U*.FOR or SJCHK_V*.FOR module files associated with the TABLO-generated program for the data-manipulation TAB file SJCHK.TAB.

All the .FOR files are needed, until you have completed the next, LTG, stage to produce the model EXE file. *Then you can delete any .FOR files produced by TABLO* (usually LTG deletes most or all of them).

7.1.3 Running GEMPACK programs in a DOS box

Under DOS, this is done just by typing in the name of the program, for example,

sagem

Because the GEMPACK directory is on your DOS PATH, your computer should find SAGEM.EXE in your GEMPACK directory and start it running.

For TABLO-generated programs, change to the directory where the TABLO-generated program is. Type in the name of the program, for example,

sj

to run the program SJ.EXE.

New command line options for some programs were introduced in Release 9.0 – see section 10.1 of GPD-5 for details. If you like working at the DOS prompt, or if you use DOS batch files (see section 7.2), you will probably make considerable use of these command line options.

Example

The command

tablo sj -pgs -log sj.log

provides complete user input for running TABLO to produce output for GEMSIM (-pgs) and the log file SJ.LOG.

We strongly suggest that you always work in the DOS box that WinGEM provides via the Menu option:

File / Shell to DOS

The settings of this DOS box have been designed to produce good productivity, especially for GEMPACK users who are running long simulations in a DOS box and simultaneously working with other Windows programs (for example, word processing or spreadsheet programs). In particular, programs running "in the background" in these DOS boxes usually receive a reasonable fraction of total CPU time without disrupting the foreground task.

7.1.4 Interrupting programs

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

7.1.5 Controlling screen output

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

Control-S Control-Q

keystrokes. (For Control-S, hold down the Control key, which is usually on the left of your keyboard and may be labelled "Ctrl", and, while holding it down, type S).

Use Control-S to stop the screen output and Control-Q to start it again. You can repeat these as needed. However, if you get out of step, say by typing two Control-S in a row, you will lose control

of the output and have to wait until the program ends; even Control-C (see section 7.1.4 above) will probably fail then.

On some machines the Scroll Lock key works in a similar way. (It first stops screen output, then starts it, then stops it, and so on.)

7.1.6 Using stored-input files in a DOS box

You can use Stored-input files under DOS either via the GEMPACK **sti** option, or using the command line **-sti** feature (see section 5.5 of GPD-1), as in, for example,

modhar -sti modsj.sti

You can create Stored-input files using option **sif** – see section 4.3 of GPD-1 for introductory examples.

If you make your own Stored-input files using a text editor, it is a good idea to include the line

bat

at the start of these files. This means that, if the program encounters invalid input, it will stop. (See section 5.3 of GPD-1.)

We do not recommend using input redirection as in "modhar < modsj.sti". We have found that the **-sti** feature is more robust.

7.1.7 Use a log file to see screen output

If the screen output is too fast for you to read, you can always save a GEMPACK log file. This is just a text file that contains all the output to the screen. See section 5.5 of GPD-1. For example,

modhar -sti modsj.sti -log modsj.log

saves a log file called MODSJ.LOG which you can look at in your text editor. Again, we recommend using the **-log** feature instead of output redirection via ">".

7.2 DOS batch files

If you create DOS batch (.BAT) files for carrying out tasks including running GEMPACK programs, you may like to take advantage of the fact that, if any GEMPACK program ends with a fatal error, it sets the value of the DOS parameter ERRORLEVEL value to 1. You can test for this in .BAT files to stop the batch job early in such a case.

For example, the .BAT file below runs SAGEM and then GEMPIE. If the line 3 SAGEM run ends unsuccessfully, the line 6 ERRORLEVEL test aborts the batch job and gives a message saying that the job was unsuccessful.

```

REM Beginning of batch file
REM Run SAGEM
REM Next uses the -sti option (see section 5.5 of GPD-1)
sagem -sti sag1.sti -log sag1.log
REM test ERRORLEVEL to see if this was successful
if errorlevel 1 goto error
REM Run GEMPIE
gempie -sti gemp1.sti -log gemp1.sti
if errorlevel 1 goto error
echo off
echo BATCH JOB SUCCESSFUL
goto endbat
:error
echo off
echo *** ERROR: BATCH JOB FAILED ***
:endbat
REM End of batch file

```

All GEMPACK programs have options **dro** and **dre** which can be used in BAT files to check whether or not a program has run successfully. See section 5.4.1 of GPD-1 for documentation about these options and for an example BAT file showing how they can be used. These provide an alternative to the use of ERRORLEVEL tests, the effects of which can be difficult to predict and can vary slightly between the different versions of Windows.

We strongly recommend that you use "-sti" (as above – see also section 7.1.6) to run GEMPACK programs in a DOS batch file (rather than using input redirection). We have found "-sti" more robust than input redirection. The same applies to "-log" (rather than output redirection) if you want to generate a LOG file – see also section 7.1.7.

New command line options for some programs were introduced in Release 9.0 – see section 10.1 of GPD-5 for details. You may be able to use these command line options to reduce the number of Stored-input files you need in your batch files.

7.2.1 Trapping for LTG errors in DOS batch files

If LTG.BAT (which is used to compile and link a TABLO-generated program) encounters an error, it sets ERRORLEVEL to 1.¹⁷

If you are calling LTG in your own DOS batch file, and if you run it via “call ltg ..”, you can now catch errors by testing ERRORLEVEL. See section 7.2 above for more details.

Example.

```

call ltg sj
if errorlevel 1 goto error
sj -cmf sjlb.cmf
if errorlevel 1 goto error
goto endbat
:error
echo ** ERROR: BAT JOB FAILED
:endbat

```

¹⁷ This did not happen with Release 8.

7.2.2 Making executable images of GEMPACK programs in a DOS box

In general, executable images of the GEMPACK programs are made as part of the installation of GEMPACK Source code. However you may wish to remake just one of these programs in a DOS box.

In a DOS box change to the directory where you installed the GEMPACK programs (usually C:\GP).

```
c:  
cd \gp
```

Then to make a main program for example ACCUM, enter the command

```
mkmain accum
```

This command *mkmain* works with all the main programs except GEMSIM and TABLO, where the appropriate commands are

```
mkgemsim
```

```
mktablo
```

[To make executable images of TABLO-generated programs, see section 7.1.1.]

7.3 Running BuildGP

The program **BuildGP** is designed to carry out the following tasks:

1. Check your system to see: if Fortran is installed, if there is enough disk space, whether the licence is in the correct place, and if the PATH and GPDIR environment variables are correctly set.
2. Make the GEMPACK libraries by compiling many groups of subroutines.
3. Make the Fortran-based GEMPACK programs (executable images) by compiling the programs and linking to the libraries.

Usually BuildGP does all these automatically when you install GEMPACK. However, there may be situations when you need to run BuildGP yourself. For example, if you discovered a GEMPACK bug, you might be sent a patch file to repair the problem. Detailed instructions would come with the patch file. We provide only brief notes here.

Run the program BUILDGP.EXE in the GEMPACK directory from *My Computer* or *Windows Explorer*.

Check that BuildGP correctly displays your GEMPACK directory and compiler (LF90 or LF95 or Intel).¹⁸

Now click on the *Start build* button.

7.4 FLIB_DVT_BUFFER environment variable

EXE files produced by the Lahey LF or LF95 compiler tend to use memory wastefully. Probably you would only notice this when running a very large model. The problem is fixed by creating and setting the Environment variable FLIB_DVT_BUFFER equal to 0 (ie, zero). The EXE file does not need to be recompiled.

¹⁸ If you installed GEMPACK successfully with one compiler (say, LF95), and you later wished to use another compiler (say, Intel), you need to *install from the CD again*. It is not enough to merely re-run BuildGP. The installer copies compiler-specific files from the CD.

Install packages for GEMPACK Release 10 automatically set FLIB_DVT_BUFFER=0, *even if you chose the Intel compiler*. The reason is, you may wish to run an LF-produced EXE from another source. For example, you may wish to run the GTAP.EXE that came with your RunGTAP¹⁹ package. If that GTAP.EXE was made by LF or LF95, its performance on your PC might be improved by the FLIB_DVT_BUFFER setting.

You might, for some reason, need to manually set the Environment variable FLIB_DVT_BUFFER equal to 0 (ie, zero). You could copy the procedure to set Environment variable TMP, described in section 2.2.1.

7.5 Variants of LTG

In your GEMPACK directory, there are various BAT files related to the main LTG.BAT file and used in various situations to compile and link Fortran programs to the GEMPACK libraries.

7.5.1 Debug LTG: LTGDEB.BAT (LF95 and LF only)

When compiling and linking a TABLO-generated program with Fortran compilers LF95 or LF, you may occasionally get the error message

"8694-U The number of procedure or data reference exceed compiler limit" .

This problem can be sometimes be solved by using LTGDEB.BAT to cut the main .FOR file into smaller pieces²⁰. To use LTGDEB.BAT, in your GEMPACK directory, type the commands

```
copy LTG.BAT LTGWAS.BAT
copy LTGDEB.BAT LTG.BAT
```

Then try LTG (Compile and link) again with your TABLO-generated program.

7.5.2 Keep/Delete Fortran LTG: LTGKPFOR.BAT and LTGNOFOR.BAT (all compilers)

After compiling and linking a TABLO-generated program, the BAT file LTGNOFOR.BAT deletes the relevant Fortran .FOR files (the one for the TABLO-generated program and the .FOR files containing its modules).

The BAT file LTGKPFOR.BAT does not delete these .FOR files.

LTG.BAT is usually the same as LTGNOFOR.BAT which tries to clean up .FOR and .OBJ and .MOD files once the EXE file for the TABLO-generated program has been created. LTGKPFOR.BAT is mainly used for debugging TABLO-generated programs.

7.5.3 No Modules LTG: LTGNOMOD.BAT and LTGFNMOD.BAT (all compilers)

A few GEMPACK Source-code users may be interested in writing their own Fortran programs using GEMPACK subroutines. If you are so interested, please see Harrison *et al* (2005).

If you are writing your Fortran program in fixed format, you can compile and link to the GEMPACK libraries using LTGNOMOD.BAT instead of LTG.BAT. The "NOMOD" refers to the fact that there are no Fortran module files for your program unlike TABLO-generated programs which have several module files – see section 7.1.2.

If you are writing your Fortran program in free format, you can compile and link to the GEMPACK libraries using LTGFNMOD.BAT.

¹⁹ If you have never heard of GTAP, visit www.gtap.org to find out more!

²⁰ The LF95 "-split" option does not work.

The compiler options used in LTGNOMOD.BAT are the same as those in LTG.BAT – see section 6.7.

7.6 Uninstalling GEMPACK

To uninstall GEMPACK from your computer:

1. Delete all the files in the GEMPACK directory and its subdirectories. Then remove this directory.
2. Remove the changes to your PATH and environment made in section 4.2.1.

7.7 Compiler options – Technical information

We introduced this topic in section 6.6 above. Here we give some more technical details.

7.7.1 Compiler options for the Intel compilers

We have included O1 optimization in the standard configuration file (INTEL_1.FIG) in the SUBS, MODULES and TABLO subdirectories but no optimization in the GEMPACK directory. This default setting is produced by running the BAT file **GPFIG.BAT**.²¹

If you change the optimization level (by changing the INTEL_1.FIG file in one or more subdirectories), you need to build GEMPACK libraries and programs again to see the effects. You also need to recompile and link your TABLO-generated programs.

The Intel compilers offer higher levels of optimization than O1. We strongly recommend against using them since we have found some bugs with /O2 with the 64-bit Intel compiler. Anyhow likely speed gains from /O2 are small.

TABLO-generated programs. The configuration file **INTEL_1_LTG.FIG** (in your GEMPACK directory) is used when compiling and linking TABLO-generated programs. This controls the level of optimization. The default is no optimization since we have found that O1 optimization increases the compile time by too much for the corresponding reduction in run time. See section 6.6.2 for more details, including how and why you might want to change the default level of optimization.

7.7.2 Compiler options for Lahey/Fujitsu LF95 or LF Version 7 compiler

For Release 10 (as for Release 9) we have included O1 optimization in the standard configuration file (LF95.FIG) in the SUBS, MODULES and TABLO subdirectories but no optimization in the GEMPACK directory. This default setting is produced by running the BAT file GPFIG.BAT.²²

If you change the optimization level (by changing the LF95.FIG file in one or more subdirectories), you need to build GEMPACK libraries and programs again to see the effects. You also need to recompile and link your TABLO-generated programs.

TABLO-generated programs. The configuration file **LF95LTG.FIG** (in your GEMPACK directory) is used when compiling and linking TABLO-generated programs. This controls the level of

²¹ The file INTEL_1.FIG controls the level of optimization (and other compiler options) for the Intel compilers. We provide two main variants, namely INTEL_1_O1.FIG (which includes /O1) and INTEL_1_NOOPT.FIG (which includes /O0). The BAT file NOOPTFIG.BAT sets /O0 in the GEMPACK directory and all subdirectories. The file INTEL_2.FIG is used when linking programs.

²² The file LF95.FIG controls the level of optimization (and other compiler options) for LF95 and LF. We provide two main variants, namely LF95O1.FIG (which includes -o1) and LF95NOOPT.FIG (which includes -o0). The BAT file NOOPTFIG.BAT sets -o0 in the GEMPACK directory and all subdirectories.

optimization. The default is no optimization. See section 6.6.2 for more details, including how and why you might want to change the default level of optimization.

7.7.3 Compiler options for Lahey compiler LF90

For Release 10 (as for Release 9) we have included O1 optimization in the standard configuration file (LF90.FIG) in the SUBS and MODULES subdirectories but no optimization in the TABLO subdirectory for LF90 because the compile and link of GEMSIM.EXE and TABLO.EXE is too slow with O1. This default setting is produced by running the BAT file GPFIG.BAT.²³

If you change the optimization level (by changing the LF90.FIG file in one or more subdirectories), you need to build GEMPACK libraries and programs again to see the effects. You also need to recompile and link your TABLO-generated programs.

TABLO-generated programs. With LF90, the configuration file LF90.FIG in the main GEMPACK directory is used when compiling and linking TABLO-generated programs. This controls the level of optimization. The default is no optimization because O1 causes unacceptably long compile times.²⁴

7.7.4 Changing the optimization setting (all compilers)

Two BAT files in the GEMPACK directory (usually C:\GP) are used to swap between optimization settings. The file NOOPTFIG.BAT sets no optimization option in all relevant directories and the file O1FIG.BAT (not available with LF90) sets the optimization at O1 in all relevant directories and subdirectories.

To use these BAT files, go to the DOS prompt and change directory to your GEMPACK directory. Then type in the name of the BAT file. After you have run either of the BAT files NOOPTFIG.BAT or O1FIG.BAT, you need to build the GEMPACK libraries and programs again. You also need to recompile and link your TABLO-generated program using LTG.

7.8 Advanced compiler use

7.8.1 Changing the stack size of TABLO-generated programs

This section applies only to the Lahey compilers LF, LF95 and LF90.

You may get an error message saying that a TABLO-generated program has run out of program stack. In this case you must increase the size of the program stack. The file LTG.BAT which is used to compile and link TABLO-generated programs (see section 7.1.1) includes a default stack size following "-stack". (To see the current default value, look in this file in directory C:\GP.) You can increase the stack size for TABLO-generated programs by passing you desired stack size to LTG as a second argument.²⁵ For example

²³ The file LF90.FIG controls the level of optimization (and other compiler options) for LF90. We provide two main variants, namely LF90O1.FIG (which includes -o1) and LF90NOOPT.FIG (which includes -o0). The BAT file NOOPTFIG.BAT sets -o0 in the GEMPACK directory and all subdirectories.

We have found one example in which -o1 with LF90 produced bad code. We have worked around this by changing our code slightly, so this example is no longer a problem. Asking compilers to do optimization always introduces a slight risk that bad code will be produced (which is mainly why we did not recommend -o1 by default prior to Release 9). We now think that the speed up gained from using -o1 outweighs this risk. However, if you think that the compiler may be producing bad code, please contact us.

²⁴ For this reason, we do not recommend that you put option -o1 in the LF90.FIG file in your GEMPACK directory. [If you do, you will be using -o1 when compiling TABLO-generated programs.]

²⁵ The LTG2 command used with Release 5.1 to increase stack size is now simply LTG.

ltg model 800000

produces an executable image of the TABLO-generated program MODEL.FOR and sets the stack size to 800,000 bytes. If, when you use LTG to increase the stack size, your current size isn't large enough, try increasing it again.

You can get a good indication as to how large to make the stack size by looking at the compilation phase. When the main program is compiled (this happens at the start of the LTG command), you will probably see a warning message about the minimum size of the stack in cases where the stack size given by LTG is not large enough.

7.8.2 Maximum amount of memory available to programs

Windows will only allow each 32-bit program to access at most 2 Gigabytes of memory – even if you have more memory than this on your computer. 64-bit programs (which only work under 64-bit Windows) have no such limit²⁶.

²⁶ Although even 64-bit Tablo-generated GEMPACK programs can allocate at most 2 billion elements in the sparse linear system solved in the LU phase. The 2 billion entries would use 24GB of RAM – so few users will hit this limit.

CHAPTER 8

8 Installing GEMPACK on a network

Some organisations have found it desirable to run the Windows PC Source-code Version of GEMPACK and the associated Fortran compiler from a network. For example, this can reduce the need for separate copies of the Lahey compiler.

Below are some pointers to using GEMPACK and/or Fortran on a network.

- If you have the Lahey Fortran LF90 or LF95 on a network, please note the following.

When compiling (eg via LTG), the Fortran compiler needs to know where the relevant Fortran library is to be found. You can avoid having to edit the various .BAT files by having all users set the value of the DOS Environment variable **GPFLIB** to point to this directory. You can set this environment variable using the method described in section 2.2.1. The person installing GEMPACK (building the libraries and executable images) should set Environment variable GPFLIB (and reboot, if necessary) before running BuildGP (see section 7.1).

- Another issue (relevant to all compilers) is who has access to the GEMPACK source etc on the network. Everyone needs read and execute permissions but only the administrator may need write access. Since Release 9, GEMPACK Windows programs do not write to any subdirectories of the Windows directory.

CHAPTER 9

9 Older versions of Windows

As stated earlier, GEMPACK is no longer tested or supported on Windows versions prior to XP. However you may well succeed using Windows 98, NT or 2000. For such cases, we have retained the following installation notes from previous editions of this manual.

9.1 *Setting the TMP directory*

- **If you are running Windows 2000**

You can right-click on "My Computer" and select "Properties | Advanced | Environment Variables" and then edit the User Environment variable called TMP to be C:\TEMP.

Create a new directory C:\TEMP if you do not already have one.

- **If you are running Windows NT**

You can right-click on "My Computer" and select "Properties | Environment Variables" and then edit the User Environment variable called TMP to be C:\TEMP.

Create a new directory C:\TEMP if you do not already have one.

- **If you are running Windows 95, 98 or ME**

You need to change your AUTOEXEC.BAT file. For example, put the line

```
set TMP=c:\temp
```

to set TMP to point to directory c:\temp. [Make sure that the only space in this line is the one between "set" and "TMP".] You need to reboot after making a change to AUTOEXEC.BAT.

If you are using Windows 95 or 98 or ME and you made changes to your AUTOEXEC.BAT file, you just need to **restart your computer** now. This will complete the main part of the installation of GEMPACK. Once your computer restarts, you can go on to go to section 4.4 below.

9.1.1 **PATH and GPDIR changes in AUTOEXEC.BAT (Windows 95, 98 or ME)**

If you are using Windows 95, 98 or ME, the usual method is to use AUTOEXEC.BAT. The installation program may have already made the changes to AUTOEXEC.BAT (with your permission – see section 4.1 above). In that case you must **restart your computer** for the changes to take effect.

If you prefer to make changes to AUTOEXEC.BAT yourself, AUTOEXEC.BAT is a text file in the directory C:\ of your computer. You must edit it in a text editor.

The important changes to AUTOEXEC.BAT are that

- the directory into which you installed GEMPACK must be on your DOS path.
- if this directory is not C:\GP, the DOS environment variable GPDIR must be set equal to this directory.

Once you have made these changes, you must **restart your computer** before proceeding. Examples of these changes are given in section 9.1.1.1 below.

9.1.1.1 Example of Changing PATH and GPDIR in AUTOEXEC.BAT

You have installed GEMPACK in a directory on your hard disk. The default directory name for the GEMPACK directory is C:\GP. However you may have chosen some other directory name in section 4.1 above. For this example we will assume you have used C:\GP90.

If you have not allowed the installer to make changes to your AUTOEXEC.BAT, you must edit the appropriate file (called AUTOEXEC.BAT in your default directory C:\) which is executed when you turn on your PC. (If you have no such file, create one.) You should add these directories to the PATH line in that file. (Use a text editor, such as the editor EDIT which comes with DOS, or GemEdit or TABmate.)

For example, if you find a line

```
PATH = C:\;C:\WINDOWS
```

you should change it to

```
PATH=C:\;C:\WINDOWS;C:\GP90;C:\LF9045\BIN          (if using LF90 Version 4.5)
PATH=C:\;C:\WINDOWS;C:\GP90;C:\LF9557\BIN          (if using LF95 Version 5.7)
```

Note that it is important not to include any spaces in these lines.

If you do not find a PATH line, make a new line

```
PATH=C:\GP90;C:\LF9045\BIN          (if using LF90 Version 4.5)
PATH=C:\GP90;C:\LF9557\BIN          (if using LF95 Version 5.7)
```

(If you installed GEMPACK on a disk drive different from C:, specify that one in the PATH line.) Do not leave any spaces in the PATH line.

If you installed the GEMPACK files in directory C:\GP, no other change is required to AUTOEXEC.BAT. But if you installed these files in another directory, you need to add an extra line to AUTOEXEC.BAT. For example, if you installed the GEMPACK files in directory C:\GP90, add the following line:

```
SET GPDIR=C:\GP90
```

(Change this appropriately to indicate where you actually installed these files.) Note that it is important not to include any spaces in this line, apart from the one between SET and GPDIR.

9.1.1.2 Uninstalling

For Windows 95, 98 and ME, remove the lines in your AUTOEXEC.BAT which add the GEMPACK directory to your path, and remove the SET GPDIR=... statement from AUTOEXEC.BAT.

10 REFERENCES

Harrison, W.J., J.M. Horridge and K.R. Pearson (2005), 'Using GEMPACK Subroutines in Your Fortran Programs'; presented at the GTAP conference in Lubeck, Germany in June 2005. This is available at <http://www.monash.edu.au/policy/gpusesub.htm>

11 GEMPACK DOCUMENTS

Harrison, W.J. and K.R. Pearson (2002), *An Introduction to GEMPACK*, GEMPACK Document No. 1 [**GPD-1**], Monash University, Clayton, Sixth edition, October 2002, pp.207+9.

Harrison, W.J. and K.R. Pearson (2002), *TABLO Reference*, GEMPACK Document No. 2 [**GPD-2**] Monash University, Clayton, Fourth edition, October 2002, pp.191+10.

Harrison, W.J. and K.R. Pearson (2002), *Simulation Reference: GEMSIM, TABLO-generated Programs and SAGEM*, GEMPACK Document No. 3 [**GPD-3**], Monash University, Clayton, Second edition, October 2002, pp.262+12.

Harrison, W.J. and K.R. Pearson (2002), *Useful GEMPACK Programs*, GEMPACK Document No. 4 [**GPD-4**] Monash University, Clayton, Second edition, October 2002, pp.138+10.

Harrison, W.J. and K.R. Pearson (2005), *Release 9.0 of GEMPACK: New Features and Changes from Release 8.0*, GEMPACK Document No. 5 [**GPD-5**] Monash University, Clayton, First edition, April 2005.

Horridge, J.M., M. Jerie and K.R. Pearson (2008), *Installing and Using the Source-Code Version of GEMPACK on Windows PCs with Lahey or Intel Fortran*, GEMPACK Document No. 6 [**GPD-6**], Monash University, Clayton, Thirteenth edition, May 2008.

Horridge, J.M., M. Jerie and K.R. Pearson (2008), *Installing and Using the Executable-Image Version of GEMPACK on Windows PCs*, GEMPACK Document No. 7 [**GPD-7**], Monash University, Clayton, Tenth edition, May 2008.

Harrison, W.J. and K.R. Pearson (2002), *Getting Started with GEMPACK: Hands-on Examples*, GEMPACK Document No. 8 [**GPD-8**], Monash University, Clayton, Third edition, October 2002, pp.110+8.

Horridge, J.M., M. Jerie and K.R. Pearson (2008), *Release 10.0 of GEMPACK: New Features and Changes from Release 9.0*, GEMPACK Document No. 9 [**GPD-9**], Monash University, Clayton, First edition, May 2008.

12 INDEX

A

- Address
 - Contact, 4
- Administrator
 - Privileges, 4
- Allocating memory, 22
- AnalyseGE, 15
- Asian characters, 3, 17
- AUTOEXEC.BAT, 39, 40

B

- Batch files, 29
- BuildGP
 - Error, 17
 - Overview, 31
- Buying a Fortran compiler
 - Intel recommended, 7

C

- Centre of Policy Studies, 4
- Characters
 - Non-English, 3, 17
- CMD.EXE
 - Required, 4
- Compiler options
 - For TABLO-generated programs, 23
 - Introduction, 22
 - Technical details, 33
- Compilers supported
 - On Windows PCs, 7
- Contact
 - GEMPACK, 4
- Control-C, 28
- Control-Q, 28
- Control-S, 28
- Courses
 - Exe-Image may be best, 25

D

- Diagnostic information, 19
- DOS batch files, 29
 - Errorlevel from LTG, 30
- DOS box, 27
- DOS path, 40

E

- Environment variables
 - GPDIR, 17
 - GPFLIB, 37
 - GPTEMP and GPKEEP, 24
 - Setting, 3
 - TMP, 3, 24
- Error
 - BuildGP, 17
- Errorlevel
 - DOS batch files, 30

- Executable image
 - Of a TABLO-generated program, 27, 34

F

- F77L-EM/32, 7
- Fortran compiler
 - Intel recommended, 7
 - Purchasing, 7
- Fortran library, 37

G

- GemEdit
 - Text editor, 22
- GEMPACK
 - Installing, 15
- GEMPACK directory, 15
- GEMPACK licence, 16, 18, 24
 - Introductory, 24
- GEMPACK Source-Code CD. *See* Source-Code CD
- GPDIR, 40
- GPFIG.BAT
 - Sets default optimization, 33, 34
- GPFLIB, 37
- GPKEEP, 24
- GPTEMP, 24

I

- IF32, 8
- IF64, 8
- IFORT, 3
- Impact Project, 4
- Input redirection via "<"
 - Not recommended, 29
- Installing Source-code GEMPACK, 15
- Intel compilers
 - Compiler options, 33
 - Do not use /O2 optimization, 33
- Intel Fortran compiler
 - Installing, 9
- Intel Fortran compilers
 - Versions supported, 8
- INTEL_1.FIG, 33
- INTEL_1_LTG.FIG
 - For TG-programs with Intel compiler, 33
- INTEL_2.FIG, 33
- Interrupting programs, 28
- Introductory GEMPACK licence, 24
- IT support
 - Notes for, 4

K

- Keep directory, 24

L

- Lahey Fortran compilers
 - Installing, 9
 - Maintenance updates, 7

- Versions supported, 7
- LF, 3, 7
- LF90, 3, 7
 - Compiler options, 34
- LF90.FIG, 34
- LF95, 3, 7
 - Compiler options, 33
- LF95.FIG, 33
- LF95LTG.FIG
 - For TG-programs with LF95 compiler, 34
- LICEN.GEM, 16, 18, 19
- Licence file, 16, 18, 24
- LOG files, 29
- LTG, 27, 34
 - Sets errorlevel, 30
 - Variants, 32
- LTG.BAT, 34
 - Variants, 32
- LTG2, 34
- LTGFIG00.BAT, 23
- LTGFIG01.BAT, 23
- LU decomposition
 - Time taken, 23

M

- Machine requirements, 3
- Memory, 3
 - Allocating, 22
 - Insufficient, 22
- Module .FOR files, 27

N

- Network, 37
- Non-English characters, 3, 17

O

- Optimization, 22
- Options
 - GEMPACK - DRE, 30
 - GEMPACK - DRO, 30
- Output redirection via ">"
 - Not recommended, 29

P

- PATH, 40
 - Privileges, 4

R

- RunDynam
 - TG-programs running under, 23
- RunGEM, 15, 27

S

- SAGEM
 - May require GEMPACK licence, 24
- Screen output, 28
- Setting Environment variables, 3
 - TMP, 3
- Source-Code CD, 15
- Space character, 3
- Stack size, 34
- Stored-input files, 29
 - Creating via "sif" option, 29
 - Using, 29
 - Via option "sif", 29

T

- TABLO-generated programs, 27, 34
 - Compiling, 23
 - Module .FOR files, 27
- TABmate, 15, 27
 - Text editor, 22
- Teaching courses
 - Exe-Image may be best, 25
- Temporary directory, 24
- Temporary files, 24
- Testing
 - Diagnostic information, 19
 - Stylized Johansen, 19
- Text editor
 - Choosing, 22
- TMP
 - Environment variable, 24

V

- ViewHAR, 15
- ViewSQL, 15

W

- Web pages
 - CoPS, 25
- Windows 95, 98, ME
 - AUTOEXEC.BAT, 39
- Windows NT, 2000, XP
 - Environment variable, 39
- Windows PC
 - Meaning, 1
- Windows programs
 - Downloading, 25
- Windows XP
 - User Accounts, 4
 - Write error, 4
- WinGEM, 21
- Working directory, 21
- Working in a DOS box, 27
- Write error
 - Windows XP, 4