GPD-6

Seventh edition
August 1997

INSTALLING AND USING

THE SOURCE-CODE VERSION OF GEMPACK

ON DOS/WINDOWS PCs WITH LAHEY FORTRAN

GEMPACK Document No. 6

Copyright 1985-1997 by the Impact Project and KPSOFT

ISSN 1030-2514

This is part of the documentation of the GEMPACK Software System for solving large economic models, developed by the IMPACT Project, Monash University, Clayton Vic 3168, Australia.

Abstract

80386/80486/pentium machines with extended memory running DOS, Windows, Windows 95, Windows NT or OS/2 provide excellent platforms for doing serious general equilibrium modelling. This document describes how to install and use GEMPACK on such machines which have either of the Lahey Fortran compilers F77L-EM/32 or LF90 installed on them.

It also introduces WinGEM, the Windows version of GEMPACK.

Authors and Earlier Editions

Date	*Author(s)*	*Comment*
	[The first two editions of this were numbered GED-29.]	
19/2/90	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]

Table of Contents

1. THE 80386/80486/PENTIUM DOS/WINDOWS VERSION OF GEMPACK 1

1.1 Current Release 1

2. SYSTEM REQUIREMENTS FOR INSTALLING GEMPACK 2

2.1 Summary 2

2.2 Details 2
2.2.1 DOS version 2
2.2.2 Disk space free 2
2.2.3 Memory required 2
2.2.4 Lahey FORTRAN compilers 3

3. INSTALLATION INSTRUCTIONS 4

3.1 F77L3, LF90 and Phar Lap 4

3.2 If You Have An Earlier Release of GEMPACK Installed 4

3.3 Changes to the DOS Settings 4
3.3.1 DOS PATH and AUTOEXEC.BAT 4
3.3.2 CONFIG.SYS 5
3.3.3 If you are installing on a Network 6
3.3.4 Reboot 6

3.4 Pre-installation Check 6

3.5 Copying the GEMPACK Files 6

3.6 Make the GEMPACK Libraries 7

3.7 Making Executable Images 8
3.7.1 Making an Executable Image of TABLO 8
3.7.2 Making an Executable Image of GEMSIM 9

3.8 Memory Required for the Programs 9

3.9 A Final Check 10

3.10 Running GEMPACK Programs Including TABLO-generated Ones 11

3.11 Compiling and Linking TABLO-generated Programs 11

3.12 Installing GEMPACK on a Network 11

4. TESTING THE INSTALLATION 13

4.1 Making a Directory for the Stylized Johansen Model 13

4.2 Simulation via a TABLO-generated Program 13

4.3 Simulation via GEMSIM 14

5. INSTALLING AND TESTING WINGEM 16

5.1 Installing WinGEM 16

5.2 Installing ViewSOL 16

5.3 Testing WinGEM 17

5.4 How WinGEM Works 17

6. FAMILIARISING YOURSELF WITH THE SOFTWARE 19

6.1 Using WinGEM 19

6.2 Not Using WinGEM 19

7. BUILDING YOUR OWN MODELS 20

7.1 New Model's Directory Location 20

7.2 Use TABLO-generated Programs 20

7.3 Using Stored-input Files in a DOS Box 20

7.4 Increasing (or Decreasing) Program Parameters 20

7.5 Changing The Size of Program Stacks for TABLO-generated Programs 21

7.6 Memory Required by TABLO-generated Programs 21

7.7 Compiler Options 22

7.8 Stand-alone Executable Images 23
7.8.1 Making Stand-alone Executable Images 23

7.9 Compiling and Linking TABLO-generated Programs 24

7.10 Interrupting Programs and Controlling Screen Output 24
7.10.1 Interrupting Programs 24
7.10.2 Controlling Screen Output 24

7.11 DOS Batch Files 25

7.12 Summary of Windows NT changes 25
7.12.1 Installation under Window NT 26
7.12.2 Using GEMPACK in a DOS box for Windows NT 26

1. The 80386/80486/pentium DOS/Windows Version of GEMPACK

GEMPACK runs essentially unchanged on a variety of machines. This document tells you how to install the source-code version of GEMPACK on an 80386, 80486 or pentium PC running under DOS, Microsoft Windows or Windows 95 or Windows NT which has either of the Lahey Fortran compilers F77L-EM/32 or LF90 installed, and tells you machine-specific information you may need to know to maximise your use of GEMPACK on such a machine. Indeed this document also applies to 80386, 80486 or pentium machines running other operating systems (such as OS/2) which give users access to a DOS box.

This document describes

1. how to use the source-code version of GEMPACK under DOS or in a DOS box, and
2. how to use **WinGEM**, the Windows interface to GEMPACK under Windows or Windows 95 or Windows NT.

The source-code version of GEMPACK runs in DOS or in a DOS box much like it runs on other machines (including unix machines and Macintosh machines). GEMPACK and the fortran compiler need a DOS box with special options under Windows NT.

WinGEM is only available under Microsoft Windows or Windows 95 or Windows NT.

The user documentation for GEMPACK can be found in the other GEMPACK documents, of which GPD-1[1] *Introduction to GEMPACK* should be your starting point. You can carry out the installation and testing on a DOS machine as described below without being familiar with GEMPACK. But if you intend to use GEMPACK for modelling, we recommend you at least quickly read chapters 1 to 3 in GPD-1 before attempting any modelling on your PC (and perhaps before you install GEMPACK on your PC).

An introduction to the different GEMPACK programs can be found in section 1.1 of GPD-1, while a guide to the models supplied with GEMPACK (including the DOS version) is given in section 1.3 of GPD-1. A guide to the full user documentation for GEMPACK can be found in chapter 9 of GPD-1.

1.1 Current Release

The current version of GEMPACK is Release 5.2-002 (August 1997).

[1] 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 referenced are listed in a separate section at the end of the References section 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 address such comments to one of the authors at the Impact Project.

The numbering of GEMPACK Documents has been re-started with Release 5 of GEMPACK, when the abbreviation "GPD" was first used. Previous editions of these documents did not have the same numbers as the current editions. Pre-Release-5 documents are numbered "GED-xx".

2. System Requirements for Installing GEMPACK

2.1 Summary

1. An 80386 machine with maths coprocessor (80387), or an 80486 DX machine, or an 80486 SX machine with a coprocessor (80487 SX), or a pentium machine.

2. DOS version 3.3 or higher.

3. Hard disk, requiring at least 30 MB free to install and test GEMPACK.

4. At least 8MB bytes of memory (RAM) if you are using F77L-EM/32 and at least 32MB[2] of RAM if you are using LF90.

5. Either the Lahey Fortran 77 compiler **F77L-EM/32** (version 5.01 or later), referred to as F77L3 or the Lahey Fortran 90 compiler **LF90**. Each of these compilers comes with a DOS Extender (Phar Lap) which gives access to the memory above 1MB on your machine. Note that the compiler MUST be installed on your machine BEFORE you install GEMPACK.

Below we give more details about each of these.

2.2 Details

2.2.1 DOS version

You can determine which version of DOS you have by turning on your PC and, at the DOS prompt, entering the DOS command:

ver

2.2.2 Disk space free

You can determine how much disk space is free by entering the DOS command:

dir

The number of bytes free is shown on the screen at the end of the output. (1Mb is about a million bytes, so you will need about 30,000,000 bytes free to install, test and use GEMPACK.) Implementing your own models takes more disk space. As you would expect, the larger the model, the more space you will need.

2.2.3 Memory required

The amount of extended memory you have limits the size of models you can build. Our experience is that many models that are being built now can be implemented with 8MB of RAM and most can be implemented in

[2] At present some of the GEMPACK routines cannot be compiled with LF90 on a PC with 16 Mb; these routines are compiled successfully on a PC with 32 Mb. We hope that, at some time in the not-too-distant future, 16Mb will be suficient with LF90.

16MB of RAM (although more is required to implement very large models such as ORANI and its extensions, or large intertemporal models).[3]

How to check for the amount of available extended memory

You can determine how much extended memory is available on your PC by turning on your PC and at the DOS prompt entering the command:

mem

If the available extended memory is less than 4MB then you may be unable to run GEMPACK software on your PC. See section 3.8 below for more details.

2.2.4 Lahey FORTRAN compilers

The Lahey FORTRAN compilers for 80386/80486/pentium PCs are produced by Lahey Computer Systems Inc. You need either

- their Fortran 77 compiler F77L-EM/32 compiler which is called F77L3,[4] or
- their Fortran 90 compiler LF90.

Instructions for installing either of these compilers and their associated DOS Extender are given in the Lahey Fortran manuals.

If you are using F77L3, you must have version 5.01 or later (and we recommend version 5.1 or later).

If you are using LF90, we strongly recommend that you have version 3.50c or later.

For more information, contact

Lahey Computer Systems Inc
P.O. Box 6091
Incline Village, NV 89450, U.S.A.

Telephones: (800) 548-4778
(702) 831-2500
FAX: (702) 831-8123
Telex: 9102401256

The Australian agent is Devin Trussell at

Computer Transition Systems
Box 4553, Melbourne 3001

Telephone : (03) 9530-6633
FAX: (03) 9530-6644

[3] The two Lahey fortran compilers and Phar Lap can use 'virtual memory'. However we have found that the resulting virtual memory programs run too slowly (when they actually need virtual memory) to be useful. Accordingly the GEMPACK executable images produced by the GEMPACK installation process cannot access virtual memory.
[4] You cannot use either Lahey F77L, Lahey Personal FORTRAN or the discontinued Lahey F77L-EM/16 compilers, and you cannot use versions of F77L3 earlier than version 5.01).

3. Installation Instructions

Most of this installation needs to be carried out either in DOS or in a DOS box.

If you are installing under **Windows NT** the installation should be carried out in a DOS box with special options. We have provided a PIF file called NT-RUN.PIF on the first disk which sets up these options. Before starting the installation, open Windows Explorer and locate the file called NT-RUN on the first GEMPACK disk. Drag this file onto the desktop. Double click on this file to open a special DOS box with the necessary options for installing GEMPACK. Then follow the normal DOS installation procedure. When you are installing GEMPACK, be sure to use NT-RUN to open any DOS box to work in.

[We realise that many Windows users are not familiar with DOS commands, and so we spell out in complete detail the relatively few DOS commands you need to use.]

3.1 F77L3, LF90 and Phar Lap

Before installing GEMPACK you must install your extended memory FORTRAN compiler F77L3 or LF90 and the associated DOS Extender Phar Lap. Follow the instructions in your Lahey FORTRAN compiler manuals.

For use with GEMPACK, you must install at least the following parts of the compiler (see the menu when you run the INSTALL program that comes with these compilers).

The Compiler and Libraries
Phar Lap DOS Extender, Linker, and Library Manager

Of course, you may well wish to install more than these, such as the Sample Programs and the Lahey Blackbeard Editor. You need a text editor for use with GEMPACK and the Lahey Blackbeard. Editor is a good one. [The windows version of GEMPACK also comes with a good text editor GEMEDIT - see section 5.2.]

3.2 If You Have An Earlier Release of GEMPACK Installed

If you have an earlier release of GEMPACK installed on your machine, when installing Release 5.2, you can either choose to leave the earlier version on the disk (Release 5.2 goes in a different directory) or to first remove the earlier version. Indeed, it is probably best (if you have enough disk space) to leave the earlier version on the disk until you have successfully installed and tested Release 5.2 (in case an unexpected problem occurs).

3.3 Changes to the DOS Settings

3.3.1 DOS PATH and AUTOEXEC.BAT

You will install GEMPACK in a directory \GP on part of your hard disk.

The directory \GP must be on the DOS path, as must the directory (usually \F77L3\BIN for F77L-EM/32 or \LF90\BIN for LF90) containing the relevant parts of the fortran compiler. To arrange this, you must edit the appropriate file (called AUTOEXEC.BAT in your default directory \) 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 EDIT which comes with DOS or the Lahey Blackbeard Editor.)

For example, if you find a line

PATH = C:\;C:\DOS

you should change it to

PATH = C:\;C:\DOS;C:\GP;C:\F77L3\BIN (if using F77L-EM/32), or
PATH = C:\;C:\DOS;C:\GP;C:\LF90\BIN (if using LF90)
or
PATH = C:\;C:\DOS;C:\GP;C:\LF9035\BIN (if using LF90 Version 3.5)

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

PATH = C:\GP;C:\F77L3\BIN (if using F77L-EM/32), or
PATH = C:\GP;C:\LF90\BIN (if using LF90)
PATH = C:\GP;C:\LF9035\BIN (if using LF90 Version 3.5)

(If you plan to install GEMPACK on a disk drive different from C:,specify that one in the PATH line.)

Note that the file COMMAND.COM (usually found in directory \ on your starting hard disk) and the external DOS commands (often found in directory \DOS on your starting hard disk) must also be on the PATH. If they are not, add them to the PATH line in your AUTOEXEC.BAT, as described above.

If you plan to install the GEMPACK files in directory C:\GP, no other change is required to AUTOEXEC.BAT. But if you intend to install these files in another directory, you need to add an extra line to AUTOEXEC.BAT. For example, if you intend to install the GEMPACK files in directory D:\PROGRAMS\GP52, add the following line to AUTOEXEC.BAT.

SET GPDIR=D:\PROGRAMS\GP52

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

EXAMPLE

Suppose you have a machine with starting hard disk C:, suppose that COMMAND.COM is in C:\, that your external DOS commands are in C:\DOS, that you have installed F77L3 on drive D: in directory D:\F77L3, and that you intend to install GEMPACK on drive E in directory \GP52. Then your AUTOEXEC.BAT file should contain at least the following:

PATH C:\;C:\DOS;D:\F77L3\BIN;E:\GP52
SET GPDIR=E:\GP52

3.3.2 CONFIG.SYS

Check the file CONFIG.SYS in your default directory \. Look for the lines

FILES = xx
BUFFERS = yy

If necessary, change these (use your text editor) so that the number xx is at least 60 and yy is at least 20. If either of these lines is not present, add new lines

FILES = 60
BUFFERS = 20

as appropriate. (If you do not have a CONFIG.SYS file, create a new one containing the two lines above.)

3.3.3 If you are installing on a Network

If you are installing GEMPACK on a network, please read section 3.12 before proceeding.

3.3.4 Reboot

As you have changed AUTOEXEC.BAT and possibly CONFIG.SYS, reboot your system as follows before proceeding to the rest of the installation:

(a) If you are running Windows, exit from Windows (choose ***File | Exit*** from the Windows main menu) to get back to DOS. Then press Ctrl, Alt and Del simultaneously.

(b) If you are running Windows 95, you will need to restart your computer. To do this, from the ***Start*** menu of Windows 95, select ***Shut down...*** Then select the option "Restart your computer?" and click on ***Yes***.

(c) If you are running Windows NT, you will need to restart your computer. To do this, from the ***Start*** menu of Windows NT, select ***Shut down...*** Then select the option "Restart your computer?" and click on ***Yes***. Open the special DOS box using **NT-RUN.PIF** as described in section 3 above and carry out the rest of the installation in this box.

3.4 Pre-installation Check

Before starting the installation of GEMPACK, we suggest you check that the PATH etc is correctly set. To do this, first exit from Windows (if relevant) or go into a DOS box. Then type the command

set

Check that the PATH is as you expect. If you installed GEMPACK in a directory other than C:\GP, also check that the variable GPDIR has its expected value. If either of these is not as required, re-read section 3.3 (and check that you rebooted). Only proceed when everything is as required.

3.5 Copying the GEMPACK Files

Attach to the relevant hard disk. (If this is C:, type in the command

C:

for example.) Then type in the commands below (as given in bold). Note that some of these commands involve your floppy drive (usually A: or B:). Below we refer to this as A:. If yours is different, you should replace A: appropriately.

1. Create the directory \GP and change directory to it, by typing the commands below.

mkdir \GP
cd \GP

2.If this is the first time you have installed Release 5.2 (or later), you need to copy your GEMPACK licence file LICEN.GEM from the first GEMPACK disk. [This is necessary even if you had a previous version of

GEMPACK since Release 5.2 licence files are different from Release 5.1 licence files.] To do this, insert the first GEMPACK disk into your floppy drive. Type in the next command (replacing A: if necessary).

copy A:licen.gem

3. Leave (or insert) GEMPACK disk 1 in your floppy drive. Type in the next commands (replacing A: if necessary).

copy A:*.bat
gem1 A:

4. When this has finished, insert GEMPACK disk 2 into your floppy drive. Type in the next command (replacing A: if necessary).

gem2 A: 77	(if you are using F77L-EM/32), or
gem2 A: 90	(if you are using LF90)
gem2 A: 35	(if you are using LF90 Version 3.5 or later.)

5. When this has finished, insert GEMPACK disk 3 into your floppy drive. Type in the next command (replacing A: if necessary).

gem3 A:

You have now copied all the files from the GEMPACK disks. The files in directory \GP and its subdirectories will occupy about 5-6Mb of disk at this stage. More precisely, there should be

- at least 17 .FOR files, about 20 .BAT files, 4 .FIG files and at least the files LICEN.GEM, PKUNZIP.EXE, TMEM.EXE, GP52MN.ZIP in directory \GP,
- 35 .ZIP files plus one .FIG file and 4 .BAT files (or 12 .BAT files for LF90) in subdirectory \GP\SUBS,
- 8 .ZIP files, one .FIG file plus 18 other files (with suffix .FOR if you are using F77L-EM/32 or with no suffix if you are using LF90) in subdirectory \GP\TABLO, and
- about 91 files in subdirectory \GP\EXAMPLES (the exact number may be different).

If these are not all there, go through the steps above again.

3.6 Make the GEMPACK Libraries

If you are running Windows 95 it is best to restart your computer in DOS before continuing with the installation. To do this, ***Shut down...*** from the ***Start*** menu of Windows 95. Then select the option "Restart the computer in MS-DOS mode?" and click on ***Yes***.

If you are running Windows NT make sure you are using the special DOS box set up using the PIF file called NT-RUN.PIF as described in section 3 above.

Type in the following commands.

cd \gp
mklib

(There will now be a lot of screen activity. The GEMPACK subroutines are being compiled and added to the library or libraries in \GP. This will take somewhere between 5 and 50 minutes depending on your machine.

Ignore "file not found" messages since some of these are expected.) If you are using F77L-EM/32 these is just one library **GP52.LIB** while if you are using LF90 there are two libraries **GP52A.LIB** and **GP52B.LIB**.

Check that the relevant .LIB file(s) are in directory \GP using the commands

cd \gp
dir *.lib

If you are using F77L-EM/32, the file GP52.LIB should be approximately 2.6 Mb in size. If you are using LF90, the files GP52A.LIB and GP52B.LIB should be approximately 2.4 Mb and 1.7 Mb respectively in size.

If the file(s) are not there or if they are a lot smaller, you will need to redo this. First check again the points in section 3.4 above and the files detailed in section 3.5 above, then repeat this section.

3.7 Making Executable Images

The programs are supplied as Fortran source code (.FOR files). You can see which programs are available via the command

dir \gp*.for

To make the executable image of any of the GEMPACK programs except for TABLO and GEMSIM, you can use the MKMAIN.BAT file in \GP. You simply attach to \GP and then run MKMAIN followed by the relevant program name, for example GEMPIE, as in

cd \gp
mkmain gempie

Later, you can repeat the above for other programs as desired.

At present, we suggest that you make executable images of the most frequently-used GEMPACK programs (other than TABLO and GEMSIM - see below for these) by entering the command

mksome

Note that the executable images produced have extension .EXE (for example SAGEM.EXE).

3.7.1 Making an Executable Image of TABLO

To make an executable image of TABLO, change directory to \GP and run MKTABLO.BAT via the commands

cd \gp
mktablo

This will take several minutes to complete.[5] At the end, the file TABLO.EXE should be in directory \GP.

[5] If you are using LF90 and you receive an "Out of memory" error from the compiler, please rerun this in DOS itself (rather than a DOS box); if the error persists, please contact us at the Impact Project.

3.7.2 Making an Executable Image of GEMSIM

If you have 12 or more megabytes of memory on your PC, first issue the commands

cd \gp
cd tablo

followed by

copy gsinc.gei gsinc.for (if using F77L-EM/32)
copy gsinc2.gei gsinc2.for (if using F77L-EM/32)

copy gsinc.gei gsinc. (if using LF90)
copy gsinc2.gei gsinc2. (if using LF90)

[After these, the executable image of GEMSIM you make will be able to handle larger models than if you had used the default GSINC and GSINC2 files sent on the disks.]

To make an executable image of GEMSIM, change directory to \GP and run MKGEMSIM.BAT via the commands

cd \gp
mkgemsim

This will take several minutes to complete. At the end, the file GEMSIM.EXE should be in directory \GP.

At this point \GP and its subdirectories will occupy about 16Mb of disk space if you are using F77L-EM/32 or about 20 Mb if you are using LF90.

3.8 Memory Required for the Programs

If you have 16Mb or more of memory on your PC, you can skip this section and go to section 3.9.

This test needs to be carried out in DOS itself. A DOS box will not give correct results.

(a) If you are running Windows, exit from Windows (choose ***File | Exit*** from the Windows main menu) to get back to DOS.
(b) If you are running Windows 95, you will need to restart your computer in DOS mode. To do this, select ***Shut down...*** from the ***Start*** menu of Windows 95. Then select the option "Restart the computer in MS-DOS mode?" and click on ***Yes***.

In either of cases (a) or (b) above you can return to Windows after carrying out this test by typing

win

When you start running a fortran program which has been compiled and linked with the Lahey Fortran, the program displays a box which tells you how much memory is available.

You can test how much memory is available to Lahey programs by running the test program TMEM.EXE sent on the GEMPACK disks. Issue the commands

cd \gp

tmem

(i) If you plan to run GEMPACK under DOS (rather than in a DOS box of Windows etc), you need to see that at least 4000K of memory is available.
(ii) If you plan to run GEMPACK under Windows, Windows 95 (or in a DOS box under OS/2), you need to see that at least 6000K of memory is available.

The amounts indicated above[6] will allow you to run all the GEMPACK programs as sent on the installation disks. But they may require more memory if you need to reconfigure them, following the procedure described in section 7.4 below.

If the above test shows that less memory is available than indicated above, you may not be able to run all of the programs as sent. You may be able to increase the amount of memory available by removing device drivers and/or caches. (You will probably need to edit your AUTOEXEC.BAT and/or CONFIG.SYS files to do this. You will need to restart your computer for such changes to take effect.)

3.9 A Final Check

If you are intending to run GEMPACK under Windows, Windows 95 or in a DOS box under OS/2, return to this operating system and go into a DOS box for the next testing.

To check that the DOS PATH has been set correctly, that the DOS variable GPDIR is set appropriately if required (that is, if you installed GEMPACK in a directory other than C:\GP), and that your GEMPACK licence file is in the correct place, issue the following commands

**cd **
tablo

The program TABLO should start running, access the licence file, and then present you with its options screen.

If this happens, all is well. Simply type 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). This will interrupt the program and return you to the DOS prompt. (You may have to type Control-C twice to achieve this.)

If TABLO does not begin running as described above, something in your setup is not correct. First recheck the points in section 3.4 above and then check that the licence file LICEN.GEM is in directory \GP via the commands

cd \gp
dir *.gem

Also check that you have sufficient memory (see section 3.8 above). Once these all appear correct, repeat the test at the start of this section. If all is still not well, you may need to repeat parts of the installation.

3.10 Running GEMPACK Programs Including TABLO-generated Ones

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

sagem

[6] These amounts apply if you are using F77L-EM/32. Possibly a little more is required if you are using LF90.

or

sj

3.11 Compiling and Linking TABLO-generated Programs

In most cases this can be done from any directory by issuing the command LTG followed by the name of the program as in, for example,

LTG SJ

However, for some large programs, it is necessary to use a variation of this to specify the stack size, as explained later in section 7.5.

If you are using Windows NT, use the special DOS box using NT-RUN.PIF as described in section 3.

3.12 Installing GEMPACK on a Network

Some organisations have found it desirable to run the DOS-Lahey 80386/80486/pentium PC source-code version of GEMPACK from a network. For example, this can reduce the need for separate copies of the Lahey compiler.

Below are some pointers to using GEMPACK on a network.

If you have the Lahey fortran F77L3 or LF90 on a network,

(i) you must put the statement

SET DOSX=-SWAPDIR C:\

into your AUTOEXEC.BAT if you are using F77L3. [We are not sure if this is necessary with LF90.]

(ii) you must edit LTG.BAT in the directory where GEMPACK is installed to add after "386link %1" [in the F77L3 case] or after “lf90 %1” [in the LF90 case] "-LIBPATH XXX" where XXX is replaced by the LIB directory where the compiler is installed. For example, if F77L3 is installed in V:\PROD\F77L3, this line of LTG.BAT should begin

386link %1 -libpath v:\prod\f77l3\lib -lib

(iii) Having all .EXE files on the network can have some associated problems because of the differing amounts of memory different users have on their desk PC. For eaxmple, if you link up a very large version of say SAGEM (for 20x20 GTAP say) this may require 30Mb to run. While you may have this memory, another user running smaller models may not have and so won't be able to run the program. You can solve this problem by high-memory users having their own local copy of the EXE. [It is easy to modify LTG.BAT or MKMAIN.BAT to produce the EXE locally rather than on the network.]

(iv) Another issue is who has access to the GEMPACK source etc on the network. Only people with access (write access) can increase parameters etc.

4. Testing the Installation

In this section we suggest that you test the main features of the installation by carrying out a simulation with the Stylized Johansen model in two different ways.[7]

If either of these simulations does not work, you will need to go back to some of the steps in section 3 above. In particular, re-check the points mentioned in sections 3.4 and 3.9 above.

4.1 Making a Directory for the Stylized Johansen Model

First make a new subdirectory, for example \SJ, and copy the files relevant to Stylized Johansen into it as below.

mkdir \sj
cd \sj
copy \gp\examples\sj*.*

4.2 Simulation via a TABLO-generated Program

Run TABLO by typing in the command

tablo

and then give the responses as shown in section 2.6.2 of GPD-1. This should create the TABLO-generated program SJ.FOR. Then compile and link this via the command[8]

LTG SJ

This should create the executable image SJ.EXE. You can check this by entering

dir *.exe

[You have just carried out Steps 1(a) and 1(b) described in section 2.6.1 of GPD-1.]

Next run SJ.EXE via the command

sj

When prompted, give the two responses

cmf
sjlb.cmf

These tell the program to take all input from the GEMPACK Command file SJLB.CMF (see section 2.3.1 of GPD-1). There will be a lot of activity on the screen. When the program finishes running, check that the following message is on the screen near the end:

```
(The program has finished without error.)
```

[7] If you are going to use GEMPACK under Windows or Windows 95 or in a DOS box under OS/2 or Windows NT, you should do this testing in a DOS box.
[8] We only use capitals in this command to distinguish clearly between "el" and "one" at the start.

This run should create the Solution file SJLB.SL4 and a Solution Accuracy file SJLB.XAC. You can check this via the commands

dir sjlb.sl4
dir *.xac

[You have just carried out Step 2 in section 2.6.1 of GPD-1.]

Now run GEMPIE via the command

gempie

to convert the Solution file SJLB.SL4 to the GEMPIE Print file SJLB.PI5 by giving the responses in section 2.2.4 of GPD-1. Check that the results are (approximately) as shown in Table 2.2.4 in section 2.2.4 of GPD-1. [You can use the DOS editor "edit" to check the results via the command

edit sjlb.pi5

Go to the end of the file and check the results against Table 2.2.4 in GPD-1. You can exit from "edit" by typing **Alt-F** (hold down the **Alt** key and hit **F**) and then touching **x** (exit).]

4.3 Simulation via GEMSIM

You might like to test that you can also use GEMSIM to carry out the same simulation. To do so, proceed as follows.

(a) First delete the important output files (produced by the testing in section 4.2 above) via the commands

del *.sl4
del *.xac
del *.pi5

(b) Run TABLO by typing in the command

tablo

and then give the responses as shown in section 2.2.2 of GPD-1. This should create the GEMSIM Auxiliary files SJ.GSS and SJ.GST.

(c) Run GEMSIM by typing in the command

gemsim

As before give the responses

cmf
sjlb.cmf

As above, check that the files SJLB.SL4 and SJLB.XAC have been created.

(d) Run GEMPIE as in section 4.2 above. Check the results in file SJLB.PI5 as before.

If any of these tests does not work, re-check the steps in the installation in section 3 above.

5. Installing and Testing WinGEM

If you are using Microsoft Windows or Windows 95 or Windows NT, you will want to install WinGEM, the Windows interface to GEMPACK. [If you are using a different operating system, you should skip this section and go on to the next section.]

5.1 Installing WinGEM

This installation must be done under Windows, Windows 95 or Windows NT. To install WinGEM, insert the **WinGEM installation disk** into your floppy disk drive.

(a) If you are using Windows, select ***File | Run*** from the Program Manager's main menu.
(b) If you are using Windows 95 or Windows NT, select ***Run...*** from the Start menu.

In either case you need to enter

a:install.exe

(where you should change "a:" to "b:" etc as necesary) and then click on ***Ok***.

This will carry out the installation of WinGEM, during which you will be asked the following questions.

(a) When asked about which directory you wish to install WinGEM in (the installation program refers to this as the "Target Directory"), respond with the name (and drive) of the directory in which you installed GEMPACK (usually C:\GP).
(b) We recommend that you say "yes" when asked if you want to put GEMPACK.INI into C:\GP.
(c) We recommend that you say "yes" when asked if you would like to use WinGEM's default editor GEMEDIT.
(d) We recommend that you say "no" when asked at the end of the installation if you would like to start WinGEM running.

5.2 Installing ViewSOL

This installation must also be done under Windows, Windows 95 or Windows NT. To install ViewSOL, insert the **ViewSOL installation disk** into your floppy disk drive.

(a) If you are using Windows, select ***File | Run*** from the Program Manager's main menu.
(b) If you are using Windows 95 or Windows NT, select ***Run...*** from the Start menu.

In either case you need to enter

a:install.exe

(where you should change "a:" to "b:" etc as necesary) and then click on ***Ok***.

You will be asked which directory you wish to install ViewSOL in.

(a) If you installed GEMPACK in directory C:\GP (the default), you should install ViewSOL in directory C:\GP\WINGEM\VIEWSOL (the default offered by the ViewSOL installation program).

(b) If you installed GEMPACK in a different directory, you should install ViewSOL in a subdirectory VIEWSOL of the subdirectory WINGEM of the directory in which you installed GEMPACK. For example,

if you installed GEMPACK in directory D:\GP52-002, you should install ViewSOL in directory D:\GP52-002\WINGEM\VIEWSOL.

Installing ViewSOL as above is necessary for WinGEM to find ViewSOL.[9]

5.3 Testing WinGEM

We suggest that you carry out Examples 1.1 to 1.5 in Appendix B of GPD-4. This will test that WinGEM (and GEMPACK) are working correctly on your PC. It will also introduce you to the basic features of WinGEM. At the end of Example 1.4, you should see two alternatives ***Go to GEMPIE*** and ***Go to ViewSOL***. If you select ***Go to ViewSOL*** you will see how ViewSOL lets you look at simulation results.

If you have previously used a version of GEMPACK under DOS (or another operating system), you will probably think initially of WinGEM as providing an interface to this DOS version. If you are new to GEMPACK, you should probably not worry about trying to distinguish between WinGEM and GEMPACK.

If any of these examples do not work, you should check the steps in the installation again. You might also like to check if you can carry out the same examples in the DOS box.

Note that WinGEM comes with its own text editor **GEMEDIT** (which can handle large text files up to about 16 Mb in size). We suggest that you use this editor when required to edit text files (such as Command files, TABLO Input files) for GEMPACK. [However, if you are already familiar with another text editor, you can tell WinGEM to use this alternative editor by selecting ***Options | Change editor...*** from WinGEM's main menu.]

WinGEM also comes with the Windows programs **ViewHAR,** which allows you to view the data on GEMPACK's Header Array files directly, and **ViewSOL**, which allows you to view Solution files directly. [Those who have used GEMPACK before can think of ViewHAR as an alternative to the GEMPACK program SEEHAR and ViewSOL as an alternative to the GEMPACK program GEMPIE.] ViewHAR and ViewSOL are written by our colleague Mark Horridge (Centre of Policy Studies and Impact Project, Monash University).

ViewHAR and ViewSOL are not documented in the GEMPACK user documentation. However you will find them well documented via their ***Help*** menus.

5.4 How WinGEM Works

You will see that when you carry out a modelling task using WinGEM, WinGEM starts one of the GEMPACK programs running in a DOS box. The WinGEM program windows are designed to enable you to carry out most modelling tasks simply.

You should be aware that not all modelling tasks are automated via WinGEM. For example, when you use WinGEM to run SEEHAR to look at the data on a Header Array file, you get to look at all the data on the file. If you only want to look at some of the arrays on the file, you will need to run SEEHAR interactively. You could do this by going to a DOS box and running SEEHAR interactively. But you will probably find it easier to do this through WinGEM's ***Programs | Run programs interactively...*** menu.

[9] The only alternative to those described above is to install ViewSOL in some directory which is on the DOS PATH. However we strongly recommend that you locate ViewSOL as indicated earlier.

In running GEMSIM or a TABLO-generated program, it is not possible to give terminal input under the normal **Run** button method where a Command file is selected and used in the simulation. In models which use terminal input, you will need to use **Run interactively** or **Run from STI file.**

WinGEM is really an interface to the DOS version of GEMPACK rather than a separate version of GEMPACK. WinGEM can only function correctly if the DOS version is also functioning properly. This is why we asked you in section 4 above to test the DOS version of GEMPACK before installing WinGEM. Note that WinGEM requires Release 5.2 of GEMPACK - it will not function as expected in conjunction with Release 5.1 or earlier of GEMPACK.

6. Familiarising Yourself with the Software

In this section we give suggestions for hands-on computing which will help you to become familiar with many important features of the use of GEMPACK on 80386/80486/pentium machines. These are based on the models supplied with GEMPACK (see Appendix B of GPD-1), especially the Stylized Johansen and Miniature ORANI models.

Follow section 6.1 below if you are going to use WinGEM (that is, if you are running Windows or Windows 95). Follow section 6.2 below otherwise.

Note that the files corresponding to the example models sent with GEMPACK are all in your directory \GP\EXAMPLES. You should be able to carry out simulations with all of these models in 8Mb of memory (if running DOS directly) or in 12Mb (if running under Windows or Windows 95).

6.1 Using WinGEM

Detailed suggestions for hands-on computing using WinGEM can be found in Appendix B "Getting Started with GEMPACK via WinGEM" of GPD-4. This begins with examples based on the Stylized Johansen and goes on to examples based on Miniature ORANI, GTAP and ORANIF.

6.2 Not Using WinGEM

Simulations with Stylized Johansen

We recommend that you begin by carrying out the hands-on computing Examples in sections E.1 and E.5 of Appendix E of GPD-1. You will not need to do Examples 1-3 or 1aT,1bT,2T,3T there since these are the same as the testing you carried out in sections 4.3 and 4.2 above.

Simulations with Miniature ORANI

Next we recommend that you carry out the hands-on computing Examples in section E.2 and then section E.5.2 of Appendix E of GPD-1. This will involve simulations with Miniature ORANI (first via GEMSIM and secondly using a TABLO-generated program).

Other Hands-on Computing

Then we suggest you work through some of the Examples in sections E.3 and E.4 of Appendix E in GPD-1.

7. Building Your Own Models

This section contains other information relevant to working with GEMPACK on your DOS or Windows machine. Most of it applies whether you are running WinGEM or not (though most of the changes discussed below need to be made in DOS or in a DOS box).

7.1 New Model's Directory Location

We suggest that each new model you build is put in a separate directory on the hard disk, outside of \GP. Note that your PATH command will ensure that the GEMPACK programs are found correctly when, for example, you (or WinGEM) issue the command

tablo

7.2 Use TABLO-generated Programs

You have installed a source-code version of GEMPACK on your PC and have a suitable Fortran compiler (F77L3 or LF90), so 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.)

7.3 Using Stored-input Files in a DOS Box

This section does not apply if you are running WinGEM.

You can use Stored-input files under DOS on 80386/80486 machines either via the GEMPACK **sti** option or using redirection of input as in, for example,

gemsim < sjlb.sti

If you make your own Stored-input files to use via input redirection, 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 setion 5.3 of GPD-1 for more about this.)

7.4 Increasing (or Decreasing) Program Parameters

As indicated in section 5.5 of GPD-1, you may need to increase (or decrease) the size of one or more parameters in a GEMPACK program or a TABLO-generated one.

To do this for a GEMPACK program, change directory to \GP and edit the appropriate source file (for example GEMPIE.FOR for GEMPIE) as described in section 5.5 of GPD-1. Then, to make the new executable image (.EXE file), use MKMAIN (see section 3.7 above) as in, for example,

mkmain gempie

For a TABLO-generated program, edit the source (.FOR file) and then use LTG to create the new executable image as in, for example,

ltg sj

(or else pass in the stack size as a second argument if you need to specify the stack size, as explained in section 7.5 below).

For TABLO, the parameter values are held in the Include files in subdirectory TABLO of \GP. If you are using F77L-EM/32, these Include files have suffix **.FOR** (for example TABLE1.FOR) while if you are using LF90 these Include files have **no suffix** (for example TABLE1.) You should edit the relevant one. (See section 8.2 of GPD-4 and section 2.4 of GPD-2 to find in which Include file each parameter is defined.) Then remake TABLO.EXE by changing directory to \GP and typing in the command mktablo.

For GEMSIM, the parameter values are held in the two Include files in subdirectory TABLO of \GP. If you are using F77L-EM/32 these files have suffix **.FOR** (for example GSINC.FOR) while if you are using LF90 they have **no suffix** (for example GSINC.). You should edit the relevant one. (See section 8.2 of GPD-4 and section 5.13.1 of GPD-2 to find in which Include file each parameter is defined.) Then remake GEMSIM.EXE by changing directory to \GP and typing in the command mkgemsim.

7.5 Changing The Size of Program Stacks for TABLO-generated Programs

You may get an error message from the Phar Lap Dos Extender 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 includes a default stack size following "-St". (To see the current default value, look in this file in directory \GP.) You can increase the stack size for TABLO-generated programs by passing you desired stack size to LTG as a second argument.[10] For example

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.

For more information about stack sizes, see section 2.8.3 of the Lahey manual "DOS Extender & Tools".

7.6 Memory Required by TABLO-generated Programs

As explained in section 5.10.2 of GPD-2, there is a trade-off between speed of execution of TABLO-generated programs and the amount of memory they require. TABLO has various options at the Code stage which affect how much memory the program for a given model will require. (In general, those requiring more memory will run faster.)

If you find that, using the default options in the Code stage of TABLO, you do not have enough memory to run the program, you will need to select one or more of the low memory options described in section 5.10.2 of GPD-2. In particular, read the paragraphs under the heading "Advice on Which of These to Select" near the end of that section.

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

7.7 Compiler Options

The compiler options used in the standard installation (as carried out in section 3 above) are ones that, in our experience, produce executable images which run as fast as possible. In particular, we have not included the compiler option which checks at run time that array subscripts remain within the expected range. [11] Provided there is no bug in the GEMPACK software, such checking should be unnecessary since GEMPACK software also checks array subscripts.

If you prefer to have the extra security provided by option /B, and are not concerned about extra run time[12], you may change the setup on your computer to include option /B by following the steps below.

cd \gp
chkfig

This will copy the options in F77L3CHK.FIG or LF90CHK.FIG to all relevant .FIG files. To implement the option, you must first rebuild the GEMPACK library file(s) via

mklib

Then remake any desired executable images (via mktablo, mkgemsim mkmain sagem and/or "mkmain sagem" etc).

If you are using the F77L3 compiler, to achieve the same bound checking for TABLO-generated programs, type in the command

chkltg

This will put the same options into the batch files LTG.BAT and LTGS.BAT used to compile and link TABLO-generated programs. This change will affect any TABLO-generated program compiled and linked in the future.
[If you are using the LF90 compiler then LTG.BAT and LTGS.BAT use the same LF90.FIG that is present in your GEMPACK directory GPDIR (usually C:\GP).]

If, later, you want to return to the standard (faster running) options, you can do so via the commands

cd \gp
fstfig
mklib
mktablo
mkgemsim
mkmain xxx (for main programs "xxx" as desired)

[11] If you are using F77L-EM/32, this compiler option is denoted by /B ("Check array subscripts and character substring bounds") - see section 2.2.5.2 of the Lahey F77L-EM/32 manual "Programmer's Reference". If you are using LF90 this compiler option is denoted by -chk.
[12] In our experience, adding the compiler option to do bounds checking increases the running time of programs by between 20 and 40 per cent.

7.8 Stand-alone Executable Images

If you copy one of the executable images from your machine to another machine, it will not run unless that machine has Lahey Fortran and Phar Lap installed. This is because the executable images you have made on your machine (using "mkmain" or "ltg") need to be able to find the Phar Lap run-time system.

It is easy to make stand-alone executable images of the various programs. These stand-alone versions have the Phar Lap run-time system included in them so they will run when copied to any suitable DOS machine, that is an 80386 machine with an 80387 coprocessor or an 80486 DX machine or a pentium machine; they are not restricted to machines with Lahey Fortran and Phar Lap installed. We describe below how you can make such stand-alone executable images.

However you should note that the GEMPACK programs TABLO and GEMSIM require a GEMPACK licence (see section 8.3 of GPD-4). 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 you GEMPACK licence. [But you are allowed to send copies of the other GEMPACK programs,. including TABLO-generated ones, outside of the site covered by your GEMPACK licence.]

When you copy a stand-alone executable image to a machine which does not have Lahey Fortran or Phar Lap installed on it, you may need to make changes to the CONFIG.SYS file, following the instructions in section 3.3.2 above.

7.8.1 Making Stand-alone Executable Images

You can make a stand-alone executable image of the GEMPACK programs except for TABLO, GEMSIM and TABLO-generated ones by using the batch file MKMAINS.BAT in \GP. For example, to make a stand-alone executable image of GEMPIE, enter the commands

cd \gp
mkmains gempie

To create a stand-alone executable for a TABLO-generated program, use the batch file LTGS.BAT (the "S" means "stand-alone") in \GP. For example, to make a stand-alone image of SJ, enter the command

ltgs sj

which produces a stand-alone executable SJ.EXE. To carry out multi-step simulations using this TABLO-generated program on other suitable DOS machines, copy SJ.EXE and its associated Auxiliary Statement and Table files SJ.AXS and SJ.AXT to the other machine. You will also need to copy the data file SJ.DAT and probably suitable GEMPACK Command files or Stored-input files.

You can make a stand-alone executable images of TABLO and GEMSIM by using the batch files MKTABLOS.BAT and MKGMSIMS.BAT respectively in \GP. But such images require your GEMPACK licence and must only be used on machines covered by your GEMPACK licence.[13] To make such executable images enter the commands

cd \gp

[13] You will need to copy your GEMPACK licence LICEN.GEM (usually found in directory C:\GP) to the other machine. If you install the GEMPACK executable images in a directory different from C:\GP, you will need to assign a suitable value for the DOS environment variable GPDIR by inserting a line "SET GPDIR=..." in the AUTOEXEC.BAT file on this machine (see section 3.3.1 above). [Of course you are only allowed to copy your licence file to machines within the site covered by your GEMPACK licence.]

mktablos (for TABLO)
mkgmsims (for GEMSIM)

The resulting files TABLO.EXE and GEMSIM.EXE will run on any suitable DOS machine. To use GEMSIM.EXE on such a machine to carry out multi-step simulations with models, you will also need to copy the GEMSIM Statement and Table files (these have suffixes .GSS and .GST) for the model to the other machine. (These files are produced when you run TABLO and ask it to produce output for GEMSIM.) You will also need to copy and data files and relevant Command or Stored-input files.

The batch files MKMAINS.BAT, LTGS.BAT, MKTABLOS.BAT and MKGMSIMS.BAT differ from the non-S versions MKMAIN.BAT, LTG.BAT, MKTABLO.BAT and MKGEMSIM.BAT by linking in a so-called "stub" which is a version of the Lahey/Phar Lap DOS Extender; see section 1.7 of the Lahey manual "DOS Extender & Tools" for more information.

If your TABLO-generated program is one that requires a large stack (see section 7.5bove), you can pass to stack size to LTGS.BAT to make a stand-alone executable as in, for example,

ltgs model 800000

See also section 4.3 of GPD-1 for information about carrying out simulations on other machines.

7.9 Compiling and Linking TABLO-generated Programs

Wherever your TABLO-generated program is located on the disk, you can compile and link it using one of the commands ltg or ltgs depending on whether or not you want to make a stand-alone image. See sections 7.5 and 7.8 above for more details.

7.10 Interrupting Programs and Controlling Screen Output

This section only applies when you are running under DOS or in a DOS box.

7.10.1 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.

7.10.2 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 7.10.1 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.)

7.11 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 SAGEM run ends unsuccessfully, the test of ERRORLEVEL after it aborts the batch job and gives a messsage saying that the job was unsuccessful.

```
REM Beginning of batch file
REM  Run SAGEM
sagem < sag1.sti
REM test ERRORLEVEL to see if this was successful
if errorlevel 1 goto error
REM  Run GEMPIE
gempie < 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
```

7.12 Summary of Windows NT changes

GEMPACK and the fortran compiler need a dos box with special options under Windows NT. We supply a special PIF file on the first installation disk called NT-RUN.PIF. If this is not available you can create your own as follows:

1. Right click on the desktop
2. Select New|Shortcut
3. Type in "command" and click next.
4. Type in a filename, eg "nt-run" and click finish.
5. Right click on the new icon and select Properties.
6. Select the Memory tab.
7. Set Extended (XMS) memory|Total to 1.
8. Click on OK.

This NT-RUN pif file is used for installing GEMPACK on Windows NT and to run GEMPACK jobs in a DOS-like way under NT.

7.12.1 Installation under Window NT

Open Windows NT Explorer and locate the file called **NT-RUN** on the first install disk. Drag this file onto the desktop. Double-click on this file to open it, then follow the normal DOS installation procedure.
(You may like to redirect the working directory of the PIF file to some convenient directory - right click

on the NT-RUN icon and select Properties.)

7.12.2 Using GEMPACK in a DOS box for Windows NT

Be sure to use the NT-RUN file to open the DOS box you work in. Alternatively use WinGEM, which has been designed to work under all Windows platforms. GEMPACK and the Lahey fortran compiler do not always work properly in ordinary Windows NT DOS boxes.

REFERENCES

GEMPACK DOCUMENTS[14]

GPD-1, An Introduction to GEMPACK, Third Edition, September 1996, pp.250+15.

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

GPD-4, Release 5.2 of GEMPACK - New Features and Changes from Release 5.1, First edition, September 1996, pp.98+9.

[14] The numbering of GEMPACK Documents has been re-started with Release 5 of GEMPACK, when the abbreviation "GPD" was first used. Previous editions of these documents did not have the same numbers as the current editions. Pre-Release-5 documents are numbered "GED-xx".

INDEX

—A—

AUTOEXEC.BAT, 4, 6, 10
Auxiliary Statement file, 21
Auxiliary Table file, 21

—B—

batch files, 23
Blackbeard, 4
BUFFERS, 5

—C—

cache, 10
Command file, 12, 15
compiler options, 20
CONFIG.SYS, 5, 6, 10
Control-C, 22
Control-Q, 23
Control-S, 23

—D—

disk space, 2
DOS batch files, 23
DOS path, 4, 10
DOS version, 2

—E—

Editor
  text, 4, 15
executable image
  GEMSIM, 9
  TABLO, 8
executable images, 8

—F—

F77L3, 4
F77L-EM/32, 2, 3
FIG files, 20
FILES, 5
Fortran compiler, 2, 3, 4

—G—

GEMEDIT, 15
GEMPACK library, 8
GEMPACK licence, 7, 21
GEMSIM, 13
  Include files, 19
GEMSIM Auxiliary files, 13
GPDIR, 5, 6, 10

—I—

Include files
  GEMSIM, 19
  TABLO, 19
interrupting programs, 22

—L—

LF90, 2, 4
library, 8
LICEN.GEM, 7, 21
licence file, 7, 10, 21
LTG, 11
LTG.BAT, 19
LTG2, 19

—M—

Machine requirements, 2
memory, 2, 9, 19

—N—

network, 6, 11

—O—

OS/2, 1

—P—

parameters, 18
  GEMSIM, 19
  TABLO, 19
PATH, 4, 6, 10
Phar Lap, 2, 4, 19
program parameters, 18

—R—

reboot, 6

—S—

screen output, 22
simulation, 12, 13
stack size, 19
stand-alone executable images, 21
Stored-input files, 18

—T—

TABLO
  Include files, 19
TABLO-generated programs, 11, 12, 18, 21, 22
text editor, 4, 15

—V—

ViewHAR, 15
ViewSOL, 14

—W—

Windows, 1, 14
Windows 95, 1, 14

Windows NT, 1, 4, 14, 23

WinGEM, 1, 14, 15, 17