A PARALLEL IMPLEMENTATION OF GNFS IN MATHEMATICA

1. A Brief Review of Modules

In this section we will briefly review some of essential Modules that will be used through the algorithm, but we will not go through technical and mathematical details and difficulties.

1.1. **CalcBounds.** This module simply determines theoretically the best bounds for several parameters of the algorithm depend on our choice of size for calculation. We will refer to all available options for size of calculation in description of `ProcessGNFS`.

1.2. **PolyLNorm.** This module computes norm of a polynomial introduced by Morphy to choose the best one.

1.3. **SelectMonicPolynomial, SelectMonicPolynomialParallel, ParallelSelPol.** These modules are main routines for polynomial selection. First module works on a single CPU and the next two modules working on both single and multiple CPU chips.

1.4. **GenerateRFB, GenerateAFB, GenerateQBC.** These routines generating three different factor bases according to previously determined bounds. Actually AFB and RFB working on one big prime mode which is suitable for integers with less than 200 digits.

1.5. **SmoothOverRFB, SmoothOverAFB.** These modules check the smoothness over integer numbers and over ideal factor base.

1.6. **LinearSieveRegion, ParallelSieve.** First module implements linear sieve over factor bases for a certain sieve region, the second module, distributes sieve over all available CPUs.

1.7. **BuildMatrix.** After sieving step, the algorithm needs to solve a very large linear system, this module generates the linear system.

1.8. **ScanningNullSpace, ParallelScanNullSpace.** These modules scanning the null space of solutions of the linear system for all possible combination of pairs. First module works on a single CPU and second works on multiple CPU.

1.9. **FindSquares.** The solutions of linear system are the source of generating desirable difference of squares, this module checks the solutions for a proper difference of squares.

1.10. **ProcessGNFS.** This is the main module combining all other defined modules for running GNFS algorithm.

2. Configurations

To initiate variables one needs to set `Dir`, which locates working path on the host computer for I/O, so make sure that the assigned path is not read only and current user has permission to write. `Echo` (default value = `True`. `False` causes to hide all visual information that generated during the run time, but it still saves all data in I/O mode.

The parameters of `ProcessGNFS[n,Size,mod]` are followings:

- **n**: The integer number to factor.
- **Size**: The selected scale for determining parameters and bounds which accept the following values:
  - "s" for small scale,
  - "m" for medium scale,
  - "l" for large scale,
  - "xl" for extra large scale which may cause in a higher degree for polynomial selection,
  - "h" for huge scale which actually using a higher degree for polynomial and extremely wide region for sieve and huge bounds for factor bases.
- **mod**: Determines the method for running the algorithm; Three options are available:
  - "Simple": Runs the algorithm on a single CPU (probably best choice for small integers with less than 20 digits).
  - "Parallel": Runs the algorithm on all available CPU's.
  - "PIO": Runs the algorithm on all available CPU's with this possibility that one can abort the process without losing many previous calculations. This mode automatically saves the results of each step on a file. Next time one runs the program with the same parameters it loads all previous results and continues from last break point.