UNIX installation of the DISCUS/DIFFEV/KUPLOT software

Preparation:
The compilation requires several libraries, especially their development versions, not all of them may be installed automatically. Especially Ubuntu leaves off many development versions.... Some of these libraries might be present on your system with different version numbers or slightly different names/paths, please be flexible:

```
libX11-devel        ! X11 development library
libm                ! Usually installed
libXmu-devel
libreadline6-dev    ! Needed for command editing
libpng16-devel      ! PNG Graphics library  currently version 16
                    ! The current version is 16, but older versions seem to work fine
                    ! as well
libpgplot           ! Can be installed with UBUNTU, may often have to be installed
                    ! manually, See below
Optional Libraries
Python
OpenMP              ! DIFFUSE needs this for parallel processing
mpich               ! An alternative, OpenMP is preferred
                    ! DIFFUSE needs this for parallel processing
openmpi-dev         ! DIFFUSE needs this for parallel processing
NeXuS               ! In the future DISCUS will use this for 3D data
```

Mandatory development tools:
cmake, ccmake, make
gcc
gfortran ! At least Version 4.6

As an example, with UBUNTU 16.04 or 18.04 use apt-get to install packages:

```
sudo apt-get install libx11-dev
sudo apt-get install libXmu-dev
sudo apt-get install libreadline-dev
sudo apt-get install libopenmp-dev
sudo apt-get install libopenmpi-dev
sudo apt-get install libopenmpi-bin
sudo apt-get install gfortran
sudo apt-get install c++
sudo apt-get install cmake
sudo apt-get install cmake-curses-gui
```

Installation:

Download the latest source code archive from GitHUB at:
github.com/tproffen/DiffuseCode/releases
The archive is called DiffuseCode-V.M.P.tar.gz, where V.M.P stands for the major Version, the Minor version and the Patch numbers, currently 5.20.0
Copy the source code archive to a suitable directory and unpack:

```
mkdir -p $HOME/develop
cp DiffuseCode-5.20.0.tar.gz $HOME/develop
cd $HOME/develop
tar -zxf  DiffuseCode-5.20.0.tar.gz
```

create a „build“ directory, and change to build directory:

```
mkdir -p $HOME/develop/DiffuseBuild
cd $HOME/develop/DiffuseBuild
```

execute ccmake with source code directory as parameter.
ccmake should open a graphical interface:

```
ccmake ../DiffuseCode-5.20.0/
```

ccmake operates mostly via one letter commands, the main are:
c for configure
e exit the message screen
g to generate the make files and exit ccmake

In ccmake toggle OFF the options:
DIFFUSE_PYTHON, DISCUS_CUDA, DISCUS_NEXUS, DISCUS_OMP

In ccmake toggle ON the options:
DIFFEV_MPI

press „t“ to toggle to advanced mode. Go down with cursor and inspect pgplot settings
they should point to the directory in which the pgplot library is found:
/usr/local/pgplot OR may be: /usr/local/lib64/pgplot

The pgplot library need at least the following files in this directory:
grfont.dat
libcpgplot.a or libpgplot.so
libpgplot.a or libpgplot.so
pgxwin_server
Especially if you use a pgplot installation provided by the linux system, these files might be in different directories. It might be best to create a directory
/usr/local/pgplot
and to copy these files into this directory or to create symbolic links within this directory that point to the actual files.

To edit an entry within ccmake hit the „Enter key“ then type or change text.

ccmake wants an entry for „CMAKE_BUILD_TYPE“, edit this field ad leave it blank.

Once done hit „c“ to configure ccmake
You will get an info screen with hopefully no error messages.
If errors are listed, type „e“ and then „q“ and fix the error

If no errors occur hit „e“ to leave the info screen
Hit „g“ to generate the actual make files and to exit ccmake

then you need to compile the program, type without options

```
make
```

If this worked out without error messages you can install DISCUS, DIFFEV etc.
Our default installation directory is /usr/local/bin thus you can:
A) do it with „sudo“
B) Change the ownership of /usr/local to your own account:

A)

```
sudo make install
```

B)

```
sudo chown -R /usr/local  your_user_name
make install
```

With UBUNTU the style B) seems to create issues with other packages and is discouraged.

To clean up type

```
make clean
```

for the on-line help to work, a couple of environment variables should be set:

```
DISCUS_DIR="/usr/local/bin"; export DISCUS_DIR
DIFFEV_DIR="/usr/local/bin"; export DIFFEV_DIR
KUPLOT_DIR="/usr/local/bin"; export KUPLOT_DIR
MIXSCAT_DIR="/usr/local/bin"; export MIXSCAT_DIR
#
PGPLOT_DIR="/usr/local/pgplot"; export PGPLOT_DIR
PGPLOT_DEV="/XSERVE"; export PGPLOT_DEV
```

define these within $HOME/.bashrc.local if a „bash“ is used.

PGPLOT Library

With Ubuntu the PGPLOT library can be installed using the package manager but needs fine tuning. On other systems you might have to install the PGPLOT library locally.

PGPLOT library needs the files:
libpgplot.a
libpgplot.so
grfont.dat
pgxwin_server

in the directory under the PGPLOT_INCLUDE_DIR entry in ccmake you need the files:
cpgplot.h
grpckg1.inc
pgplot.inc
pgxwin_server

Manual installation of LIBPGPLOT:

Make /usr/local your own, or proceed throughout with sudo. This is best on an UBUNTU system, as a change of ownership seems to interfere with other packages and their updates…. Changing the ownership is not an issue with SuSe

sudo chown -R your_user_name /usr/local

INSTALL x11-dev, libreadline6, libpng, libpng-dev

I made small adjustements to the PGPlot library to ease the installation on a linux system. Please download pgplot.5.2.3.tar.gz alond with the DIFFUSE source code. The following instructions apply to this version.

copy pgplot.5.2.3.tar.gz to /usr/local/src

sudo mkdir -p *usr*local/src
sudo cp pgplot.5.2.3.tar.gz /usr/local/src

unpack the library

sudo tar -zxf pgplot5.2.3.tar.gz

Create pgplot directory

sudo mkdir /usr/local/pgplot
cd /usr/local/pgplot

Copy your drivers.list to /usr/local/pgplot

```
sudo cp /usr/local/src/pgplot/drivers.list /usr/local/pgplot
```

Create makefile:

```
cd /usr/local/pgplot
sudo /usr/local/src/pgplot/makemake /usr/local/src/pgplot linux gfortran_gcc
```

ONLY if you have the older version pgplot.5.2.tar.gz, do the following instead:

```
cd /usr/local/pgplot
sudo /usr/local/src/pgplot/makemake /usr/local/src/pgplot linux f77_gcc
```

Edit makefile
change line 25 and 26 to:
25: FCOMPL=gfortran
26: FFLAGC=-ffixed-form -ffixed-line-length-none -u -Wall -fPIC -O

48: Remove "-lf2c"

Copy line 875(?), change to
pndriv.o:

Exit makefile

run makefile with:

```
sudo make
sudo make cpg
sudo make clean
```

If you use the „bash“ then

edit /etc/profile.d/profile.local to contain:

```
PGPLOT_DIR=/usr/local/pgplot
#PGPLOT_DEV=/XSERVE
PGPLOT_DEV=/XWINDOW
export PGPLOT_DIR
export PGPLOT_DEV
```

Edit your local ".basrc", add at end:
source /etc/profile.d/profile.local

Alternatively you can of course edit your local .bashrc.local .

Note on current releases of png_dev libpng16 and later. As the file pngconf has been modified, you

might get an error while compiling file pndriv.c. If this occurs, please edit pndriv.c in the folder drivers and comment lines 225 to 233.

Finally run one or more of the pdgemo programs to verify that the installation proceeded properly. Sometimes, if graphics libraries are missing, the PGPLOT make file seems to quietly turn off the corresponding driver in drivers.list. In this case verify that you have installed the required graphics libraries, especially in their “devel” version. Make sure you edit drivers.list again before compiling the pgplot library.