Reconfiguration with Line Tracking Motions

by

William J. Lenhart
Department of Computer Science, Williams College
Williamstown, Massachusetts USA
lenhart@cs.williams.edu

and

Sue H. Whitesides
School of Computer Science, McGill University
3480 University St., Montréal H3A 2A7 CANADA
sue@cs.mcgill.ca

abstract: We consider the problem of moving a closed chain of n links confined to the plane from one given configuration to another. The links have fixed lengths and may rotate about their endpoints, possibly crossing over one another. We define the notion of "line tracking motion". Such a motion can be easily calculated and described by computer so that the locations of all joints during the motion are implicitly specified. We show by giving an algorithm that when reconfiguration is possible by any means, then a sequence of $O(n)$ of our simple line tracking motions can be used to carry out the reconfiguration. These motions can be computed in $O(n)$ time on real RAM.

1. Introduction

While there are general techniques [SS] for solving motion planning problems having a bounded number of degrees of freedom in polynomial time, problems having an unbounded number of degrees of freedom are often at least NP-complete. Hence it is of interest to find examples of motion planning problems that can be solved quickly despite having an unbounded number of degrees of freedom. The problem of reconfiguring chains of links under various conditions has been considered in this regard in [HJW], [KK], [K] and [LW], and surveyed in [W]. In particular, these papers describe polynomial time algorithms for motion planning problems that have an unbounded number of degrees of freedom.

A simple criterion for the reconfigurability of a closed chain of n links confined to the plane, with links allowed to cross, is given in [LW]. In particular, this criterion, described in more detail later, says that a closed chain can be moved between any given pair of configurations if the lengths of its second and third longest links sum to at most half the sum of all the link lengths. In this paper, we give an algorithm that reconfigures closed chains of links provided that the reconfiguration is possible by any arbitrary means. Our algorithm requires $O(n)$ simple "line tracking" motions for an n-link chain, and the descriptions of these motions can be calculated in $O(n)$ time. Here, a key element is our definition of a "line tracking" motion, which makes possible a fast reconfiguration algorithm.

A *chain* is a sequence of n *links*, $L_1, \cdots, L_n$, connected by *joints*. L_i has joints v_{i-1} and v_i and length l_i. Each link can rotate freely about its joints. A *configuration* of a chain $L_1, \cdots, L_n$ is a polygonal curve (possibly self-intersecting) that ~~consists of n~~ consecutive links of lengths $l_1, \cdots, l_n$, respectively. A *closed chain* is a chain such that v_0 and v_n are the same joint. Hence a configuration of a closed chain is just a closed polygonal curve. We use the term *linkage*, often denoted L, to refer to a closed chain or a piece of a closed chain.

Definition. Two configurations of a closed chain L are *equivalent* if one configuration can be continuously moved to the other (links may cross, but L must remain in the plane). A configuration of L is *invertible* if it is equivalent to its mirror image (with respect to some arbitrary line).

In [LW], it is shown that with respect to the above notion of equivalence, a closed chain of links in the plane has at most two equivalence classes of configurations. If the chain satisfies the property that the lengths of the second and third longest links sum to at most half the sum of all the link lengths, then the chain has just one equivalence class of configurations: it is possible to move between any given pair of configurations. If this property does not hold (see Figure 1), then the chain has exactly two equivalence classes of configurations, and the configurations in the one class are the mirror images of the configurations in the other class. In this case, it is possible to move between two given configurations if and only if they lie in the same one of the two possible equivalence classes.

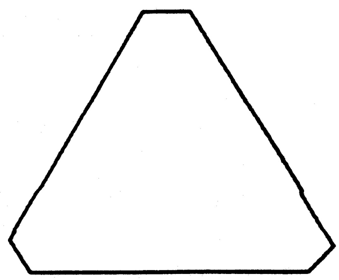

Figure 1. A 6-link chain that is not invertible.

In Section 2, we discuss the concept of "simple motions", and a define the notion of a "line tracking" motion that makes possible an algorithm that reconfigures chains of links when such reconfiguration is possible by any means. The reconfiguration algorithm is presented in Section 3. It takes $O(n)$ time to compute and produces $O(n)$ simple line tracking motions.

2. Simple Motions and Line Tracking Motions

It is essential for a motion planning algorithm to be able to describe motions unambiguously. To achieve this, it is useful to define one or more kinds of basic or simple motion steps, so that complicated motions can be described as a sequence of the simple ones. Of course, the simple motions chosen should not be limiting: it should be possible to carry out any reconfiguration in terms of the simple motions available to the algorithm. Based on [HJW], here is a list of criteria for a good "simple motions":

1) The description of the motion should uniquely determine the geometric movement of all parts of the linkage.
2) The motion should be one that can be computed and described by an algorithm.
3) If the angle at a joint changes, it should change monotonically. In other words, a motion in which a given angle increases and then decreases should be regarded as a combination of simpler motions. This is desirable so that the algorithmic notion of simple motion has some bearing on a mechanical notion of simplicity that would take into account wear and tear on joints, say.

It should be noted that any motion other than translation or rotation of the entire linkage (or a combination of the two) must involve changing some joint angles. It is impossible to change the relative positions of any joints in a closed chain without altering at least four of the joint angles simultaneously.

The criteria given above allow for many kinds of motions. We next define a motion (called *simple elbow bending*) that satisfies the above criteria and that is an essential ingredient of our definition of line tracking motion. The elbow bending motion applies to an open chain of links.

Definition: An *elbow* consists of an open chain of links $v_0,...,v_k,...,v_n$, where the location in the plane of v_0 is kept fixed, and the links between v_0 and v_k are stretched out in a straight chain, and the links from v_k to v_n are also stretched out in a straight chain. The elbow joint v_k may flex, and the entire elbow linkage may rotate about the fixed joint v_0. The *elbow bending* motion consists of moving the elbow so that v_n moves in a straight line directly from its initial location to a specified final location.

Observation 1: It is easy to show that an elbow can be moved so that v_n tracks a straight line from its initial location to any given point within its reach.

Observation 2: There are two configurations of an elbow that place v_n at a given point in the interior of its reachability region. However, note that an elbow motion can be specified completely by giving 1) a reachable closed straight line segment for v_n to track and 2) the initial configuration of the arm -- provided the line segment lies in the interior of the reachability region and does not pass through the location of v_0; if v_n reaches either v_0 or the boundary of its region, then additional information can be given to specify on which side of the line through v_0 and v_n the elbow joint v_k should move. Hence elbow motions easily satisfy criteria 1) and 2) above.

Observation 3: In the process of moving v_n along a straight line segment within its reachability region, the elbow joint at v_k may sometimes be required to close (though not necessarily fold completely), then open (though not necessarily straighten completely). Also, the first link L_1 may be required to rotate first in one sense, then in the opposite sense (so the angle formed at v_0 between L_1 and, say, a horizontal reference line both opens and closes).

Definition: A *simple elbow motion* is an elbow motion in which the joint v_0 and the elbow joint v_k each either open or close monotonically.

Clearly every elbow motion can be decomposed into at most a constant number of simple elbow motions.

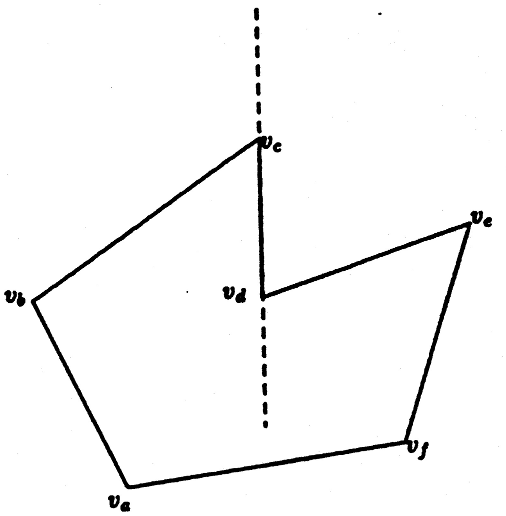

Figure 2. Line tracking motion.
The portions of chain between the labeled joints are drawn as straight lines to indicate that the shape of these portions does not change during the motion. The portion of the chain between v_c and v_d slides along the dotted line. Joints v_a and v_f act as fixed joints for elbows with elbow joints at v_b and v_e and free ends at v_c and v_d, respectively.

Next, we define the simple motion of a closed chain L that is essential to the reconfiguration algorithm. It is a combination of simple elbow motions that will cause at most six joint angles of L to change simultaneously. The idea is that most of the linkage should remain fixed while a particular piece of it slides along some specified line. In order to create the flexibility needed for the sliding, part of the linkage behaves as an elbow whose free end is attached to the initial joint of the sliding piece, and another part acts as an elbow whose free end is attached to the final joint of the sliding piece. The links connecting the fixed joints of the two elbows do not move.

Definition: (See Figure 2.) Let $L = v_0, \cdots, v_n$ be a closed chain, and let v_a, v_b, v_c, v_d, v_e and v_f be six (not necessarily distinct or consecutive) joints of L as they would appear in cyclic order around the chain. These are the only joints whose angles will be allowed to change, as each of the (at most) six link-disjoint chains determined by these joints will each remain rigid during the motion. Let M be a line containing both v_c and v_d. A *line tracking* motion is defined by the following.

1) Both v_c and v_d are to move along M while the shape of the subchain from v_c to v_d remains unchanged.
2) The locations of joints v_a and v_f remain fixed throughout the motion, and none of the portion of the chain from v_f to v_a moves.
3) The portion of the chain between v_a and v_c acts as an elbow with v_a as the fixed joint, v_b as the flexing elbow joint, and v_c as the free end joint. The subchain from v_a to v_b and the subchain from v_b to v_c are not necessarily straight chains of links, but they behave like straight chains of links as their shapes do not change internally during the motion.
4) Similarly, the portion of the chain between v_d and v_f acts as an elbow with v_f as the fixed joint, v_e as the flexing elbow joint, and v_d as the free end joint.

Clearly any line tracking motion can be decomposed into at most a constant number (independent of n) *simple* line tracking motions in which joints change monotonically only. Specifying the initial configuration, the line M, and the stopping position for the joint v_d can be regarded as specifying the motion completely. (From this, specifications for the constituent simple motions can be computed.)

3. The Algorithm

In this section, we assume that we are given two configurations of a closed chain that are in the same equivalence class, so it *is* possible to move from one to the other. Our goal is to show how this may be done using $O(n)$ simple line tracking motions as defined in Section 2. The first step is to show how to move any initial configuration of a closed chain to a triangular configuration with $O(n)$ of these simple line tracking motions.

Theorem 1. Any configuration of a closed chain of n links can be moved to a triangular configuration using $O(n)$ simple line tracking motions, which can be computed in $O(n)$ time.

proof. We use a special case of line tracking in which $v_c = v_d$, so the portion of the chain from v_c to v_d is empty. Furthermore, $v_a, v_b, v_c{=}v_d, v_e$ and v_f will be consecutive joints. The line M along which the joint $v_c{=}v_d$ moves will be the line containing v_c that is perpendicular to the line through v_a and v_f (see Figure 3 below).

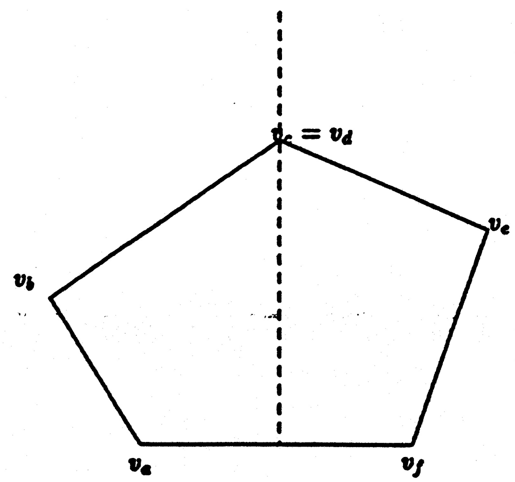

Figure 3. A special line tracking motion with $v_c = v_d$.

The algorithm for moving a closed chain to a triangular configuration using this specialisation of line tracking motions is as follows. Take any five consecutive joints, say v_1 through v_5 (this step may be skipped if the initial configuration has only four sides); now, using the motion described above, begin to move v_3 *away* from the line through v_1 and v_5 along the line through v_3 perpendicular to the line through v_1 and v_5. This guarantees that both v_2 and v_4 are opening because v_3 is moving *away* from both v_1 and v_5. Hence the motion can continue until one or both of v_2, v_4 straighten. This may require more than one (but at most a constant number) of simple line tracking motions because the angles at joints v_1 and v_5 may first increase, then decrease. The joint (or joints) that becomes straight is then held rigid, and the process is repeated until only three or four joints remain unextended.

Once the configuration becomes a quadrilateral, we apply one more line tracking motion, with the joints v_a and v_f identified, the joints v_c andv_d identified, and with v_c=v_d moving away from v_a=v_f. This moves the original configuration into a triangle using only $O(n)$ simple line tracking motions that are easy to describe and compute. □

To convert one configuration of a closed chain to another, a natural way to proceed is to move both the initial configuration and the desired final configuration to triangular forms. If these forms were congruent, then we could move the initial form to the final form via the triangular form as an intermediary: we would just move the initial configuration to triangular form, then undo the motions that take the final form to the triangular form. (Here we are concerned with obtaining a correctly shaped and correctly oriented configuration, as such a configuration can be moved by a single translation and a single rotation to its desired final location.) However, it need not be the case that applying the algorithm in the proof of Theorem 1 to both the initial and final configurations will result in a congruent, correctly oriented triangle. There are two problems. Firstly, the two triangles could have different joints functioning as vertices of the triangle. Secondly, the two triangles could differ in orientation (clockwise versus counterclockwise as the vertices are read in order of increasing label). The next lemma handles the first problem.

Lemma 1. Given two triangular configurations A and B of the same closed n-link chain, it is possible to move A to a triangular configuration that has the same vertices as B using at most a constant (independent of n) line tracking motions. The resulting triangle either looks like B or a mirror image of B.

proof. The straightforward proof goes by case analysis of the location of the vertex joints of A in the chains making up the sides triangle B.

Now let us address the second problem: suppose that the initial and final configurations of the original closed chain are moved into triangles whose shapes are mirror images of one another. From [LW], we know that if the chain is not invertible, then it is impossible to move between the two given configurations by any means. Hence let us assume that the chain is invertible (i.e., that the lengths of its second and third longest links sum to at most one half of the sum of all the link lengths). The only remaining step in carrying out the scheme outlined above Lemma 1 to move an invertible chain from triangular configuration A to a mirror image triangular configuration B. To do this, we observe the following.

Observation 2. If an invertible chain has a joint v_j that cannot be straightened because $l_j + l_{j+1}$ is greater than the sum of the remaining link lengths, then it must be possible to fold joint v_j. This is because any transformation that changes the orientation of a chain must make each triple of joints collinear at some time during the transformation. If v_{j-1}, v_j and v_{j+1} cannot be made collinear by straightening the joint at v_j, then v_j must be capable of folding. In particular, this means that the rest of the chain must be flexible enough to allow this to happen. Consequently, the chain section from v_{j+1} to v_{j-1} cannot contain a link that is too long – i.e., a link whose length, minus the sum the the remaining link lengths in that section of chain, is greater than the absolute value of the difference between l_{j+1} and l_j. Such a situation in that section of chain would prevent v_j from folding, contradicting the invertibility of the entire chain.

Theorem 2. Given an invertible n-link chain in the shape of a triangle A, the chain can be moved with a number of simple line tracking motions bounded by a constant independent of n to a configuration B whose shape is the mirror image of A. The time required to compute the motions is in $O(n)$.

proof. Let the joints at the vertices of A be v_0, v_i and v_k. Without loss of generality, suppose that the joints have been labeled so that the longest link of the chain occurs in the side of A between v_0 and v_i.

If v_k is not v_{i+1}, then in the side joining v_i to v_k, find that joint v_j such that the length of the chain from v_0 to v_j is at most half the perimeter, but the chain from v_0 to v_{j+1} has length greater than half the perimeter. If v_j is not v_i, hold v_0 fixed and move v_j away from v_0 using a line tracking motion until v_i straightens. We now have a triangle whose vertices are v_0, v_j and v_k, with the longest link in the chain occurring in the side containing v_0 and v_j.

Now freeze all the joints in the side containing v_0 and v_j. The resulting closed chain is still invertible because the sum of the second and third longest links in the new chain (this sum remained the same or decreased) is at most half the sum of all the link lengths (this quantity did not change). However, joint v_j cannot be straightened in the new chain. Hence by Observation 2, it must be possible to fold v_j.

If v_{j+1} is not v_k, hold v_0 fixed and move v_{j+1} away from v_0 until v_k straightens. Now we have a triangle whose vertices are v_0, v_j and v_{j+1}, and the chain section from v_{j+1} to v_0 has no links that are too long to prevent v_j from folding.

In the section of chain from v_{j+1} to v_0, locate joints v_r and v_{r+1} such that the length of chain from v_{j+1} to v_r is at most half the length of chain from v_{j+1} to v_0, but the length of chain from v_{j+1} to v_{r+1} is more than half the length of chain from v_{j+1} to v_0. (Possibly $v_r = v_{j+1}$ or $v_{r+1} = v_0$, but there is at least one joint interior to the section of chain from v_{j+1} to v_0.)

Now suppose v_r is not v_{j+1}; otherwise skip the following step. Hold v_0 fixed and use a line tracking motion to move v_{j+1} toward v_0, bending at v_j and v_r, until v_j or v_r folds.

Suppose that v_r folds or that $v_{j+1} = v_r$. (If v_j folds first, then the following step can be skipped.) Joints v_{j+1}, v_r, v_{r+1} and v_0 are collinear. Using a line tracking motion, move the section of chain between v_{j+1} and v_r on the line through it and v_0 toward v_0, bending at v_j and v_{r+1}, until v_j folds. (Note that v_{r+1} cannot fold first, or v_j would not be able to fold at all, contradicting the invertibility of the new chain.)

At this point, we have v_0, v_j, v_{j+1} and v_r collinear with v_j folded. Undo the previous motions to return v_{j+1} to its original position, but move v_j to the opposite side of the line of motion through v_0. This results in a triangle whose shape is the mirror image of the original one.

The number of simple line tracking motions is bounded by a constant independent of n, and it is easy to see that the amount of time needed to compute these motions is in $O(n)$. □

References

[HJW] Hopcroft, J, Joseph, D. and Whitesides, S., "On the movement of robot arms in 2-dimensional bounded regions", *SIAM J. Computation*, v. 14, no. 2, 1985, pp. 315-333.

[KK] Kantabutra, K. and Kosaraju, S. R., "New algorithms for multilink robot arms", *J. Computer and System Sci.*, v. 32, 1986, pp. 136-153.

[K] Kantabutra, V., "Motions of a short-linked robot in a square", *Discrete and Computational Geometry*, accepted for publication.

[LW] Lenhart, W. and Whitesides, S., "Turning a polygon inside out," Proceedings of the 3rd Canadian Conference on Computation Geometry, Vancouver, August 10-12, 1991, pp. 66-69.

[SS] Schwartz, J. T. and Sharir, M., "On the piano mover's problem II. General techniques for computing topological properties of real algebraic manifolds", *Advances in Appl. Math* v. 4, 1983, pp. 298-351.

[W] Whitesides, S., "Algorithmic issues in the geometry of planar linkage movement," *The Australian Computer Journal*, Special Issue on Algorithms, May 1992.