A New Data Structure for Genome Rearrangement Problems [1]

Cleber V. G. Mira, [2] Joao Meidanis, [3]

Keywords: Genome Representation, Computational Biology

1 Statement of the Problem.

A DNA chromosome has two strands with complementary orientation. A set of genes E and a permutation Γ over E is called a *gene system*, denoted by (E, Γ), when $Supp(\Gamma) = E$ and $\Gamma\Gamma = \iota$. The permutation Γ associates each gene to its complementary. For instance, given the set of genes $E = \{-4,\ -3,\ -2,\ -1,\ 1,\ 2,\ 3,\ 4\}$ and the permutation $\Gamma = (1\ -1)(2\ -2)(3\ -3)(4\ -4)$, the pair (E, Γ) is a gene system since the support of Γ is E and $\Gamma\Gamma x = x$ for any $x \in E$.

Given a gene system (E, Γ), a cycle α is called a *strand* when $x \in Supp(\alpha)$ implies $\Gamma x \notin Supp(\alpha)$ for each $x \in E$. The *conjugation* of a permutation α by β, denoted by $\beta \cdot \alpha$, is $\beta\alpha\beta^{-1}$ [2]. A *chromosome* is a product of two strands α and $\Gamma \cdot \alpha^{-1}$. Two chromosomes are *disjoint* when their supports are disjoint. A *genome* is a product of disjoint chromosomes. A fundamental property of genomes is: if π is a genome, then $\Gamma\pi\Gamma = \pi^{-1}$.

A genome rearrangement problem consists of finding a minimum sequence of mutational events called *rearrangement events* that transform a genome into another.

2 Data Structure

The design of new data structures [3] allowed the improvement in performance of algorithms for genome rearrangement problems involving signed reversals [4] and transpositions [1]. We present a new data structure that is based on the algebraic formalism for genome rearrangements [2]. The Figure 1 illustrates the data structure.

The data structure can be used, with a few modifications, in algorithms for different genome rearrangement problems (e.g. involving generalized transpositions or fusions, fissions, and signed reversals).

Given the genomes π and σ in (E, Γ), we can obtain a minimum sequence of fusions, fissions, and signed reversals that transforms the genome π into σ in $O(n)$, where $n = |E|$, using the data structure in Figure1.

References

[1] T. Hartman and R. Shamir, 2006. A simpler and faster 1.5-approximation algorithm for sorting by transpositions. In: *Information and Computation*. Preliminary version appeared in the Proceedings of CPM'03.204(2):275–290, 2006

[2] J. Meidanis and Z. Dias, 2000. An Alternative Algebraic Formalism for Genome Rearrangements. In: *Comparative Genomics: Empirical and Analytical Approaches to Gene Order Dynamics, Map Alignment and Evolution of Gene Families* "D. Sankoff and J. H. Nadeau",editors, Kluwer Academic Publishers. pp 213–223.

---

[1] Research supported by grants from FAPESP(03/00731-3).

[2] Institute of Computing, University of Campinas (UNICAMP), Sao Paulo, Brazil E-mail: `cleber@ic.unicamp.br`

[3] Scylla Bioinformatics, Sao Paulo, Brazil E-mail: `meidanis@scylla.com.br`

P	1	2	3	4	5	6	7	8	−1	−2	−3	−4	−5	−6	−7	−8
	−4	−3	−2	−1	6	5	3	1	−8	7	−7	8	−6	−5	2	4

invP	1	2	3	4	5	6	7	8	−1	−2	−3	−4	−5	−6	−7	−8
	8	−7	7	−8	6	5	−2	−4	4	3	2	1	−6	−5	−3	−1

S	1	2	3	4	5	6	7	8	−1	−2	−3	−4	−5	−6	−7	−8
	2	3	4	5	6	7	8	1	−8	−1	−2	−3	−4	−5	−6	−7

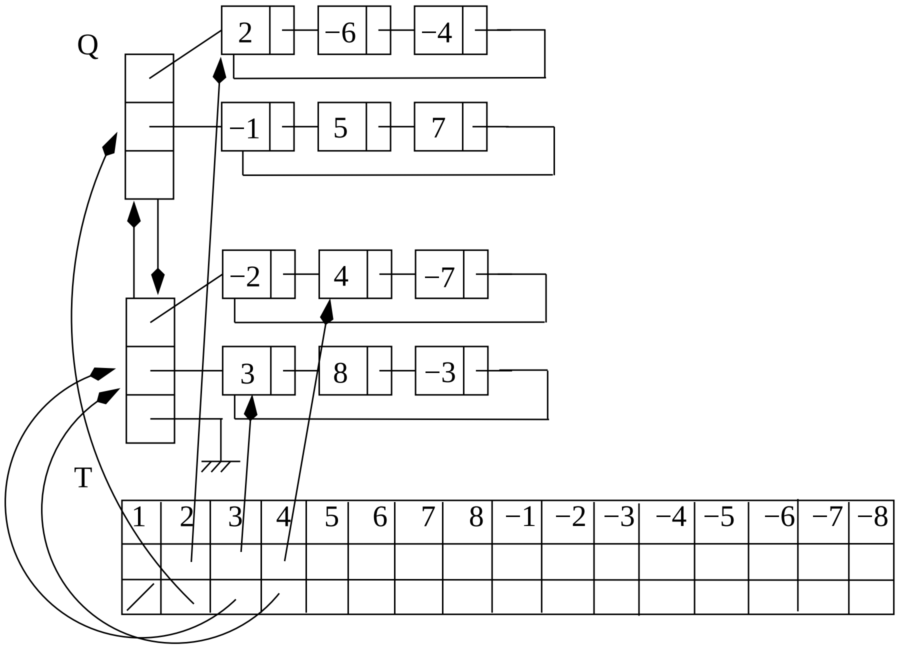

Figure 1: Genomes $\pi = (3\ \ -2\ \ 7)(-7\ \ 2\ \ -3)(-5\ \ -6)(6\ \ 5)(1\ \ -4\ \ 8)(-8\ \ 4\ \ -1)$ and $\sigma = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8)(-8\ -7\ -6\ -5\ -4\ -3\ -2\ -1)$ in the gene system (E,Γ) where $E = \{x \mid |x| \in [8]\}$ and $\Gamma x = -x$ are represented by arrays P and S. The permutation $\sigma\pi^{-1}$ is represented by the list Q. The array T makes an entry indexed by a gene x to point to the node containing x and to the pair of circular lists in Q. In order to make the figure clear, we show the pointers of the first four genes of T only.

[3] H. Kaplan and E. Verbin, 2003. Efficient data structures and a new randomized approach for sorting signed permutations by reversals. In: *Combinatorial Pattern Matching, 14th Annual Symposium, CPM 2003, Morelia, Michocán, Mexico, June 25-27, 2003, Proceedings*, volume 2676 of *Lecture Notes in Computer Science*. Springer, 2003.

[4] E. Tannier and M. Sagot, 2004. Sorting by reversals in subquadratic time. In: *Technical Report 5097, INRIA, Institut National de Recherche en Informatique et en Automatique, January 2004*.