Install Instructions for dcicpp, the algorithm for the Dynamic Control of Infeasibility method

Abel Soares Siqueira
Advisor: Francisco A. M. Gomes

12 de marco de 2012

Obs. 1: This is a provisory document.

Obs. 2: This guide is for linux.

Obs. 3: The file downall.sh *inside dcicpp will download all dependencies, except base_matrices, Goto BLAS and CUTEr*

Obs. 4: The file downcuter.sh *inside dcicpp will download CUTEr*

All the libraries were downloaded to the same folder in the home folder. For simplicity, we will assume the user do the same. Hence, the first thing to do will be create such folder.

```
$ mkdir $HOME/Libraries
$ cd $HOME/Libraries
```

1 Obtaining

You will need many things to install dcicpp.

1. The source for dcicpp itself

   ```
   $ git clone git://github.com/abelsiqueira/dcicpp.git
   ```

2. base_matrices

   ```
   git clone git://github.com/abelsiqueira/base_matrices.git
   ```

3. CHOLMOD (e outras bibliotecas do Tim Davis)

   ```
   http://www.cise.ufl.edu/research/sparse/cholmod/current/CHOLMOD.tar.gz
   http://www.cise.ufl.edu/research/sparse/amd/current/AMD.tar.gz
   http://www.cise.ufl.edu/research/sparse/camd/current/CAMD.tar.gz
   http://www.cise.ufl.edu/research/sparse/colamd/current/COLAMD.tar.gz
   http://www.cise.ufl.edu/research/sparse/ccolamd/current/CCOLAMD.tar.gz
   http://www.cise.ufl.edu/research/sparse/UFconfig/current/UFconfig.tar.gz
   ```

4. Metis

   `http://glaros.dtc.umn.edu/gkhome/fetch/sw/metis/metis-5.0.2.tar.gz`

5. Goto BLAS. Download from the site

   `http://www.tacc.utexas.edu/tacc-projects/gotoblas2`

Uncompress everything. And get ready to start installing. **Note: we will install everything in 32 bits.**

2 Installing

1.

```
METIS_PATH = $(HOME)/Libraries/metis-5.0.2
METIS = /usr/local/lib/libmetis.a
```

Now, go to the folder CHOLMOD and enter

```
$ make all
```

4. **base_matrices** add the lines from the file in the `addtobash.rc` to your configuration file `$HOME/.bashrc`

   Remove the CUTER parts, if you do not intend to use CUTEr.

   Open a new terminal or use the command

   ```
   $ source $HOME/.bashrc
   ```

   If needed, edit `make.inc` and enter

   ```
   $ make all
   ```

5. **dcicpp** If needed, edit `make.inc` and enter

   ```
   $ make all
   ```

Your installation is now complete. You can test and compare this algorithm with CUTEr. To do so, proceed to the next section.

3 CUTEr

First, download CUTEr using svn

```
$ svn co https://magi-trac-svn.mathappl.polymtl.ca/SVN/cuter/sifdec/branches/SifDec2 ./sifdec2
$ svn co https://magi-trac-svn.mathappl.polymtl.ca/SVN/cuter/cuter/branches/cuter64 ./cuter2
```

Now download the SIF problems

```
$ wget ftp://ftp.numerical.rl.ac.uk/pub/cuter/mastsif_small.tar.gz
$ wget ftp://ftp.numerical.rl.ac.uk/pub/cuter/mastsif_large.tar.gz
```

Uncompress all files

```
$ tar -zxf mastsif_small.tar.gz
$ tar -zxf mastsif_large.tar.gz
```

We need to add some lines to `$HOME/.bashrc` that depend on your configuration. We will use the following options with CUTEr, if you need to change them, then the lines could change too.

- PC

- Linux
- gfortran
- gnu g++
- double
- large

The following lines should be in your `$HOME/.bashrc`. If you did not remove, they are in the `addtobash.rc` file.

```
ROOTCUTER="$HOME/Libraries"

export CUTER="$ROOTCUTER/cuter2"
export MYCUTER="$CUTER/CUTEr.large.pc.lnx.gfo"
export SIFDEC="$ROOTCUTER/sifdec2"
export MYSIFDEC="$SIFDEC/SifDec.large.pc.lnx.gfo"
export MASTSIF="$ROOTCUTER/mastsif"
export MANPATH="$CUTER/common/man:$SIFDEC/common/man:$MANPATH"
export PATH="$MYCUTER/bin:$MYSIFDEC/bin:$PATH"
```

Open a new terminal or use the command

```
$ source $HOME/.bashrc
```

Now, go to the sifdec2 folder and edit the file `config/linux.cf`.

Search for lsxxx, where xxx reflects your choice of fortran compiler. In your case, we used `gfortran`, so we search for lsgfo. Then, add -m32 to `FortranFlags`. Also, edit the file `config/all.cf` Search for lsgpp and add -m32 to CFlags

Back in the sifdec2 directory, enter

```
$ ./install_sifdec
```

When asked if you want to run `install_mysifdec` and follow the same instruction as above.

Create a directory and test cuter with the command

```
$ runcuter -p gen -D ROSENBR
```

Now, compile dcicpp for cuter. In the dcicpp folder, enter

```
$ make cuter
```

See `TESTING.cuter` inside dcicpp to proceed with testing dcicpp with CUTEr