Title	Research on Simulation of Mechanism by Linkage
Author(s)	Feng, Tianfeng
Citation	
Issue Date	2019-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/15931
Rights	
Description	Supervisor: 上原 隆平, 先端科学技術研究科, 修士(情報科学)

Abstract

A linkage is a collection of fixed-length 1D segments joined at their endpoints to form a graph. A segment endpoint is also called a vertex. The segments are often called links or bars, and the shared endpoints are called joints or vertices. The bars correspond to graph edges and the joints to graph nodes. The linkage can be distinguished according to their graph structure. The structure may be a general graph, a tree, a single cycle, or a simple path. A linkage whose graph is a path is called a chain, an arc, or a robot arm. In this research, we only consider such a simple linkage whose graph is a path.

The motivation for this research stems from the application of linkage. It is sometimes useful to view an articulated robotic manipulator as a chain of links. The study of linkage, and more generally, mechanisms, has long been important to engineering. The kinematics of mechanisms is often of central concern in practical applications. Therefore, the simulation problem by a linkage is considered in this research for a robot arm that modeled by a linkage P and a general mechanism modeled by a graph G. The mission is to simulate the target graph G by the given linkage P. Each vertex of the graph G should be simulated by some vertices of the linkage P, and each edge of the graph G should be simulated by a subpath of the linkage P. This problem can be formalized as finding a mapping from the path P to a path on the graph G. The decision problem asks if there is an Eulerian path of G spanned by P. To solve this problem, the weighted Eulerian path problem and its variants are investigated.

At first, we try to solve the weighted Eulerian path problem. This problem can be seen as the Eulerian path problem with edge weights for a given path P and a graph G with length function. It asks us to determine if there is an Eulerian path of G spanned by P with length consistency. This problem is linear time solvable if the path P and the graph G consist of unit length edges. However, the first interesting result is that the weighted Eulerian path problem is strongly NP-hard even if edge lengths are quite restricted. Precisely, the problem is strongly NP-hard even if P and G consist of edges of lengths only 1 and 2. We reduce the 3-Partition problem to the weighted Eulerian path problem to show the NP-hardness of this problem. The 3-Partition problem is a well-known NP-complete problem. Therefore, the weighted Eulerian path problem is tackled in two different ways, and two different simulation problems are considered: the elastic linkage problem and the traverse problem of a tree by a path.

The first problem is the elastic linkage problem, and this is an optimization version of the weighted Eulerian path problem. In weighted Eulerian path problem, the path P can only cover an edge of the graph G once, and

the edge lengths of the path P are all fixed, and they cannot be changed. However, in this variant, we change the second condition. All edges in P are allowed to be elastic to simulate the target graph G by the path P, that is, we can stretch or shrink the edges in the elastic linkage P. This situation is natural not only in the context of the robot arm simulation but also in the approximation algorithm. So far, we only consider the elastic linkage problem for two paths P and G, and we use the elastic linkage P to simulate the target path G. Firstly, the elastic ratio of edges and mappings in the simulation process are defined. The elastic ratio of edges is always greater than or equal to 1, that is, the elastic ratio is the change factor of the edge in the path P, and the elastic ratio of the mapping is defined by the maximum elastic ratio of all edges in the path P. The objective of the elastic linkage problem is to minimize the elastic ratio of the mapping from the path P to the path G for given P and G. We start from a simple case which the path G consists of only one edge. In this case, we prove that the minimum elastic ratio is achieved when all ratios of edges in path P take the same value, after that, we prove that the elastic linkage problem can be solved in polynomial time by dynamic programming.

The second problem is the traverse problem of a tree by a path. In weighted Eulerian path problem, P can only cover an edge of G once, and the edge lengths of the path P are all fixed. In this variant, we change the first condition. The path P is allowed to cover an edge of the graph G twice or more, but we do not allow edges in the path P to be changed. In this situation, the path P can simulate the graph G even if G does not have an Eulerian path. In this case, we do not allow the path P to be elastic, or its ratio is fixed to 1. Firstly, the general simulation problem is proved to be weakly NP-complete even if G is an edge. It is similar to the ruler folding problem, which is weakly NP-complete. We can reduce the ruler folding problem to the general simulation problem by just letting G be an edge of length L. Thus, we consider more restricted cases. From the viewpoint of graph theory, it is natural to consider the case that G is a tree. For a given tree G and a path P with edge lengths, the traverse problem asks if G has a trail by P such that each edge of G is traversed exactly twice. When G is a tree, the problem is in a simple form, and P simulates G by traversing each edge twice in the unique spanning tree of G or G itself. However, this problem is still strongly NP-complete even in quite restricted cases; (1) G is a star, and P consists of edges of only two different lengths, and (2) G is a spider, and all edges in G and P are of length p and q, where p and q are any two positive integers that are relatively prime, or $p = 1$ and $q = 2$. We also reduce the 3-Partition problem to the traverse problem of a tree by a path to show the NP-hardness of these restricted cases. On the other hand, when G

is a star and its edge lengths are of k different values, we prove that the tree traverse problem can be solved in polynomial time by dynamic programming when k is constant.