On the DCJ Median Problem

Mingfu Shao and Bernard M.E. Moret

Laboratory for Computational Biology and Bioinformatics
EPFL, Switzerland
{mingfu.shao,bernard.moret}@epfl.ch

Abstract. As many whole genomes are sequenced, comparative genomics is moving from pairwise comparisons to multiway comparisons framed within a phylogenetic tree. A central problem in this process is the inference of data for internal nodes of the tree from data given at the leaves. When phrased as an optimization problem, this problem reduces to computing a median of three genomes under the operations (evolutionary changes) of interest. We focus on the universal rearrangement operation known as double-cut-and join (DCJ) and present three contributions to the DCJ median problem. First, we describe a new strategy to find so-called adequate subgraphs in the multiple breakpoint graph, using a seed genome. We show how to compute adequate subgraphs w.r.t. this seed genome using a network flow formulation. Second, we prove that the upper bound of the median distance computed from the triangle inequality is tight. Finally, we study the question of whether the median distance can reach its lower and upper bounds. We derive a necessary and sufficient condition for the median distance to reach its lower bound and a necessary condition for it to reach its upper bound and design algorithms to test for these conditions.

Keywords: genomic rearrangement, network flow, dynamic programming.

1 Introduction

The combinatorics and algorithmics of genomic rearrangements have seen much work since the problem was formulated in the 1990s [1]. Genomic rearrangements include inversions, transpositions, circularizations, and linearizations, all of which act on a single chromosome, and translocations, fusions, and fissions, which act on two chromosomes. These operations can all be described in terms of the single double-cut-and-join (DCJ) operation [2, 3], which has formed the basis for most algorithmic research on rearrangements since its publication [4–9]. A DCJ operation makes two cuts in the genome, either in the same chromosome or in two different chromosomes, producing four cut ends that it then rejoins, giving rise to three possible outcomes.

A basic problem in genome rearrangements is to compute the edit distance between two genomes, i.e., the minimum number of operations that are needed to transform one genome into another. Under the inversion model, Hannenhalli and Pevzner gave the first polynomial-time algorithm to compute the edit distance

A.S. Kulikov, S.O. Kuznetsov, and P. Pevzner (Eds.): CPM 2014, LNCS 8486, pp. 273–282, 2014.

between two unichromosomal genomes [10]; a linear-time algorithm for the same problem was later designed [11]. Under the DCJ model, the edit distance can also be computed in linear time, this time for two multichromosomal genomes [2]. The median problem is a generalization of the edit distance: given three genomes, we want to construct a fourth genome, the *median*, that minimizes the sum of the edit distances between itself and each of the three given genomes. The median problem is NP-hard for almost all formulations [12, 13]. Under the inversion model, several exact algorithms [14, 15] and heuristics [16, 17] have been proposed. Under the DCJ model, Zhang *et al.* presented an exact solver using a branch-and-bound framework [18]. In [19], Xu *et al.* proposed a decomposition scheme that preserves optimality by using *adequate subgraphs*, particular substructures of the multiple breakpoint graph [20]. Later, Xu produced the ASMedian software to implement a median search based on adequate subgraphs using an optimistic branch-and-bound search [21]. ASMedian uses a precomputed set of small adequate subgraphs; at each step, it tests whether the current multiple breakpoint graph contains a subgraph from that set.

We propose a new strategy to find adequate subgraphs in the multiple breakpoint graph, based on a *seed genome*. We give a polynomial-time algorithm to decide whether there exists an adequate subgraph w.r.t. this seed genome (and to identify such a subgraph if one exists) using a network flow formulation.

The DCJ *median distance* (the sum of the distances of the given genomes to their median) can be lower- and upper-bounded using the sum of the three pairwise DCJ edit distances among the three given genomes. The lower bound was recently proved to be tight [22]. We show that the upper bound is also tight. Moreover, we give testable characterizations of the equality problem: for a given instance, is the median distance equal to its upper or lower bound? We give a necessary and sufficient condition for equality with the lower bound—the necessary condition can be tested using a dynamic programming formulation—and we give a necessary condition for equality with the upper bound, a condition that can also be tested effectively.

2 Preliminaries

We assume that each genome consists of the same set of n distinct genes and that those genes form one or more circular chromosomes in each genome. The head and tail of a gene g, represented by g_h and g_t, are called *extremities*. Two consecutive genes form one *adjacency*, represented as the set of its two extremities. Since all genes are distinct, each genome is uniquely determined by its n adjacencies. We build a graph (V, E), where V has $2 \cdot n$ vertices representing the extremities and E has n edges representing the adjacencies. Note that a genome thus corresponds to a perfect matching on V (see Fig. 1).

Given genomes G_1 and G_2 represented by perfect matchings M_1 and M_2 on V, the corresponding *breakpoint graph* is defined as the multigraph (V, M_1, M_2). In the multigraph, two vertices may be connected by two edges, one from M_1 and the other from M_2. These edges are distinguished by their provenance.

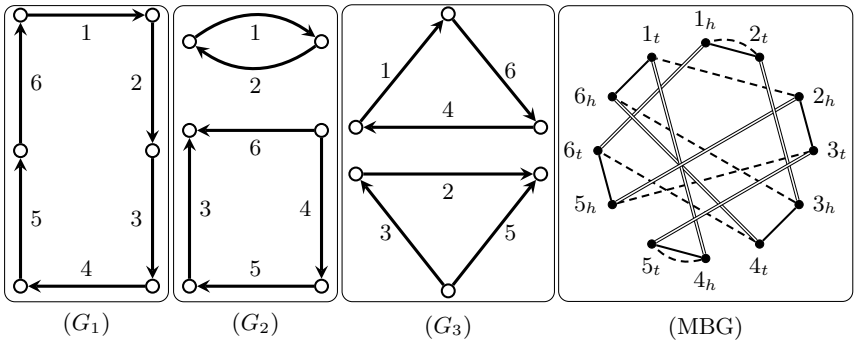

Fig. 1. Three genomes and the corresponding complete MBG. Genes, adjacencies, and extremities are represented by arrows, circles, and solid circles, respectively. Adjacencies in G_1, G_2, and G_3 are represented by solid, dashed and double lines respectively.

Each vertex in this breakpoint graph has degree 2, so that the graph consists of vertex-disjoint cycles; let $c(M_1, M_2)$ denote the number of these cycles. The *DCJ distance* between G_1 and G_2, denoted as $d(M_1, M_2)$, can be expressed as $d(M_1, M_2) = n - c(M_1, M_2)$ [2]. We can extend this concept to three given genomes, M_1, M_2 and M_3, yielding a *multiple breakpoint graph* (MBG for short, see an example in Fig. 1), denoted by (V, M_1, M_2, M_3). Given a MBG (V, M_1, M_2, M_3), the DCJ median problem asks for a perfect matching M_0 on V (another genome) that minimizes $\sum_{k=1}^{3} d(M_0, M_k)$.

We generalize the definition of MBG by allowing nonperfect matchings, distinguishing MBGs with three perfect matchings as *complete* MBGs. If M_1' and M_2' are not perfect matchings on V', then the breakpoint graph (V', M_1', M_2') consists of isolated vertices, simple paths, and vertex-disjoint cycles; we continue to use $c(M_1', M_2')$ to denote the number of cycles.

Let B' be a MBG and B a complete MBG; B' is a *subgraph* of B if we have $V' \subseteq V$ and $M_k' \subseteq M_k$, $k = 1, 2, 3$. A matching M_0' on V' is a *median* of B' if it maximizes $\sum_{k=1}^{3} c(M_0', M_k')$ over all possible matchings on V'. B' is *adequate* if, for any median M_0' of B', we have $\sum_{k=1}^{3} c(M_0', M_k') \geq 3 \cdot |V'|/4$.

Theorem 1. *[19] If B' is an adequate subgraph of B, then for any median M_0' of B', there exists one median M_0 of B such that $M_0' \subset M_0$.*

This result leads to a decomposition scheme to compute the median by iteratively finding adequate subgraphs and resolving each separately; ASMedian uses a precomputed set containing all adequate subgraphs with size less than 10.

3 Adequate Subgraphs w.r.t. a given Matching

We describe a new algorithm to compute adequate subgraphs in a complete MBG, based on the use of a "seed" genome—a perfect matching on V. In practice, this seed genome can be one of the three given matchings. Let M be a

perfect matching on V. An MBG $B' = (V', M'_1, M'_2, M'_3)$ is adequate w.r.t. M if there exists a matching M' on V' satisfying $\sum_{k=1}^{3} c(M', M'_k) \geq 3 \cdot |V'|/4$ and $M' \subseteq M$. If B' is adequate w.r.t. M, then clearly it is adequate. Given a complete MBG $B = (V, M_1, M_2, M_3)$ and a perfect matching M on V, let $\mathcal{C}_k$ be the set of cycles in the breakpoint graph (V, M_k, M), $k = 1, 2, 3$, and write $\mathcal{C} = \cup_{k=1}^{3} \mathcal{C}_k$. For a cycle $C \in \mathcal{C}$, let $V(C)$ be the set of vertices covered by C and $E(C)$ be the set of edges covered by C. For a subset $\mathcal{S} \subseteq \mathcal{C}$, set $V(\mathcal{S}) = \cup_{C \in \mathcal{S}} V(C)$ and $E(\mathcal{S}) = \cup_{C \in \mathcal{S}} E(C)$.

Lemma 1. *There exist adequate subgraphs of B w.r.t. M iff there exists a subset $\mathcal{S} \subseteq \mathcal{C}$ obeying $|\mathcal{S}| \geq 3 \cdot |V(\mathcal{S})|/4$.*

Proof. If such $\mathcal{S}$ exists, we can define the subgraph as $(V(\mathcal{S}), M_1 \cap E(\mathcal{S}), M_2 \cap E(\mathcal{S}), M_3 \cap E(\mathcal{S}))$. Let $M' = M \cap E(\mathcal{S})$; then the sum $\sum_{k=1}^{3} c(M', M_k \cap E(\mathcal{S}))$ is exactly equal to $|\mathcal{S}|$, which is larger than or equal to $3 \cdot |V(\mathcal{S})|/4$. Thus, our subgraph is adequate w.r.t. M. Conversely, suppose that there exists one adequate subgraph (V', M'_1, M'_2, M'_3) of B w.r.t. M and let $M' \subseteq M$ be a matching on V' satisfying $\sum_{k=1}^{3} c(M', M'_k) \geq 3 \cdot |V'|/4$. Let $\mathcal{S}$ be the set of all cycles in the three breakpoint graphs (V', M', M'_k), $k = 1, 2, 3$. We can write $|\mathcal{S}| = \sum_{k=1}^{3} c(M', M'_k)$. Since M'_1, M'_2 and M'_3 are all matchings on V', we have that $|V'| \geq |V(\mathcal{S})|$. Combining these formulas yields $|\mathcal{S}| \geq 3 \cdot |V(\mathcal{S})|/4$. □

We use a network flow formulation to compute such $\mathcal{S}$. Fig. 2 illustrates the construction. We add to N one vertex for each extremity in V, one vertex for each cycle in $\mathcal{C}$, plus a source s and sink t. We add to N directed edges of capacity $3/4$ from s to each extremity in V and directed edges of capacity 1 from each cycle in $\mathcal{C}$ to t. For each pair of $v \in V$ and $C \in \mathcal{C}$ with $v \in V(C)$, we add one directed edge of infinite (very large) capacity from v to C. Let f be a maximum s-t flow of N, N_f the residual network w.r.t. f, S the set of vertices reachable from s in N_f, and T the set of all other vertices.

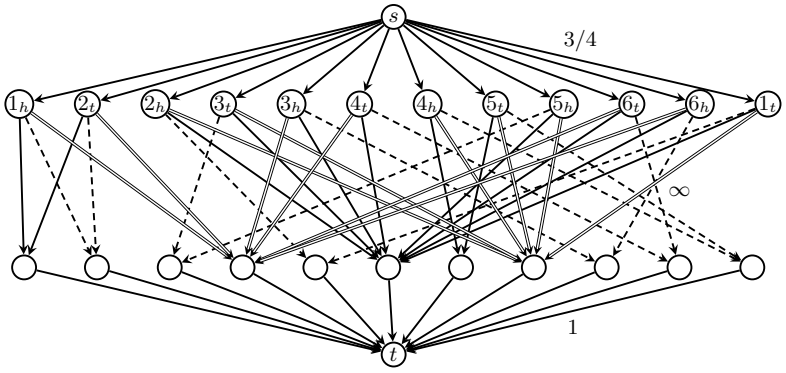

Fig. 2. The network for the complete MBG of Fig. 1 with the seed $M = M_2$

Lemma 2. *A subset $\mathcal{S} \subseteq \mathcal{C}$ with $|\mathcal{S}| \geq 3 \cdot |V(\mathcal{S})|/4$ exists iff we have $\{t\} \subsetneq T$.*

Proof. By construction of S and T, we must have $s \in S$ and $t \in T$; moreover, (S, T) is a minimum s-t cut of N. For any other minimum s-t cut (S', T'), we have $|S| \leq |S'|$. The total capacity of cut (S, T) is at most $|\mathcal{C}|$, since it is a minimum s-t cut and there is a trivial s-t cut (containing just the sink t on one side) whose total capacity is $|\mathcal{C}|$.

Assume we have $\{t\} \subsetneq T$. Let $\mathcal{S} \subseteq \mathcal{C}$ be the set of cycles in T and let $V' \subseteq V$ be the set of extremities that are in T. The edges of infinite capacity cannot belong to the (S, T) cut, so that the total capacity of the (S, T) cut is exactly $3 \cdot |V'|/4 + |\mathcal{C}| - |\mathcal{S}|$. Since the total capacity of any minimum s-t cut is at most $|\mathcal{C}|$, we must have $|\mathcal{S}| \geq 3 \cdot |V'|/4$. Because the edges of infinite capacity are not in the (S, T) cut, we also have $V(\mathcal{S}) \subseteq V'$. Thus, we can conclude $|\mathcal{S}| \geq 3 \cdot |V(\mathcal{S})|/4$.

Now assume there exists a subset $\mathcal{S}$ satisfying $|\mathcal{S}| \geq 3 \cdot |V(\mathcal{S})|/4$. We prove $\{t\} \subsetneq T$ by contradiction. Assume $T = \{t\}$; then the total capacity of the cut (S, T) is $|\mathcal{C}|$. Now we construct another s-t cut (S', T'), where T' consists of the extremities in $V(\mathcal{S})$ and the cycles in $\mathcal{S}$ and sink t. The capacity of this cut (S', T') is $3 \cdot |V(\mathcal{S})|/4 + |\mathcal{C}| - |\mathcal{S}|$, less than or equal to $|\mathcal{C}|$ since we have $|\mathcal{S}| \geq 3 \cdot |V(\mathcal{S})|/4$. Thus (S', T') is also a minimum s-t cut, but clearly we have $|S'| < |S|$, the desired contradiction. □

Thus, if there exist adequate subgraphs w.r.t. a perfect matching, one such subgraph can be found from the residual network.

4 The Upper Bound is Tight

Let M_0 be a median of a complete MBG $B = (V, M_1, M_2, M_3)$. We denote by $d_m = \sum_{k=1}^{3} d(M_0, M_k)$ the median distance of B and by $d_t = d(M_1, M_2) + d(M_1, M_3) + d(M_2, M_3)$ the *triangle distance* of B. According to the triangle inequality (the DCJ distance is a metric), we have $d(M_0, M_i) + d(M_0, M_j) \geq d(M_i, M_j)$, $1 \leq i < j \leq 3$, which yields a lower bound for the median distance, $d_m \geq d_t/2$. However, by using any of M_1, M_2, and M_3 as a possible median, we get $d_m \leq d(M_1, M_2) + d(M_1, M_3)$, $d_m \leq d(M_2, M_1) + d(M_2, M_3)$, and $d_m \leq d(M_3, M_1) + d(M_3, M_2)$, which yields an upper bound for the median distance, $d_m \leq 2 \cdot d_t/3$. Fig. 3 shows a subgraph where the upper bound is reached. Notice that this subgraph is also adequate. Thus, the combination of any number of copies of this subgraph yields a graph that also reaches the upper bound.

5 Deciding Equality to the Bounds

We now study whether the median distance of a complete MBG reaches its lower or upper bound. Let $u, v \in V$ be two distinct vertices. A DCJ operation *induced* by (u, v) on M_1 removes (u, u_1) and (v, v_1) from M_1 and adds (u, v) and (u_1, v_1) to M_1, where u_1 and v_1 are the neighbors of u and v in M_1. (If u is matched to v in M_1, then the DCJ operation induced by (u, v) on M_1 is an identity.)

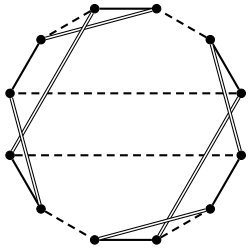

Fig. 3. Tightness of the upper bound. M_1, M_2, M_3 are represented by solid, dashed and double edges. We have $d(M_1, M_2) = d(M_1, M_3) = d(M_2, M_3) = 4$ and thus $d_t = 12$. Any M_k is a median with $d_m = \sum_{k=1}^{3} d(M_1, M_k) = 8$. Thus we have $3 \cdot d_m = 2 \cdot d_t$.

Property 1. Let M and M_1 be two perfect matchings on V and $u, v \in V$ two distinct vertices with $(u, v) \in M$ and $(u, v) \notin M_1$. Then we can write $d(M, M_1') = d(M, M_1) - 1$, where M_1' is the perfect matching obtained from M_1 after performing the DCJ operation induced by (u, v).

Definition 1. (u, v) *is* strong *w.r.t.* M_1 *and* M_2 *if* u *and* v *are in the same cycle of* (V, M_1, M_2) *and the distance between them is odd—see Fig. 4. Otherwise,* (u, v) *is* weak *w.r.t.* M_1 *and* M_2.

Property 2. Let M_1' and M_2' be the two perfect matchings after performing the two DCJ operations induced by (u, v) on M_1 and M_2 respectively. Then (u, v) is strong w.r.t. M_1 and M_2 iff we have

$$d(M_1', M_2') = \begin{cases} d(M_1, M_2) & \text{if } (u, v) \in M_1 \cap M_2; \\ d(M_1, M_2) - 1 & \text{if } (u, v) \in (M_1 - M_2) \cup (M_2 - M_1); \\ d(M_1, M_2) - 2 & \text{if } (u, v) \notin M_1 \cup M_2. \end{cases}$$

Two strong edges (u, v) and (u', v') w.r.t. M_1 and M_2 are *independent* w.r.t. M_1 and M_2 if (i) they are in different cycles of (V, M_1, M_2) or (ii) they do not "intersect" in the same cycle—where an intersection would mean that u' and v' are on the different paths from u to v.

Property 3. Let (u, v) be a strong edge w.r.t. M_1 and M_2, and M_1' and M_2' be the matchings after performing two DCJ operations induced by (u, v) on M_1 and M_2 respectively.

(a) If (u', v') is weak w.r.t. M_1 and M_2, then (u', v') is weak w.r.t. M_1' and M_2'.
(b) If (u', v') is strong w.r.t. M_1 and M_2 and (u, v) and (u', v') are independent w.r.t. M_1 and M_2, then (u', v') is strong w.r.t. M_1' and M_2'.
(c) If (u', v') is strong w.r.t. M_1 and M_2 and (u, v) and (u', v') are not independent w.r.t. M_1 and M_2, then (u', v') is weak w.r.t. M_1' and M_2'.
(d) If, w.r.t. M_1 and M_2, (u', v') and (u'', v'') are strong, (u, v) and (u', v') are independent, (u, v) and (u'', v'') are independent, but (u', v') and (u'', v'') are not independent, then (u', v') and (u'', v'') are (strong but) not independent w.r.t. M_1' and M_2'.

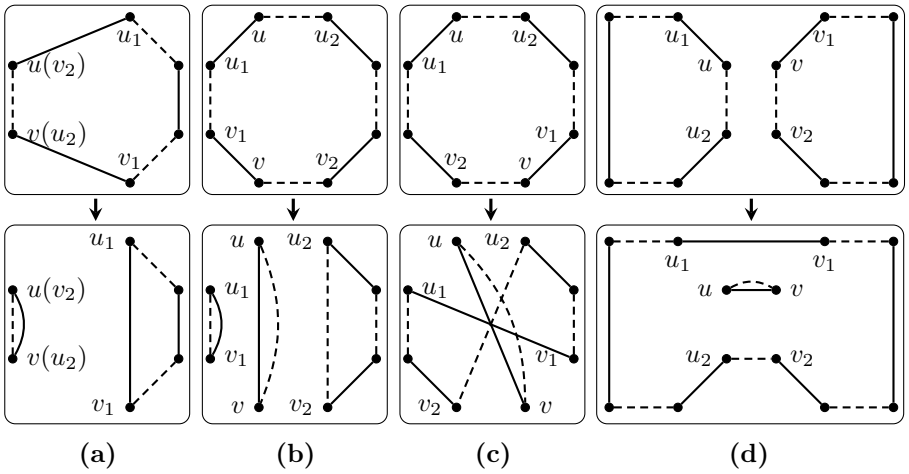

Fig. 4. The four cases for two DCJ operations induced by edge (u, v) on M_1 and M_2 (represented by solid and dashed edges respectively). u_1 and v_1 (u_2 and v_2) are the neighbors of u and v in M_1 (M_2). **(a)** u and v are neighbors in M_2; **(b)** u and v are in the same cycle at odd distance; **(c)** u and v are in the same cycle at even distance; and **(d)** u and v are in different cycles. In (a) ad (b), (u, v) is strong w.r.t. M_1 and M_2.

Lemma 3. *Let M, M_1 and M_2 be three perfect matchings on V. We have $d(M, M_1) + d(M, M_2) = d(M_1, M_2)$ iff M consists of n mutually independent strong edges w.r.t. M_1 and M_2.*

Proof. Choose one edge from M that is not in $M_1 \cup M_2$ and perform the DCJ operations induced by this edge on M_1 and M_2, and repeat until no more such operations can be performed. Let $2 \cdot o$ be the number of DCJ operations performed in this process and let M_1^* and M_2^* be the final matchings thus obtained. We must have $M = M_1^*$ or $M = M_2^*$ since at the final state we cannot find any edge in M that is not in $M_1 \cup M_2$. Without loss of generality, assume $M = M_1^*$. Using Property 1, we have $d(M, M_1) = d(M, M_1^*) + o = d(M_1^*, M_1^*) + o = o$ and $d(M, M_2) = d(M, M_2^*) + o = d(M_1^*, M_2^*) + o$.

Assume M consists of n mutually independent strong edges w.r.t. M_1 and M_2. By Property 3(b), all edges used to perform DCJ operations must be strong w.r.t. their current states. Using Property 2, we get $d(M_1^*, M_2^*) = d(M_1, M_2) - 2 \cdot o$ and thus also $d(M, M_1) + d(M, M_2) = d(M_1, M_2)$. Now suppose that there exists an edge in M that is weak w.r.t. M_1 and M_2 or that there exist two edges in M that are not independent. By the end of the iterative process, all edges in M are mutually strong w.r.t. M_1^* and M_2^*. By Property 3, there exists a weak edge that is used to perform DCJ operations. Thus, by Property 2, we have $d(M_1^*, M_2^*) > d(M_1, M_2) - 2 \cdot o$, which implies $d(M, M_1) + d(M, M_2) > d(M_1, M_2)$. □

(u, v) is strong w.r.t. to $B = (V, M_1, M_2, M_3)$ if (u, v) is strong w.r.t. M_1 and M_2, M_1 and M_3, and M_2 and M_3. Two strong edges (u_1, v_1) and (u_2, v_2) w.r.t.

B are independent w.r.t. B if they are independent w.r.t. M_1 and M_2, M_1 and M_3, and M_2 and M_3.

Lemma 4. *We have $d_m = d_t/2$ iff there are n mutually independent strong edges w.r.t. B.*

Proof. We have $d_m = d_t/2$ iff there exists a perfect matching M_0 of V satisfying $d(M, M_i) + d(M, M_j) = d(M_i, M_j)$ for all $1 \leq i < j \leq 3$. By Lemma 3, such matching consists exactly of n mutually independent strong edges w.r.t. B. □

Lemma 5. *Assume $M_1 \cap M_2 \cap M_3 = \emptyset$; then we have $d_m = 2 \cdot d_t/3$ only if there is no strong edge w.r.t. B.*

Proof. Assume edge (u, v) is strong w.r.t. B and let M_0 be a median of B. We have three cases. First, assume $(u, v) \notin M_1 \cup M_2 \cup M_3$. We perform the DCJ operations induced by (u, v) on M_1, M_2 and M_3. Let M'_k, $k = 1, 2, 3$, be the corresponding new matchings and denote by $B' = (V, M'_1, M'_2, M'_3)$ be the new complete MBG. We have $(u, v) \in \cap_{k=1}^{3} M'_k$ and the subgraph induced by $\{u, v\}$ is clearly adequate, so that there exists a median of B', call it M'_0, with $(u, v) \in M'_0$. Set $d'_m = \sum_{k=1}^{3} d(M'_0, M'_k)$ and $d'_t = d(M'_1, M'_2) + d(M'_1, M'_3) + d(M'_2, M'_3)$. Since each DCJ operation can increase the DCJ distance by at most one, we have

$$d_m \leq \sum_{k=1}^{3} d(M'_0, M_k) \leq \sum_{k=1}^{3} (d(M'_0, M'_k) + 1) = d'_m + 3.$$

However, since (u, v) is strong w.r.t. B, by Property 2, we have $d(M'_1, M'_2) = d(M_1, M_2) - 2$, $d(M'_1, M'_3) = d(M_1, M_3) - 2$, and $d(M'_2, M'_3) = d(M_2, M_3) - 2$, which gives us $d'_t = d_t - 6$. Applying the upper bound on B', we get $d'_m \leq 2 \cdot d'_t/3$. By combining these formulas, we finally get $d_m \leq d'_m + 3 \leq 2 \cdot (d_t - 6)/3 + 3 = 2 \cdot d_t/3 - 1$, implying that the upper bound cannot be achieved.

Second, assume $(u, v) \in M_1 - (M_2 \cup M_3)$. Now we perform the DCJ operations induced by (u, v) on just M_2 and M_3. We can thus write $d_m \leq d'_m + 2$. By Property 2 and using the fact that M'_1 is just M_1, we can write $d(M'_1, M'_2) = d(M_1, M_2) - 1$, $d(M'_1, M'_3) = d(M_1, M_3) - 1$, and $d(M'_2, M'_3) = d(M_2, M_3) - 2$, which gives us $d'_t = d_t - 4$. We thus get $d_m \leq d'_m + 2 \leq 2 \cdot (d_t - 4)/3 + 2 = 2 \cdot d_t/3 - 2/3$, implying that the upper bound cannot be achieved.

Third, assume $(u, v) \in (M_1 \cap M_2) - M_3$. Now we perform the DCJ operation induced by (u, v) on M_3 only. By similar reasoning, we get $d_m \leq d'_m + 1$, $d(M'_1, M'_2) = d(M_1, M_2)$, $d(M'_1, M'_3) = d(M_1, M_3) - 1$, and $d(M'_2, M'_3) = d(M_2, M_3) - 1$. Thus, we have $d_m \leq d'_m + 1 \leq 2 \cdot (d_t - 2)/3 + 1 = 2 \cdot d_t/3 - 1/3$, implying again that the upper bound cannot be achieved. □

The necessary condition of Lemma 5 is not sufficient, as illustrated in Fig. 5: the subgraph shown has no strong edge, but the median distance is not equal to its upper bound. Since the subgraph is adequate, we can build a general example by combining an arbitrary number of copies of this subgraph.

By Lemma 4, we can decide whether the median distance reaches its lower bound by checking whether there exist n mutually independent strong edges

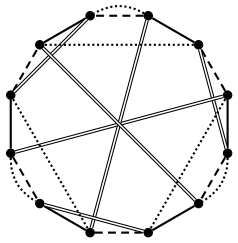

Fig. 5. A subgraph with no strong edge where the median distance does not reach its upper bound. Matchings M_1, M_2, M_3, and M_0 are represented by solid, dashed, double and dotted edges respectively. We have $d(M_1, M_2) = d(M_1, M_3) = d(M_2, M_3) = 5$, yet $d(M_0, M_1) = 2$, $d(M_0, M_2) = d(M_0, M_3) = 3$.

w.r.t. B. We can reduce this question to a maximum independent set problem by setting a vertex for each strong edge and linking two strong edges if they are not independent. Clearly, there exist n mutually independent strong edges iff the size of the maximum independent set is n. The independent set problem is NP-hard, but we can test in polynomial-time whether there exist n mutually independent strong edges w.r.t. M_1 and M_2, a necessary condition. The algorithm enumerates all possible strong edges w.r.t. M_1 and M_2; this can be done in $O(n^3)$ time. Let $C_1, C_2, \ldots, C_m$ be the cycles in the breakpoint graph (V, M_1, M_2). Because each strong edge must have both endpoints on the same cycle, we can handle each cycle separately. For cycle C_i with $V(C_i)$ vertices, we use dynamic programming to compute the maximum number of non-crossing edges, taking time in $O(|V(C_i)|^3)$. If this maximum number is less than $V(C_i)/2$, then we cannot find enough independent strong edges and thus the algorithm returns false. If the algorithm terminates after examining all cycles, it returns true. The total running time is $O(n^3)$. The necessary condition of Lemma 5 can be tested in $O(n^3)$ time as well, by checking each pair of vertices to see whether it is strong w.r.t. B.

Acknowledgements. We thank Yu Lin for helpful discussions.

References

1. Fertin, G., Labarre, A., Rusu, I., Tannier, E., Vialette, S.: Combinatorics of Genome Rearrangements. MIT Press (2009)
2. Bergeron, A., Mixtacki, J., Stoye, J.: A unifying view of genome rearrangements. In: Bücher, P., Moret, B.M.E. (eds.) WABI 2006. LNCS (LNBI), vol. 4175, pp. 163–173. Springer, Heidelberg (2006)
3. Yancopoulos, S., Attie, O., Friedberg, R.: Efficient sorting of genomic permutations by translocation, inversion and block interchange. Bioinformatics 21(16), 3340–3346 (2005)
4. Alekseyev, M.A., Pevzner, P.A.: Whole genome duplications, multi-break rearrangements, and genome halving problem. In: Proc. 18th ACM-SIAM Symp. Discrete Algs. SODA 2007, pp. 665–679. SIAM Press (2007)

5. Braga, M.D., Willing, E., Stoye, J.: Double cut and join with insertions and deletions. J. Comput. Biol. 18(9), 1167–1184 (2011)
6. Chen, X., Sun, R., Yu, J.: Approximating the double-cut-and-join distance between unsigned genomes. In: Proc. 9th RECOMB Workshop Compar. Genomics RECOMB-CG 2011, BMC Bioinformatics 12(S.9), S17 (2011)
7. Shao, M., Lin, Y.: Approximating the edit distance for genomes with duplicate genes under DCJ, insertion and deletion. In: Proc. 10th RECOMB Workshop Compar. Genomics RECOMB-CG 2012, BMC Bioinformatics 13(S. 19), S13 (2012)
8. Shao, M., Lin, Y., Moret, B.M.E.: Sorting genomes with rearrangements and segmental duplications through trajectory graphs. In: Proc. 11th RECOMB Workshop Compar. Genomics RECOMB-CG 2013, BMC Bioinformatics 14(S. 15), S9 (2013)
9. Moret, B.M.E., Lin, Y., Tang, J.: Rearrangements in phylogenetic inference: Compare, model, or encode? In: Chauve, C., et al. (eds.) Models and Algorithms for Genome Evolution, Computational Biology, vol. 19, pp. 147–172. Springer (2013)
10. Hannenhalli, S., Pevzner, P.A.: Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). In: Proc. 27th ACM Symp. Theory of Computing STOC 1995, pp. 178–189. ACM Press (1995)
11. Bader, D.A., Moret, B.M.E., Yan, M.: A fast linear-time algorithm for inversion distance with an experimental comparison. J. Comput. Biol. 8(5), 483–491 (2001)
12. Caprara, A.: The reversal median problem. INFORMS J. Comput. 15, 93–113 (2003)
13. Tannier, E., Zheng, C., Sankoff, D.: Multichromosomal genome median and halving problems. In: Crandall, K.A., Lagergren, J. (eds.) WABI 2008. LNCS (LNBI), vol. 5251, pp. 1–13. Springer, Heidelberg (2008)
14. Siepel, A.C., Moret, B.M.E.: Finding an optimal inversion median: experimental results. In: Gascuel, O., Moret, B.M.E. (eds.) WABI 2001. LNCS, vol. 2149, pp. 189–203. Springer, Heidelberg (2001)
15. Moret, B.M.E., Siepel, A.C., Tang, J., Liu, T.: Inversion medians outperform breakpoint medians in phylogeny reconstruction from gene-order data. In: Guigó, R., Gusfield, D. (eds.) WABI 2002. LNCS, vol. 2452, pp. 521–536. Springer, Heidelberg (2002)
16. Arndt, W., Tang, J.: Improving reversal median computation using commuting reversals and cycle information. J. Comput. Biol. 15(8), 1079–1092 (2008)
17. Rajan, V., Xu, A.W., Lin, Y., Swenson, K.M., Moret, B.M.E.: Heuristics for the inversion median problem. In: Proc. 8th Asia-Pacific Bioinf. Conf. APBC 2010, BMC Bioinformatics 11(S. 1), S30 (2010)
18. Zhang, M., Arndt, W., Tang, J.: An exact solver for the DCJ median problem. In: Proc. 14th Pacific Symp. Biocomputing PSB 2009, pp. 138–149 (2009)
19. Xu, A.W., Sankoff, D.: Decompositions of multiple breakpoint graphs and rapid exact solutions to the median problem. In: Crandall, K.A., Lagergren, J. (eds.) WABI 2008. LNCS (LNBI), vol. 5251, pp. 25–37. Springer, Heidelberg (2008)
20. Alekseyev, M.A., Pevzner, P.A.: Breakpoint graphs and ancestral genome reconstructions. Genome Research 19(5), 943–957 (2009)
21. Xu, A.W.: A fast and exact algorithm for the median of three problem: A graph decomposition approach. J. Comput. Biol. 16(10), 1369–1381 (2009)
22. Aganezov, S., Alekseyev, M.A.: On pairwise distances and median score of three genomes under DCJ. In: Proc. 10th RECOMB Workshop Compar. Genomics RECOMB-CG 2012, BMC Bioinformatics 13(S.19), S1 (2012)