Olden Benchmarks Suite

Bisort

Benchmark

Bitonic sorting algorithm implementation.

Definition
*A sorted sequence is a monotonically non-decreasing (or non-increasing) sequence. A **bitonic** sequence is a sequence with* $x_0 \leq \ldots \leq x_k \geq \ldots \geq x_{n-1}$ *for some k, or a circular shift of such a sequence.*

Data Structures

containers.h
containers.cpp

1) **Binary tree** based implementation (recursive calls for left and right subtrees)
2) std::sort() algorithm based on **std::vector**
3) **C array** based implementation (recursive calls on left and right array halves)

Computational method

The algorithm is based on a sorting comparator network consisting of several layers. The network can be and is implemented in a divide and conquer way similar to that of a well-known merge sort. Sort() function is called on the left and right array halves recursively. The merging step of the merge sort algorithm is substituted with compare-and-swap step. The latter is possible due to input sequences required to be bitonic.

Implementation details

Binary tree based bitonic sort implementation
olden_bitonic_sort.h
olden_bitonic_sort.cpp

C array bitonic sort implementation
bitonic_sort.h
bitonic_sort.cpp

Feasibility results