DensityBounder

This directory contains the C++ source code (DensityBounder.cpp) for DensityBounder , an implementation of the algorithm used to calculate upper bounds of Turán densities for 3-graphs. It has been tested on Windows XP using Visual C++ 2008, and on Ubuntu 9.10 (Linux) using gcc 4.4.1, both running on a Pentium 4 CPU with 1 GB of RAM. The program exits immediately after finishing, and therefore should be run from a command-line interface rather than through the graphical user interface.

It is by default setup to calculate an upper bound for $\pi(\mathcal{F}')$. This can be changed by commenting out

```
#define _HYPERGRAPHS_DO_JUMP_
```

and uncommenting one of

```
#define _FORBIDDING_K4MINUS_,
#define _RAZBOROV_FORBIDDING_K4_.
```

The positive semidefinite matrices used in each of the three problems can also be found in this directory: HypergraphsDoJump.soln, RazborovK4.soln and K4-.soln. Alternatively they can be computed using DensityBounder and a semidefinite program solver.

To compute the positive semidefinite matrices, DensityBounder generates a ".dat-s" file (the specific name for each problem is stored in the `filenameSDP[]` array). The ".dat-s" file will be the input to the semidefinite program solver. Next we use the semidefinite program solver to output a solution into a ".out" file (given by `filenameOutput[]`). In particular to get a solution using CSDP's standalone solver we type

```
csdp HypergraphsDoJump.dat-s HypergraphsDoJump.out
```

at the command-line. DensityBounder then takes the ".out" file, removes rounding errors, and stores the result in a ".soln" file (given by `filenameSoln[]`).

Once the ".soln" file has been created or loaded, DensityBounder calculates an upper bound for the Turán density using only integer type variables. It avoids any use of floating point numbers at this stage so that no rounding errors can occur.