Blue Distribution 1.1.0
4th March 2014

Changes from Blue 1.0.1 to 1.1.0

- Improved performance and scaling.

  Performance improved by about 40% for bacterial-like data, and much more for human data. Much of the latter improvement came from better handling of extremely deep-coverage reads, and not spending inordinate amounts of time trying to correct read artefacts that were never going to be successfully corrected anyway.

- Reduced memory usage.

  Better allocation of the memory used to hold the k-mer consensus tables.

- Added –*fixed* and –*variable* options.

  By default, Blue will always maintain the length of the reads it corrects. It does this by either padding or trimming those reads whose length has changes through insertions or deletions. The –*variable* option stops this happening and reads are allowed to grow or shrink.

- Added –*paired* and –*unpaired* options.

  By default, if Blue is asked to correct a set of files it will treat them as a set and maintain the read pairing between each file in the set when writing out the corrected reads. If the *good* option is used to discard poor reads after correction, then both members of a pair will be discarded if any one of them fails the sufficient-good-k-mers test. The –*unpaired* option stops this behaviour, and the –*good* option will only discard those reads that actually fail the specified goodness test.

- Keeping good-but-unpaired reads.

  If Blue is correcting a set of 'paired' reads, those reads that pass the goodness test but have mates that fail the test are now written to a '_singles' file. The failing reads themselves will be written to a '-problems' file if the –*problems* option is set.

- Added a –help option to display a fuller version of the command-line options

Changes from Tessel 1.0.1 to 1.1.0

- Reduced memory requirements and improved performance.

  Tessel is now lock-free, allowing it to scale better on multi-processor systems. The memory allocation algorithms have also been tuned to use less memory.

Changes from GenerateMerPairs 1.0.1 to 1.1.0

- Reduced memory requirements and improved performance.

  GenerateMerPairs is now lock-free, allowing it to scale better on multi-processor systems. The memory allocation algorithms have also been tuned to use less memory.