involve
a journal of mathematics

Proving the pressing game conjecture on linear graphs

Eliot Bixby, Toby Flint and István Miklós

msp

2016 vol. 9, no. 1

INVOLVE 9:1 (2016)
dx.doi.org/10.2140/involve.2016.9.41

Proving the pressing game conjecture on linear graphs

Eliot Bixby, Toby Flint and István Miklós

(Communicated by Joshua Cooper)

The pressing game on black-and-white graphs is the following: given a graph $G(V, E)$ with its vertices colored with black and white, any black vertex v can be pressed, which has the following effect: (1) all neighbors of v change color; i.e., white neighbors become black and vice versa; (2) all pairs of neighbors of v change adjacency; i.e., adjacent pairs become nonadjacent and nonadjacent ones become adjacent; and (3) v becomes a separated white vertex. The aim of the game is to transform G into an all-white, empty graph. It is a known result that the all-white empty graph is reachable in the pressing game if each component of G contains at least one black vertex, and for a fixed graph, any successful transformation has the same number of pressed vertices.

The pressing game conjecture states that any successful pressing sequence can be transformed into any other successful pressing sequence with small alterations. Here we prove the conjecture for linear graphs, also known as paths. The connection to genome rearrangement and sorting signed permutations with reversals is also discussed.

1. Introduction

Sorting signed permutations by reversals (or inversions as biologists call it) is the first genome rearrangement model introduced in the scientific literature. The hypothesis that reversals change the order and orientation of genes — called genetic factors at the time — arose in [Sturtevant 1921] and was implicitly verified upon the discovery of chromosomes [Sturtevant and Novitski 1941]. At the same time, geneticists realized that "the mathematical properties of series of letters subjected to the operation of successive inversions do not appear to be worked out" [Sturtevant and Tan 1937]. In constructing phylogenies, maximum parsimony — supposing the

MSC2010: primary 05A05; secondary 05CXX.
Keywords: bioinformatics, sorting by reversals, pressing game, irreducible Markov chain.
This paper presents the results of the undergraduate research of E. Bixby and T. Flint in the 2012 Fall semester at the Budapest Semesters in Mathematics.

least evolutionary change as the most likely explanation — is a desirable characteristic. As such, the construction of minimum length sorting by reversals is both a biologically and mathematically interesting problem. This computational problem was rediscovered at the end of the 20th century, and its solution is known as the Hannenhalli–Pevzner theorem [1995; 1999].

The Hannenhalli–Pevzner theorem gives a polynomial running time algorithm that finds one such minimum length sorting sequence, that is, a series of reversals that transforms one signed permutation into another. However, there might be multiple solutions, and the number of solutions typically grows exponentially with the length of the permutation. Therefore, a(n almost) uniform sampler is required which gives a set of solutions from which statistical properties of the solutions can be calculated. The Markov chain Monte Carlo method (MCMC) is a typical approach to such sampling. MCMC starts with an arbitrary solution, and applies random perturbations on it, thus exploring the solution space. In the case of most parsimonious reversal sorting sequences, two distinct methods of perturbation have been considered:

(1) The first approach encodes the most parsimonious reversal sorting sequences with the intermediate permutations which appear as the result of the perturbations: $\pi_{\text{start}} = \pi_1$ is transformed into π_2, which is transformed into $\pi_3, \ldots,$ which is transformed into $\pi_n = \pi_{\text{end}}$. Then it cuts out a random interval from this sequence, $\pi_i, \pi_{i+i}, \ldots, \pi_j$ and gives a new, random sorting sequence between the permutations at the beginning and end of the window, namely, between π_i and π_j.

(2) The second approach encodes the scenarios with the series of mutations applied, and perturbs them in a sophisticated way, described in detail later in this paper.

As random perturbations are applied, the Markov chain randomly explores the solution space and will be at a random state after some number of steps. This random state is described by its distribution over the state space. A Markov chain is said to *converge* to a distribution ϕ if the distribution of its random state after some number of steps converges to ϕ as the number of steps tends to infinity.

A Markov chain for sampling purposes should fulfill two conditions: (a) it must converge to the uniform distribution, and as such must be irreducible, namely, from any solution the chain must be able to get to any another solution, and (b) the convergence must be fast.

Unfortunately, the first approach has been shown to be slowly mixing [Miklós et al. 2010]. This means that the necessary number of steps in the Markov chain to sufficiently approximate the uniform distribution grows exponentially with the length of the permutation. Therefore this approach is not applicable in practice.

Unfortunately, it is not known whether or not the second approach is irreducible, let alone whether or not it is rapidly mixing. In this paper, we take a step towards proving that this method is, in fact, irreducible.

This paper is organized in the following way. In Section 2, we define the problem of sorting by reversals, and the combinatorial tools necessary: the graph of desire and reality and the overlap graph. Then we introduce the pressing game on black-and-white graphs, and show that they correspond to the shortest reversal scenarios in a subset of permutations that typically appear in biology. We finish the section by stating the pressing game conjecture, a proof of which would imply the second method is irreducible. In Section 3, we prove the conjecture for linear graphs, also known as paths. The paper is finished with a discussion and conclusions.

2. Preliminaries

Definition. A *signed permutation* is a permutation of numbers from 1 to n, where each number has a $+$ or $-$ sign.

While the number of length n permutations is $n!$, the number of length n signed permutations is $2^n \times n!$.

Definition. A *reversal* takes any contiguous piece of a signed permutation and reverses both the order of the numbers and the sign of each number. It is also allowed that a reversal takes only a single number from the signed permutations; in that case, it changes the sign of this number.

For example, the following reversal flips the $-3\ +6\ -5\ +4\ +7$ segment:

$$+8\ -1\ -3\ +6\ -5\ +4\ +7\ -9\ +2 \quad \Rightarrow \quad +8\ -1\ -7\ -4\ +5\ -6\ +3\ -9\ +2.$$

The sorting by reversals problem asks for the minimum number of reversals necessary to transform a signed permutation into the identity permutation, i.e., the signed permutation $+1\ +2\ \cdots\ +n$. This number is called the *reversal distance*, and the reversal distance of a signed permutation π is denoted by $d_{\text{REV}}(\pi)$. To solve this problem, we have to introduce two discrete mathematical objects, the graph of desire and reality and the overlap graph. The graph of desire and reality is a drawn graph, meaning both edges and vertex locations affect the properties of the graph. The overlap graph is a graph in terms of standard graph theory.

The *graph of desire and reality* for a signed permutation can be constructed in the following way. Each signed number is replaced with two unsigned numbers; $+i$ becomes $2i-1, 2i$, and similarly, $-i$ becomes $2i, 2i-1$. The resulting length $2n$ permutation is framed between 0 and $2n+1$. Each number including 0 and $2n+1$ will represent one vertex in the graph of desire and reality. They are drawn in the same order along a line as they appear in the permutation.

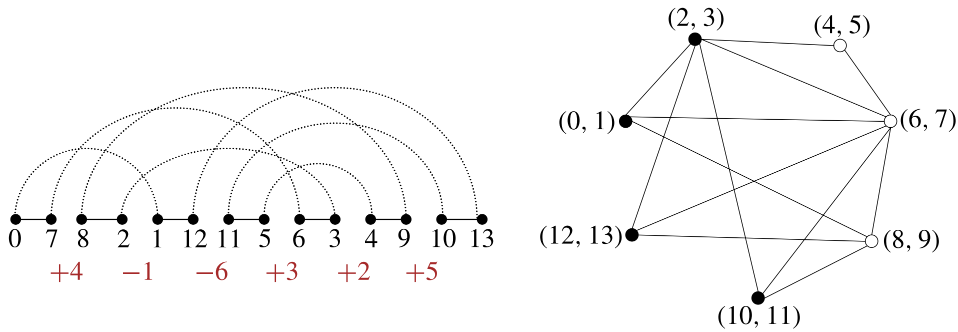


Figure 1. The graph of desire and reality and the overlap graph of the signed permutation $+4\ -1\ -6\ +3\ +2\ +5$.

We index the positions of the vertices starting with 1, and each pair of vertices in positions $2i-1$ and $2i$ are connected with an edge drawn as a straight line. We call these edges the *reality edges*. Each pair of vertices for numbers $2i$ and $2i+1$, $i = 0, 1, \ldots, n$ are connected with an edge drawn as an arc above the line of the vertices, and they are named the *desire edges*. The explanation for these names is that the reality edges describe what we see in the current permutation, and the desire edges describe the desired adjacencies in the final graph (the identity permutation): we would like 1 to be next to 0, 3 to be next to 2, etc.

Each desire edge is incident to two reality edges. We will call these edges the *legs* of the desire edge. A desire edge is called *oriented* if it spans an odd number of vertices. The rationale of this naming is that its legs point in the same direction; see, for example, the desire edge connecting 0 and 1 in Figure 1. A desire edge is called *unoriented* if it spans an even number of vertices and in this case, its legs indeed point in different directions; see, for example, the desire edge connecting 4 and 5 or the desire edge connecting 8 and 9 in Figure 1.

The *overlap graph* is constructed from the graph of desire and reality in the following way. The vertices of the overlap graph are the desire edges in the graph of desire and reality. The vertices are colored either black or white. A vertex in the overlap graph is black if it corresponds to an oriented desire edge. A vertex is white if it corresponds to an unoriented desire edge. Two vertices are adjacent if the intervals spanned by the corresponding desire edges overlap but neither contains the other. In Figure 1, we give an example for the graph of desire and reality and overlap graph.

The overlap graph might be disconnected. A component is called *oriented* if it contains at least one black vertex. If the component contains only white vertices, it is called *unoriented*. A component is *nontrivial* if it contains more than one vertex.

Any reversal changes the topology of the graph of desire and reality on two reality edges. Any desire edge is incident to two reality edges, and we say that the reversal *acts on* this desire edge if it changes the topology on the two incident reality edges.

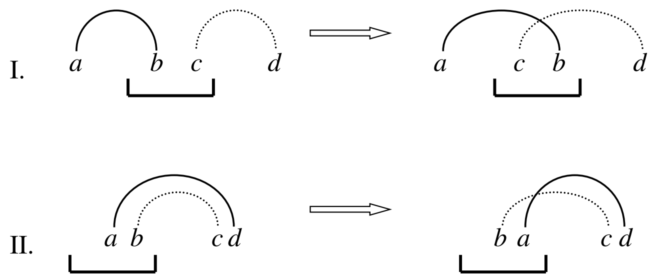


Figure 2. This picture shows how a reversal can change the overlap of two desire edges. The reversed fragment is indicated with a thick black line.

Any reversal in the underlying permutation also has the effect of reversing some segment of vertices in the graph of desire and reality. How do reversals acting on oriented desire edges change the graph of desire and reality and thus the overlap graph? We present a lemma below explaining this.

Lemma 1. *Fix a reversal, and let v be an oriented desire edge on which the reversal acts. Then the reversal*

(1) *changes whether any desire edge crossing v is oriented,*

(2) *changes whether any pair of desire edges crossing v overlaps, and*

(3) *causes the desire edge itself to become an unoriented edge without any overlapping edges (that is, neighbors in the overlap graph).*

Proof. (1) The reversal flips one of the legs of each overlapping desire edge. Therefore it changes the parity of the number of vertices below the desire edge and thus whether or not it is oriented.

(2) If two edges both overlap with v but not with each other because the intersection of their interval is empty, then the two edges must come from the two ends of v; see also Figure 2, case I. A reversal acting on v will change the order of one of the endpoints of their interval, so they will indeed overlap. If two edges overlap with v, but not with each other, since the interval of one of them contains the interval of the other, then they come from one end of v. It is easy to see that after the reversal they will overlap by definition; see Figure 2, case II. It is also easy to see that any overlapping pairs of edges which also overlap with each other are the two cases illustrated on the right-hand side of Figure 2, so after the reversal, they will not overlap.

(3) Finally, the oriented edge on which the reversal acts becomes an unoriented edge forming a small cycle with a reality edge, and thus it cannot overlap with any other desire edge. □

This lemma also shows the connection between sorting by reversals and the pressing game on black-and-white graphs: pressing a black vertex in an overlap graph is equivalent to reversing the corresponding desire edge. Below we define the pressing game on black-and-white graphs:

Definition. Given a graph $G(V, E)$ with its vertices colored with black and white, any black vertex v can be pressed, which has the following effect: (a) all neighbors of v change color, meaning that white neighbors become black and *vice versa*; (b) all pairs of neighbors of v change adjacency, meaning that adjacent pairs become nonadjacent and nonadjacent ones become adjacent; (c) finally, v becomes a separated white vertex. The aim of the game is to transform G into an all-white, empty graph.

If each component of G contains at least one black vertex, then the pressing game always has at least one solution, as it turns out, by the Hannenhalli–Pevzner theorem:

Theorem 2 [Hannenhalli and Pevzner 1999]. *Let π be a permutation whose overlap graph does not contain any nontrivial unoriented component. Then the reversal distance $d_{\text{REV}}(\pi)$, namely, the minimum number of reversals necessary to sort the permutation is*

$$d_{\text{REV}}(\pi) = n + 1 - c(\pi),$$

where n is the length of the permutation π and $c(\pi)$ is the number of cycles in the graph of desire and reality.

If the permutation π' contains a nontrivial unoriented component, then

$$d_{\text{REV}}(\pi') > n + 1 - c(\pi').$$

It is easy to see that any reversal can increase the number of cycles in the graph of desire and reality at most by 1, and the identity permutation contains $n+1$ cycles; hence the Hannenhalli–Pevzner theorem also says that if a permutation does not contain any nontrivial unoriented components, then any optimal reversal sorting sequence increases the number of cycles to $n+1$ without creating any nontrivial unoriented components. It is also true that these reversals can be chosen to act on oriented desire edges. Below we state this theorem.

Theorem 3. *Let π be a permutation which is not the identity permutation and whose overlap graph does not contain any nontrivial unoriented component. Then a reversal exists that acts on an oriented desire edge, increases $c(\pi)$ by 1, and does not create any nontrivial unoriented components.*

Furthermore, if G is an arbitrary black-and-white graph such that each component contains at least one black vertex, then at least one black vertex can be pressed without making a nontrivial unoriented component.

The proof can be found in [Bergeron 2001], and we skip it here. The proof considers only the overlap graph, and in fact, it indeed works for every black-and-white graph. A clear consequence is the following theorem.

Theorem 4. *Let G be a black-and-white graph such that each component contains at least one black vertex. Then G can be transformed into the all-white empty graph in the pressing game.*

Proof. It is sufficient to iteratively use Theorem 3. Indeed, according to Theorem 3, we can find a black vertex v such that pressing it does not create a nontrivial all-white component; on the other hand, v becomes a separated white vertex, and it will remain a separated white vertex afterward. Hence, the number of vertices in nontrivial components decreases at least by one, and in a finite number of steps, G is transformed into the all-white, empty graph. □

Consider the set of vertices as an alphabet; any sequence over this alphabet is called a *pressing sequence*. It is a *valid* pressing sequence when each vertex is black when it is pressed, and it is *successful* if it is valid and leads to the all-white, empty graph. The length of the pressing sequence is the number of vertices pressed in it. The following theorem is also true.

Theorem 5. *Let G be a black-and-white graph such that each component contains at least one black vertex. Then every successful pressing sequence of G has the same length.*

The proof can be found in [Hartman and Verbin 2006]. We are ready to state the pressing sequence conjecture.

Conjecture 6. *Let G be a black-and-white graph such that each component contains at least one black vertex. Construct a metagraph M whose vertices are the successful pressing sequences on G. Connect two vertices if the length of the longest common subsequence of the pressing sequences they represent is at least the common length of the pressing sequences minus* 4. *The conjecture is that M is connected.*

The conjecture means that with small alterations, we can transform any pressing sequence into any other pressing sequence, regardless of the underlying graph. By "small alteration" we mean that we remove at most four (not necessarily consecutive) vertices from a pressing sequence, and add at most four vertices, not necessarily to the same places where vertices were removed, and not necessarily to consecutive places.

It is important to note that there exist sorting sequences that are not pressing sequences. Specifically, these sequences contain two reversals which act on the same location in the permutation. These sequences also correspond to cycle-increasing reversals in the graph of desire and reality. However, the infinite site model [Ma et al. 2008] corresponds to permutations whose sorting sequences are exactly the pressing sequences, and restricting ourselves to this subset of permutations is a biologically reasonable assumption.

In this paper, we prove the pressing game conjecture for linear graphs. In addition, we can prove the metagraph will be already connected if we require that neighboring vertices have a longest common subsequence at least the common length of their pressing sequences minus 2.

3. Proof of the conjecture on linear graphs

The proof of our main theorem is recursive, and for this, we need the following notations. Let G be a black-and-white graph, and v a black vertex in it. Then Gv denotes the graph we get by pressing vertex v. Similarly, if P is a valid pressing sequence of G (namely, each vertex is black when we want to press it, but P does not necessarily yield the all-white, empty graph), then GP denotes the graph we get after pressing all vertices in P in the indicated order. Finally, let P^k denote the suffix of P starting in position $k+1$.

The convenience of linear graphs is their simple structure and furthermore, their self-reducibility:

Observation. Let G be a linear black-and-white graph and v a black vertex in it. Then Gv consists of a linear graph and the separated white vertex v.

Since any separated white vertex does not have to be pressed again, it is sufficient to consider $Gv \setminus \{v\}$, which is a linear graph. We are ready to state and prove our main theorem.

Theorem 7. *Let G be an arbitrary, finite, linear black-and-white graph, and let M be the following graph. The vertices of M are the successful pressing sequences on G, and two vertices are adjacent if the length of the longest common subsequence of the pressing sequences they represent is at least the common length of the pressing sequences minus 2. Then M is connected.*

Proof. It is sufficient to show that for any successful pressing sequences X and $Y = v_1 v_2 \cdots v_k$, there is a series $X_1, X_2, \ldots, X_m$ such that for any $i = 1, 2, \ldots, m-1$, the length of the longest common subsequence of X_i and X_{i+1} is at least the common length of the sequences minus 2, and X_m starts with v_1. Indeed, then both X_m and Y start with v_1, and both X_m^1 and Y^1 are successful pressing sequences on $Gv_1 \setminus \{v_1\}$. We can use induction to transform X_m into a pressing sequence which starts with v_2; then we consider its suffix, which is a successful pressing sequence on $Gv_1v_2 \setminus \{v_1, v_2\}$, etc.

Furthermore, it is sufficient to show that v_1 can be moved to some earlier position in some series of small alterations of the sequence, provided the intermediaries are also valid pressing sequences.

We first show that if v_1 is not in X, there exists some valid X' containing v_1, and X' differs from X by exactly one vertex. This is true for any arbitrary vertex in G and we state it in a separate lemma since we are going to use it again later.

Lemma 8. *Assume that X is a successful pressing sequence on G and that vertex v is not a separated vertex in G. Then either v is in X or there exists some valid X' containing v, and X' differs from X by exactly one vertex.*

Proof. Let $X = u_1u_2 \cdots u_k$. Assume that v is not in X. Vertex v has at least one neighbor in G and none in GX; therefore there exists at least one vertex in X which, when pressed, is adjacent to v. Consider the last such vertex, which is in position i, and call it u_i; by definition, none of the vertex pressings in X^i affect the adjacencies or color of v, so after pressing u_i, v must be a white disconnected vertex. It follows that in $Gu_1 \cdots u_{i-1}$, the vertices v and u_i have exactly the same neighbors, and as such $u_1 \cdots u_{i-1}vu_{i+1} \cdots u_k$ is a valid pressing sequence. □

We now assume that v_1 is part of the current pressing sequence, which we denote by $P_1w_1v_1P_2$, where both P_1 and P_2 might be empty.

Case 1. If w_1 and v_1 are not neighbors in GP_1, then $P_1v_1w_1P_2$ is also a valid pressing sequence, and one of the longest common subsequences of $P_1w_1v_1P_2$ and $P_1v_1w_1P_2$ is $P_1w_1P_2$, one vertex less than the original pressing sequences. In this way, we can move v_1 to a smaller index position in the pressing sequence, and this is what we want to prove.

Case 2. If w_1 and v_1 are neighbors in GP_1, then v_1 is white in GP_1, and then pressing w_1 makes it black again. However, v_1 is black in G, since it is the first vertex in the valid pressing sequence Y. As such there must exist at least one vertex in P_1 which was adjacent to a black v_1 when pressed. Let w_2 be the last such vertex in P_1, and let us denote $P_1 = P_{1a}w_2P_{1b}$.

We claim that none of the vertices in P_{1b} are neighbors of w_2 in GP_{1a}. Indeed if there were such a neighbor, call it w_3, after pressing w_2, w_3 would be adjacent to v_1. Note that w_3 cannot have already been adjacent to v_1 by linearity of GP_{1a}. As such, pressing w_3 would change the color of v_1, meaning either v_1 was black prior to pressing w_1 — a contradiction — or there were further vertices in P_{1b} which were adjacent to a black v_1 when pressed, another contradiction.

Since P_{1b} does not contain a vertex which is a neighbor of w_2 in GP_{1a}, we move w_2 next to w_1. The new pressing sequence $P_{1a}P_{1b}w_2w_1v_1P_2$ is still a valid and successful pressing sequence and the longest common subsequence of P and $P_{1a}P_{1b}w_2w_1v_1P_2$ is $P_{1a}P_{1b}w_1v_1P_2$, one vertex less than the common length of the sequences.

For sake of simplicity, we denote $P_{1a}P_{1b}$ by P_1' and now we can assume the pressing sequence is of the form $P_1'w_2w_1v_1P_2$, with P_1' and P_2 both potentially

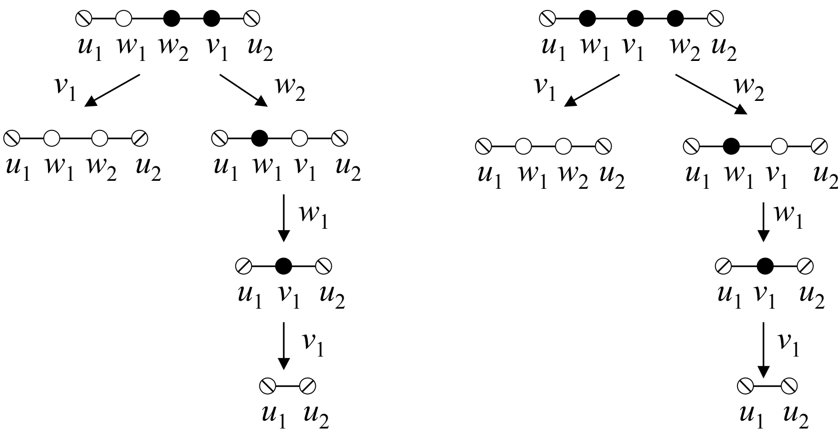


Figure 3. In the indicated two configurations, the neighbors of the $\{w_1, w_2, v_1\}$ triplet, u_1 and u_2, change color in the same way by pressing only v_1 and pressing $w_2 w_1 v_1$. The color change on u_1 and u_2 is indicated with the flipping of their crossing line.

empty. Since after pressing w_2, the vertices w_1 and v_1 become neighbors with w_1 being black and v_1 being white, the topology and colors of w_2, w_1 and v_1 in GP_1' is one of the following:

Case 2a. Assume that P_2 is not empty. The $\{w_1, w_2, v_1\}$ triplet has at least one neighbor (and at most two) in GP_1'; call them u_1 and u_2. Furthermore, either (1) one of u_1 and u_2 is pressed in P_2, or (2) we can replace some vertex in P_2 with u_1 or u_2 such that the resulting sequence is still valid, and successful on $GP_1' w_2 w_1 v_1$, due to Lemma 8. As such, we can assume that at least one neighbor of the $\{w_1, w_2, v_1\}$ triplet is pressed in P_2.

Without loss of generality, say u_1 is pressed before u_2 in P_2 and let $P_2 = P_{2a} u_1 P_{2b}$. Note that we can press v_1 instead of $w_2 w_1 v_1$, and the resulting sequence $GP_1' v_1 P_{2a}$ will be valid, as none of the vertices in P_{2a} are neighbors of w_2, w_1, or v_1. Next note from Figure 3 that the colors of u_1 and u_2 are identically altered in the pressing of either v_1 or $w_2 w_1 v_1$, and so we can press u_1. Figure 4 shows that the color of u_2 and a possible second neighbor of u_1 denoted by u_3 will be the same in $GP_1' w_2 w_1 v_1 P_{2a} u_1$ and $GP_1' v_1 P_{2a} u_1 w_1 w_2$. Therefore $P_1' v_1 P_{2a} u_1 w_1 w_2 P_{2b}$ will also be a successful pressing sequence on G, since no more vertices are affected by the given alteration of the pressing sequence. One of the longest common subsequences of $P_1' w_2 w_1 v_1 P_{2a} u_1 P_{2b}$ and $P_1' v_1 P_{2a} u_1 w_1 w_2 P_{2b}$ is $P_1' v_1 P_{2a} u_1 P_{2b}$, two vertices less than the entire pressing sequences. As intended, we have shown that v_1 is in a smaller index position of the pressing sequence.

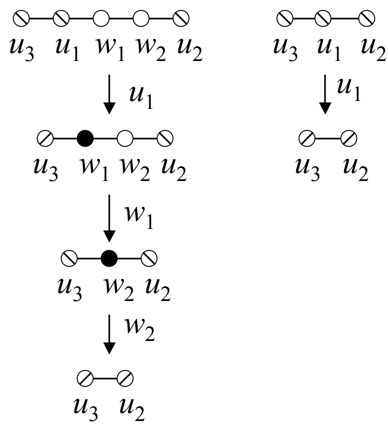


Figure 4. The color of u_2 and u_3 changes in the same way on the two indicated configurations.

Case 2b. Finally, assume that P_2 is empty. Then $GP_1'w_2w_1v_1$ is the all-white empty graph, and thus, $GP_1'w_2w_1$ contains the separated black v_1 and all separated white vertices, or contains a black v_1 connected to another black vertex and all separated and white vertices.

What follows is that GP_1' contains at most four nonisolated vertices, three of which are w_2, w_1, and v_1. Call the fourth u. If u exists, it must be black and adjacent to v_1 when v_1 is pressed. There are only four such cases, given the possible topologies for w_2, w_1, and v_1. If w_1 and w_2 are adjacent, then u is either black and adjacent to v_1 in GP_1' or it is adjacent to w_2 and is white. If w_2 and w_1 are not adjacent, then u can be adjacent to either w_2 or w_1, and must be white in both cases.

Note that all of these topologies can be described as follows; all neighbors of v_1 are black, v_1 is black, and all other vertices are white. This motivates the following lemma:

Lemma 9. *If GP is such that all neighbors of v_1 are black, v_1 is black, and all other vertices are white, and furthermore, there is a successful pressing sequence on G that starts with v_1, then there exists at least one vertex u in P such that when u is pressed u is not adjacent to v_1.*

Proof. Suppose instead that every vertex in P is adjacent to v_1 when pressed. P cannot be empty since then GP would be G and pressing v_1 in G would create an all-white nontrivial graph, contradicting that there exists a successful pressing sequence starting with pressing v_1. Furthermore, if all vertices in P are neighbors of v_1 when pressed, then P must contain an even number of vertices since v_1 is black both in G and GP.

Let $P = P_1' u_2 u_1$. In order for u_1 and u_2 to be adjacent to v_1 when pressed, and for GP to fit the given criteria, GP_1' must also have v_1 and all neighbors black, and all other vertices white. By repeated application, we see that G must also fit these criteria. By assumption then, there are no black vertices not adjacent to v_1, and as such, pressing v_1 results in an all-white nontrivial graph. However, this is a contradiction, as there exists a successful pressing sequence for G in which v_1 is pressed first. □

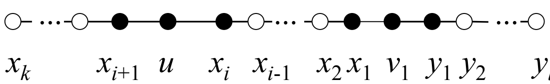
From the above lemma, we have that there exists some vertex in P_1' not adjacent to v_1 when pressed, and there are vertices which are adjacent to v_1 when pressed. For technical reasons, we have to separate them in the pressing sequence, which is doable due to the following lemma.

Lemma 10. *Let Pxu be a valid pressing sequence on G such that x is a neighbor of some v in GP and u is not a neighbor of v in GPx. Then Pux is a valid pressing sequence on G and $GPxu = GPux$.*

Proof. It is sufficient to show that x and u are not neighbors in GP. If x and u were neighbors, then the two neighbors of x would be u and v, causing u and v to become neighbors in GPx, a contradiction. □

Due to Lemma 10 it is possible to "bubble up" vertices that are not neighbors of v_1 in the pressing sequence so that the pressing sequence becomes $P_u P_n v_1$, where P_u contains the vertices that are not neighbors of v_1 when pressed and P_u contains the vertices that are neighbors of v_1 when pressed. Each bubbling-up step is allowed since the length of the longest common subsequence of two consecutive sorting sequences is their common length minus 1. We know that neither P_u nor P_n is empty due to Lemma 9 and due to the fact that w_1 and w_2 are in P_n.

Let u be the last vertex in P_u and let $P_u = P_u' u$. Without loss of generality, we can assume that u is on the left-hand side of v_1 in GP_u' and then GP_u' is



The vertices on the left-hand side of v_1 are denoted by $x_1, x_2, \ldots, x_k$ and we distinguish u amongst them. The vertices on the right-hand side of v_1 are denoted by $y_1, y_2, \ldots, y_l$.

Obviously, no x is a neighbor of any y when pressed, so we can bubble up the y vertices in P_n such that first the y vertices are pressed and then the x vertices. After a finite number of allowed alterations, $P_n = y_1 y_2 \cdots y_l x_1 x_2 \cdots x_k$.

Similarly to the previous cases, we can move down vertex u in the pressing sequence before x_i. We know that v_1 is black in $GP_u' u$ since it is black in G and neither of its neighbors is pressed in $P_u' u$. We are going to press some of the vertices amongst the x and y vertices provided that v_1 will be black after that

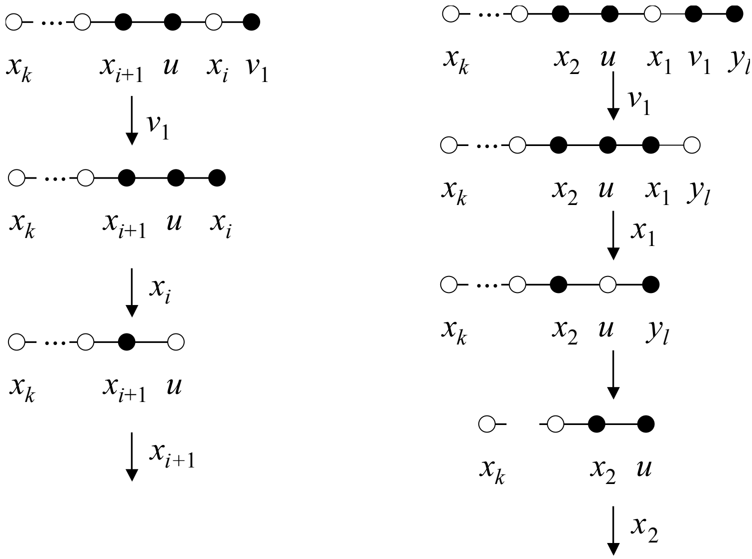
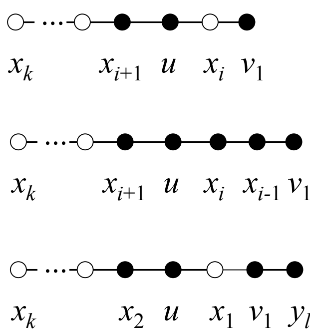


Figure 5. Alternative pressing sequences for two cases.

series of pressing. We consider the graph $GP'_u y_1 \cdots y_l x_1 \cdots x_{i-1}$ if v_1 is black in it (the runs of x vertices might be empty if $i = 1$), and otherwise the graph $GP'_u y_1 \cdots y_l x_1 \cdots x_{i-2}$ (also the runs of x vertices might be empty if $i = 2$) or $GP'_u y_1 \cdots y_{l-1}$ if $i = 1$ and the number of y vertices is odd (if $i = 1$ and the number of y vertices is even, then v_1 will be black in $GP'_u y_1 \cdots y_l$). We have one of the following graphs



on which $ux_i \cdots x_k v_1$, $ux_{i-1} \cdots x_k v_1$, $y_l u x_1 \cdots x_k v_1$ is the current successful pressing sequence, respectively.

A successful pressing sequence replacing $ux_i \cdots x_k v_1$ is $v_1 x_i \cdots x_k u$, as can be seen on the left-hand side of Figure 5. The length of the longest common subsequence of the two pressing sequences is 2 less than their common length, as required. The pressing sequence $y_l u x_1 \cdots x_k v_1$ can be replaced by $u x_1 y_l x_2 \cdots x_k v_1$ since y_l is a neighbor of neither u nor x_1. Then this pressing sequence can be

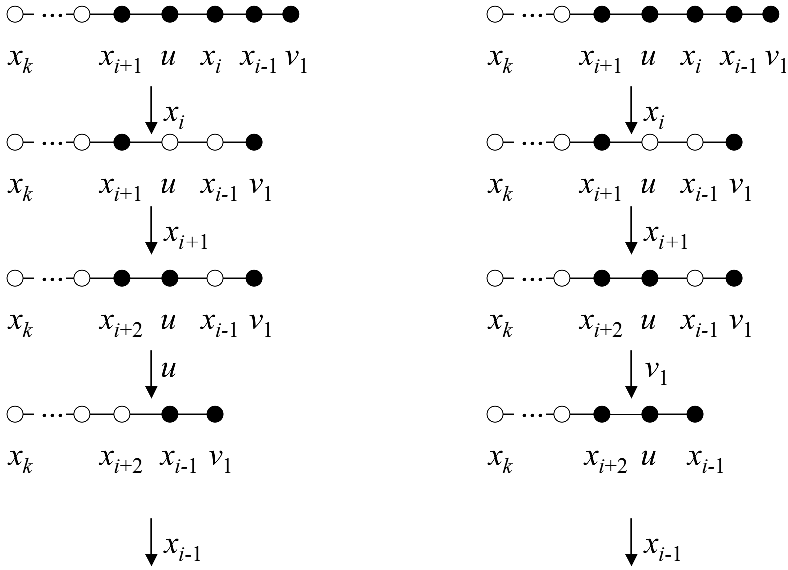


Figure 6. Changing the pressing sequence $ux_{i-1}\cdots x_k v_1$ in two steps such that v_1 is in a smaller index position.

replaced by $v_1 x_1 y_l x_2 \cdots x_k u$, as can be seen on the right-hand side of Figure 5. The length of the longest common subsequence of $ux_1 y_l x_2 \cdots x_k v_1$ and $v_1 x_1 y_l x_2 \cdots x_k u$ is again 2 less than their common length.

Finally, the pressing sequence $ux_{i-1}\cdots x_k v_1$ can be replaced in two steps; first it is changed to $x_i x_{i+1} u x_{i-1} x_{i+2} \cdots x_k v_1$, then to $x_i x_{i+1} v_1 x_{i-1} x_{i+2} \cdots x_k u$, as can be checked in Figure 6. In both steps, the length of the longest common subsequences of two consecutive pressing sequences is 2 less than their common length as required.

We proved that in any case, v_1 can be moved into a smaller index position with a finite series of allowed perturbations. Iterating this, we can move v_1 to the first position. Then we can do the same thing with v_2 on the graph $Gv_1 \setminus \{v_1\}$, and eventually transform X into Y with allowed perturbations. □

4. Discussion and conclusions

In this paper, we proved the pressing game conjecture for linear graphs. Although the linear graphs are very simple, this proof technique provides a direction for proving the general case. Indeed, it is generally true that if a vertex v is not in a successful pressing sequence P, then a successful pressing sequence P' exists which contains v and the length of the longest common subsequence of P and P' is only 1 less than their common length. Case 1 in the proof of Theorem 7 holds for arbitrary graphs, and in a working manuscript, we were able to prove that the conjecture is true for Case 2a using linear algebraic techniques similar to that used in [Hartman and Verbin 2006]. The only missing part is Case 2b, which seems to be very complicated for general graphs; for example, Lemma 10 cannot be generalized for arbitrary graphs.

A stronger theorem holds for the linear case than is conjectured for the general case. One possible direction above proving the general conjecture is to study the emerging Markov chain on the solution space of the pressing game on linear graphs. We proved that a Markov chain that randomly removes two vertices from the current pressing sequence, adds two random vertices to it, and accepts it if the result is a successful pressing sequence is irreducible. It is easy to set the jumping probabilities of the Markov chain such that it converges to the uniform distribution of the solutions. The remaining question is the speed at which this Markov chain converges.

Acknowledgements

István Miklós was supported by OTKA grant PD84297. Alexey Medvedev is thanked for fruitful discussions.

References

[Bergeron 2001] A. Bergeron, "A very elementary presentation of the Hannenhalli–Pevzner theory", pp. 106–117 in *Combinatorial pattern matching* (Jerusalem, 2001), edited by A. Amir and G. M. Landau, Lecture Notes in Comput. Sci. **2089**, Springer, Berlin, 2001. MR 1904571 Zbl 0990.68050

[Hannenhalli and Pevzner 1995] S. Hannenhalli and P. A. Pevzner, "Transforming men into mice (polynomial algorithm for genomic distance problem)", pp. 581–592 in *36th Annual Symposium on Foundations of Computer Science* (Milwaukee, WI, 1995), IEEE Comput. Soc. Press, Los Alamitos, CA, 1995. MR 1619106 Zbl 0938.68939

[Hannenhalli and Pevzner 1999] S. Hannenhalli and P. A. Pevzner, "Transforming cabbage into turnip: polynomial algorithm for sorting signed permutations by reversals", *J. ACM* **46**:1 (1999), 1–27. MR 2000j:92013 Zbl 1064.92510

[Hartman and Verbin 2006] T. Hartman and E. Verbin, "Matrix tightness: a linear-algebraic framework for sorting by transpositions", pp. 279–290 in *String processing and information retrieval* (Glasgow, 2006), edited by F. Crestani et al., Lecture Notes in Comput. Sci. **4209**, Springer, Berlin, 2006. MR 2337809

[Ma et al. 2008] J. Ma, A. Ratan, B. J. Raney, B. B. Suh, W. Miller, and D. Haussler, "The infinite sites model of genome evolution", *Proc. Nat. Acad. Sci. USA* **105**:38 (2008), 14254–14261.

[Miklós et al. 2010] I. Miklós, B. Mélykúti, and K. Swenson, "The metropolized partial importance sampling MCMC mixes slowly on minimum reversal rearrangement paths", *IEEE/ACM Trans. Comput. Biol. Bioinform.* **4**:7 (2010), 763–767.

[Sturtevant 1921] A. H. Sturtevant, "A case of rearrangement of genes in Drosophila", *Proc. Nat. Acad. Sci. USA* **7**:8 (1921), 235–237.

[Sturtevant and Novitski 1941] A. H. Sturtevant and E. Novitski, "The homologies of chromosome elements in the genus Drosophila", *Genet.* **26** (1941), 517–541.

[Sturtevant and Tan 1937] A. H. Sturtevant and C. C. Tan, "The comparative genetics of Drosophila pseudoobscura and D. melanogaster", *J. Genet.* **34** (1937), 415–432.

Received: 2013-04-08 Revised: 2015-01-19 Accepted: 2015-01-21

eli.bixby@gmail.com
Budapest Semesters in Mathematics, H-1071 Budapest, Bethlen Gábor tér 2, Hungary

tobycollege@gmail.com
Budapest Semesters in Mathematics, H-1071 Budapest, Bethlen Gábor tér 2, Hungary

miklosi@renyi.hu
Rényi Institute, H-1053 Budapest, Reáltanoda utca 13-15, Hungary

Budapest Semesters in Mathematics, H-1071 Budapest, Bethlen Gábor tér 2, Hungary

involve

msp.org/involve

MANAGING EDITOR

Kenneth S. Berenhaut, Wake Forest University, USA, berenhks@wfu.edu

PRODUCTION

Silvio Levy, Scientific Editor

Cover: Alex Scorpan

See inside back cover or msp.org/involve for submission instructions. The subscription price for 2016 is US $160/year for the electronic version, and $215/year (+$35, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues from the last three years and changes of subscribers address should be sent to MSP.

Involve (ISSN 1944-4184 electronic, 1944-4176 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

Involve peer review and production are managed by EditFlow® from Mathematical Sciences Publishers.

PUBLISHED BY

mathematical sciences publishers

nonprofit scientific publishing

http://msp.org/

involve

2016 vol. 9 no. 1