Title	ヒッチコック型輸送問題に対する途中棄却を含む段階的解法
Author(s)	池田，隆之
Citation	
Issue Date	2003-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1664
Rights	
Description	Supervisor:浅野 哲夫，情報科学研究科，修士

Stepwise Algorithm with Abort for Hitchcock Type Transportation Problem

Takayuki Ikeda (110007)

School of Information Science,
Japan Advanced Institute of Science and Technology

February 14, 2003

Keywords: network flow, Hitchcock, transportation problem.

In this paper, we deal with the Hitchcock type transportation problem. First, we describe what the Hitchcock type transportation problem is. Given a network, the problem is to find flow of some previously fixed quantity with the least cost. We call it a minimum cost flow problem. Hitchcock type transportation problem is a special case of the problem. Namely, a given graph is bipartite and it has only edges from one side to the other. Hitchcock type transportation problem, as its name suggests, includes the following problem. Suppose there are some widely scattered factories and shops. Transportation cost from each factory to each shop is fixed by their condition. For instance, it depends on their distance, geography between their, and cost of custom duty. The problem is to transport the items composed at factories to shops with the least transport cost. On the other hands, this problem can be seen in problems that look like completely different. We consider a problem of looking for a picture similar to a query one in the database of pictures. For this purpose, we need a measure for determining "similarity" or "dissimilarity" among pictures. One of the measures is a distance function called the Earth Mover's Distance (EMD). First, we describe a picture by a Color Signature that is a set of points with weight. The Color Signature is taken the frequency distribution of the picture in the color space. Second, we calculate EMD with two Color Signatures, then we take a distance reflecting how similar two pictures are. To calculate the EMD, we need to find a solution of a Hitchcock transportation problem using EMD. The Hitchcock transportation problem has many other applications. Hence, we propose an efficient algorithm for comparing a minimum cost for a solution of the Hitchcock type transportation problem with a given threshold.

In an application to the EMD that we describe above, it is not so important to compute a correct distance between two pictures, but it is important to know that the distance between two pictures is below or above the given threshold. If the distance is below the given threshold, we decide two pictures are similar. And if the distance is above the given threshold, they are not similar. So we are going to focus on a decision problem for the Hitchcock type transportation problem in the following. Moreover, we may have extremely dissimilar pictures or those pictures whose distance is near the given threshold. This is not desirable. Therefore, in this paper, we propose an algorithm using iterative improvement that gradually compares the threshold with a lower bound of minimum cost of the Hitchcock type transportation problem that can be calculated with less computation time than that for finding an optimum minimum cost. We calculate a lower bound of the cost stepwisely, then we determine it in the stage that pictures are dissimilar if the lower bound is greater than the given threshold. Since it is rejected in an earlier stage when there is a big difference between the lower bound and the threshold, computation time also becomes shorter. In addition we can also find an optimum solution if we want.

In the general Hitchcock type transportation problem, it is difficult to establish a lower bound. Therefore, in this paper, we suppose that the nodes of the given graph are points with weights in the Euclidean space and the edges have the costs per unit flow between two points which are given by their distance. This hypothesis is correct when we consider a transportation problem in a practical situation and also EMD we have described above. We continue our argument under this hypothesis. An input graph is contracted, which scales down the problem. First, we contract a given graph. Concretely, we replace arbitrary two points with their centroid that has weight equal to a sum of their weights. We perform this operation until all points have been replaced. Considering result by this replace operation as one level, we continue to contract graph step by step until there is only one point in one level. Second, in the order of a level that has less point, we calculate a Hitchcock type transportation problem for the graph consisting points of the level. And we compare the cost of optimum solution taken in the stage with the given threshold. Since the cost of optimum solution in the stage is a lower bound of one in after stage, we can reject and answer "Pictures are not similar" in this stage if the taken cost is greater than the given threshold. Otherwise we proceed to the next stage to calculate a next lower bound of minimum cost and compare it. Then we finally take the optimum solution for the given graph if we reach the final stage. We can prove that the minimum cost of a graph contracted by our algorithm is a lower bound of minimum cost of a graph before it is contracted. Here we consider a case that the cost of the optimum solution is close to the given threshold. Namely, we cannot answer similarity or dissimilarity of two pictures without calculating a solution of the given graph that is not contracted. In this case, we are going to calculate all stages. But we can describe that time complexity of our algorithm does not increase for a naive algorithm even in the case.

We have implemented our algorithm using C++. We also used LEDA (is Library of Efficient Data types and Algorithms for C++). We have run our algorithm on Solaris. A simulation has been performed using points that are generated by a random number generator. We have measured average time for 100 trials in each case for 10, 20, 40, 80, 160, 320 points. In the result, we find that the computation time of our algorithm is further shorter as the number of points increases if a ratio of rejection is high enough. Conversely, if it is not, the computation time is greater than a naive algorithm. But we can look for many problems that the ratio is high in practice. Therefore, we can conclude that our method is useful.

Finally, we describe some future tasks. In this paper, we arbitrarily select two points to contract a graph. It seems that our algorithm can be improved if we have constraint on how to select them. We aim to improve a performance of our algorithm adding a constraint in selecting them. Alternatively we extend our method to Image retrieval application. After that, we will compare our method with existing methods. Additionally, we have changed value of threshold for getting desirable ratio of rejection. And so, we are going to lead theoretical value of a ratio of rejection corresponding to a threshold. Moreover we are going to derive the expected computation time that our algorithm is running under a fixed ratio of rejection.