

## Poster C-41

### Metamorphosis - Sorting by Weighted Inversions, Transpositions, and Inverted Transpositions



#### Authors:

Martin Bader (*University of Ulm*)

Simon Gog (*University of Ulm*)

Enno Ohlebusch (*University of Ulm*)

**Short Abstract:** We have implemented a 1.5-approximation algorithm for Sorting by Weighted Reversals, Transpositions, and Inverted Transpositions. The algorithm can handle any weight ratio from 1:1 to 2:1 (transpositions:reversals). The algorithm performs very well in practice and is available as a web application.

#### Long Abstract:

During evolution, genomes are subject to genome rearrangements that alter the ordering and orientation (strandedness) of genes on the chromosomes. Because these events are rare compared to point mutations, they can give us valuable information about ancient events in the evolutionary history of organisms. For this reason, one is interested in the most "plausible" genome rearrangement scenario between two (or multiple) species. More precisely, given two genomes, one wants to find an optimal (shortest) sequence of rearrangement operations that transforms one into the other. Here we will focus on genomes that consists of a single (circular) molecule of DNA such as mitochondrial, chloroplast or bacterial genomes. As usual, the genomes are represented by a signed permutation, i.e., an ordering of signed genes where the sign indicates the orientation (the strand). In the single chromosome case, the relevant genome rearrangements are inversions (where a section of the genome is excised, reversed in orientation, and reinserted) and transpositions (where a section of the genome is excised and reinserted at a new position in the genome; if this also involves an inversion, one speaks of an inverted transposition). As is usually done in bioinformatics, we will use the terms "reversal" and "transreversal" as synonyms for "inversion" and "inverted transposition." It is well known that the problem of finding an optimal sequence of rearrangement operations that transforms a permutation into another permutation is equivalent to the problem of "sorting" a permutation by the same set of operations into the identity permutation. Let us briefly recall what is known for various sets of operations. In a seminal paper, Hannenhalli and Pevzner showed that the problem of sorting by reversals can be solved in polynomial time. The Hannenhalli-Pevzner theory was simplified and the running time of their algorithm was improved several times. To date, a subquadratic time algorithm is available, and the reversal distance problem (which asks solely for the minimum number of required reversals, but not for the sequence of reversals) is solvable in linear time. It is also worth mentioning that the problem of sorting an unsigned permutation by reversals is NP-hard and the currently best approximation algorithm has the performance ratio 1.375. If one restricts the set of operations to transpositions (T), to transpositions and reversals (T + R), or to transpositions, reversals, and transreversals (T + R + TR), the complexity of the problem is still unknown. There exist polynomial-time approximation algorithms (1.375-approximation for T, 2-approximation for T + R, and

1.5-approximation for  $T + R + TR$ ). The biologically most relevant scenario is the  $T + R + TR$  case because in reality genomes are reorganized by all three kinds of operations. A drawback of Hartman and Sharan's 1.5-approximation algorithm is that it applies only to the case in which reversals and transpositions are weighted equally. Because a transposition can create two cycles in the reality-desire diagram while a reversal can create at most one cycle, the algorithm generally favors transpositions. Consequently, the sequence of rearrangement operations returned by that algorithm will often significantly deviate from the "true" evolutionary history because in most organisms transpositions are observed much less frequently than reversals. Thus, it is desirable to have the possibility of weighting reversals and transpositions differently. Given such weights, the weighted genome rearrangement problem asks for a sorting sequence of rearrangement operations such that the sum of the weights of the operations in the sequence is minimal. That is, a shortest sequence is not necessarily optimal. In a recent RECOMB paper, we have presented a 1.5-approximation algorithm for all weight proportions between 1:1 and 2:1 (transposition:reversal). This closes the gap between the results of Hartman and Sharan (1.5-approximation for ratio 1:1) and Eriksen (1+epsilon-approximation for ratio 2:1). We have implemented the algorithm, and improved the quality in practice by combining it with a greedy strategy. The algorithm has a good running time ( $O(n^2)$  without greedy and  $O(n^3)$  with greedy) and is available as a web application.