MaxFEM

Downloads

Windows

	MaxFEM-0.3.4-win32-setup.exe	(111 MB)	

Debian

	maxfem_0.3.4_all.deb	(110 MB)	

Sources

	maxfem_0.3.4_source.tar.gz	(246 MB)	

Before downloading MaxFEM you should read the information about software requirements, installation and known issues.

An extract of the MaxFEM documentation related to this is show below.

SOFTWARE REQUIREMENTS

Before installing MaxFEM, you should have installed the following software:

Requirements for graphical interface

The dependencies for graphical interface are (*v* means version):

- **Python**: Programming language, $2.5 \leq v < 3.0$.
- **wxPython**: wxWidgets bindings for Python, v 2.8.
- **Python-VTK**: VTK bindings for Python, $v \geq 5.4$.
- **Paramiko**: SSH2 protocol for Python.

Note that Paramiko is only necessary for remote connections. If you don't use it, you don't need it.

To install the previous packages, you can follow these tips:

- **Debian-based Linux distributions**

  Required packages can be downloaded from the official repositories. As an example, how to install them using *apt-get* is shown:

  sudo apt-get install python

  sudo apt-get install python-wxgtk2.8

  sudo apt-get install python-vtk

  sudo apt-get install python-paramiko

- **Windows**

  We recommend a **full installation** of the scientific-oriented Python distribution Python(x,y), containing all the packages required for the graphical interface. Be careful choosing a version of Python(x,y) containing valid versions of the packages mentioned above.

  You can find it in some external mirrors in the Python(x,y) download page. In April 2016, the links to the mirrors were:

  - **NTUA**: http://ftp.ntua.gr/pub/devel/pythonxy/

- **University of Kent**: http://www.mirrorservice.org/sites/pythonxy.com/
- **ConnectMV**: http://pythonxy.connectmv.com/

For problems with graphics or the installation in Windows XP, see **KNOWN ISSUES** below.

If you prefer to install the packages separately, these are the web pages to download them:

- **Python**: https://www.python.org/downloads/
- **wxPython**: http://www.wxpython.org/download.php
- **Python-VTK**: http://www.vtk.org/download/
- **Paramiko**: http://www.paramiko.org/installing.html

If you prefer to build the packages from source, these are the web pages to download the code:

- **Python**: https://www.python.org/downloads/source/
- **wxPython**: http://wxpython.org/download.php#source
- **Python-VTK**: https://gitlab.kitware.com/vtk/vtk
- **Paramiko**: https://github.com/paramiko/paramiko

Requirements for solver compilation and/or execution

The next software requirements are necessary to compile and/or run the MaxFEM solvers (*v* means version):

- **GCC**: The GNU C compiler, *v* > 4.4; recommended, *v* 4.7.
- **GFortran**: The GNU Fortran compiler, *v* > 4.4; recommended, *v* 4.7.
- **Make**: The GNU Make utility to maintain groups of programs, *v* ≥ 3.81.
- **Pthreads**: A POSIX threading library.

The installation of these packages depends on your operating system:

- **Debian-based Linux distributions**

  Required packages can be downloaded from the official repositories. As an example, how to install them using *apt-get* is shown:

  ```
  sudo apt-get install gcc-4.7
  sudo apt-get install gfortran-4.7
  sudo apt-get install make
  sudo apt-get install libc6
  ```

  Note that the pthreads library, *libpthread*, is included in the *libc6* package.

  In some Linux distributions, after installing GCC and GFortran you should have links named *gcc* and *gfortran* pointing to *gcc-4.7* and *gfortran-4.7* respectively. If not, compilation using *make* can fail. You can create them manually using the *ln* command:

  ```
  ln -s /usr/bin/gcc-4.7 /usr/bin/gcc
  ln -s /usr/bin/gfortran-4.7 /usr/bin/gfortran
  ```

- **Windows**

  This additional software should only be installed if you have to compile the MaxFEM solvers, for example, if you build MaxFEM from source. If you install MaxFEM from binaries you should not need this.

  We recommend installing an official build of **MinGW** by using the MinGW Installation Manager (*mingw-get*) to install the following packages:

  - **mingw32-base (bin)**: A basic MinGW installation.
  - **mingw32-gcc-fortran (bin)**: The GNU Fortran compiler.
  - **mingw32-pthreads-w32 (dev)**: A POSIX threading library for Win32.

  Note that *mingw32-base* is a meta package that provides a basic GCC installation, and includes the C compiler, linker and other binary tools, the runtime libraries and Windows API support, mingw32-make and a debugger.

  After installing MinGW we recommend to add the folder with the MinGW binaries **at the beginning** of the *PATH* environment variable in order to prevent conflicts with any other installation of MinGW. For example, with the MinGW that comes with some versions of Python(x,y).

  For problems with MinGW and Pthreads-w32, please read **KNOWN ISSUES** below.

INSTALLATION

Installation of MaxFEM can be done using binaries or building it from source.

Binaries

The installation using binaries depends on your operating system:

- **Debian-based Linux distributions**

  Go to the download section of the MaxFEM web page and download the latest Debian package.

- **Windows**

  Go to the download section of the MaxFEM web page and download the latest Windows installer.

Building from source

How to build MaxFEM from source depends on your operating system:

- **Debian-based Linux distributions**

  Go to the download section of the MaxFEM web page and download the latest source compressed file.

  In a terminal, go to the folder *INSTALLDIR*/sources (where *INSTALLDIR* is the MaxFEM installation folder) and then run:

  ```
  make
  ```

  The previous command is enough to compile all the solvers and the library MUMPS included in MaxFEM.

  If you want to compile only MUMPS, run:

  ```
  make lib
  ```

  If you want to compile only the solvers, run:

  ```
  make solver
  ```

  If you want to clean all temporary files and the result of a compilation, run:

  ```
  make clean
  ```

- **Windows**

  Go to the download section of the MaxFEM web page and download the latest source compressed file.

  In a command window, go to the folder *INSTALLDIR*/sources (where *INSTALLDIR* is the MaxFEM installation folder) and then run:

  ```
  mingw32-make -f Makefile.windows
  ```

  The previous command is enough to compile all the solvers and the library MUMPS included in MaxFEM.

  If you want to compile only MUMPS, run:

  ```
  mingw32-make -f Makefile.windows lib
  ```

  If you want to compile only the solvers, run:

  ```
  mingw32-make -f Makefile.windows solver
  ```

  If you want to clean all temporary files and the result of a compilation, run:

  ```
  mingw32-make -f Makefile.windows clean
  ```

  Note that you have to use *mingw32-make* or only *make* depending on the MinGW version or tool used.

KNOWN ISSUES

- Installation of Python(x,y), version 2.7.5.1, appears to fail in Windows XP. For more information see here.
- Errors related to *_gfortran_stop_numeric_f08* and *pthreadGC2.dll* can arise when a MaxFEM solver is tried to run without having a suitable installation of MinGW.

  In Windows we recommend to install an official build of MinGW with the following packages:

  - **mingw32-base (bin)**: A basic MinGW installation.
  - **mingw32-gcc-fortran (bin)**: The GNU Fortran compiler.
  - **mingw32-pthreads-w32 (dev)**: A POSIX threading library for Win32.

  Please, add the folder with the MinGW binaries **at the beginning** of the *PATH* environment variable in order to prevent conflicts with any other MinGW installation, for example, with the MinGW that comes with some versions of Python(x,y).

- Errors related to the struct *timespec* can arise with some versions of MinGW and Pthreads-w32.

  Some people recommend to comment the definition of the struct *timespec* in *pthread.h* to avoid the conflict with the definition in *parts\time.h*. For more information see here, here, here and here.

- Errors related to *** ERROR RETURN ** FROM ZMUMPS INFO(1)=-13* can arise when there is not enough memory to execute the solver.

  Note that *INFO(1)=-13* indicates a problem of workspace allocation of size *INFO(2)* during the factorization or solve steps. The size that the package tried to allocate with a Fortran ALLOCATE statement is available in *INFO(2)*. You can see the MUMPS documentation for more details.

  We have observed this problem with some examples when we compile in Windows using MinGW. However, we noted that these examples run well when we compile using MinGW-w64.

- If you have problems with graphics, for example, if you can't visualize some graphics or you can't visualize them well, it may be due to the version of Python-VTK. Please, make sure that the version of Python-VTK is greater than or equal to 5.4.

  If you are using a version of VTK greater than or equal to 6.0, please try with the version 5. The latest version of Python(x,y) including VTK version 5 seems to be 2.7.9.0. You can find it in some external mirrors in the Python(x,y) download page.

- Problems running *Magnetostatics 2D* can arise when the solver is obtained by compiling from sources using versions of GCC and/or GFortran different than the recommended *v 4.7*. In these cases, no error message is shown but the results of the execution are wrong because some values are *NaN*.

  We have observed these problems using GCC and/or GFortran *v 4.6* and *v 4.8* in some Linux distributions, however these errors do not appear in Windows.

- If you have problems with codecs, for example, if the application returns a message like *ERROR: AVI codec not found* when you are trying to save an animation, it could be because the required codecs are not included in the VTK distribution used.

  In this case, you have to install a VTK version that has the codecs included or build VTK from source including them. For more information see **Build VTK from source with video codecs** in the help documentation of MaxFEM.

University of Santiago de Compostela | *Telephone: 881 811 000* | *Contact*