Research Article

An Efficient Algorithm for Solving Minimum Cost Flow Problem with Complementarity Slack Conditions

Yongwen Hu1,2, Xiao Zhao1,2, Jing Liu1,3, Binyuan Liang1,2, and Chao Ma1,3

1Hubei Key Laboratory of Power System Design and Test for Electrical Vehicle, Hubei University of Arts and Science, Xiangyang 441053, China
2School of Mechanical Engineering, Hubei University of Arts and Science, Xiangyang 441053, China
3School of Automotive and Traffic Engineering, Hubei University of Arts and Science, Xiangyang 441053, China

Correspondence should be addressed to Chao Ma; mmma1123@163.com

Received 10 July 2019; Revised 11 October 2019; Accepted 6 November 2019; Published 12 February 2020

Copyright © 2020 Yongwen Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents an algorithm for solving a minimum cost flow (MCF) problem with a dual approach. The algorithm holds the complementary slackness at each iteration and finds an augmenting path by updating node potential iteratively. Then, flow can be augmented at the original network. In contrast to other popular algorithms, the presented algorithm does not find a residual network, nor find a shortest path. Furthermore, our algorithm holds information of node potential at each iteration, and we update node potential within finite iterations for expanding the admissible network. The validity of our algorithm is given. Numerical experiments show that our algorithm is an efficient algorithm for the MCF problem, especially for the network with a small interval of cost of per unit flow.

1. Introduction

The minimum cost flow (MCF) problem is to find a minimal cost of a given amount flow from a set of supply nodes to a set of demand nodes in a directed network where each arc is assigned a cost and a capacity constraint. MCF problem plays a fundamental role in network optimization. On the one hand, many network flow problems, such as shortest path problem, maximal flow problem, and transportation problem, can be viewed as special cases of the MCF problem. On the other hand, the MCF problem has a remarkable wide range of applications [1] in the real world, for example, scheduling [2], agriculture [3], communications [4], health care [5, 6], and transportation [7].

Since the MCF problem is a linear program, we can also find the optimal solution with linear programming methodologies. Indeed, many of various optimality conditions about the MCF problem and linear programming are derived by researchers. Thus, we can develop efficient algorithms for the MCF problem with theory of duality.

Several algorithms have been designed for solving the MCF problem over past decades both from theoretical and from practical aspects. Many algorithms combing ingredients of both shortest path and maximal flow are developed to solve the MCF problem. These algorithms find an optimal solution by solving a sequence of the shortest path problem within maximum flow-like residual networks and augmenting path, such as the cycle-canceling algorithm [8], successive shortest path algorithm [9], and primal-dual and out-of-kilter algorithms [10]. They are equivalent in the sense that these algorithms perform the same sequence of strategy, and the key step is defining the residual network iteratively. Moreover, both cost and capacity are to be considered for defining the residual network, and some more arcs are to be introduced with negative cost, which make the original problem much more difficult to solve. Although none of the algorithms mentioned above has been shown to be a well complexity, they usually behave much better in practice compared to their theoretical analysis. Edmonds and Karp [11] proposed the first polynomial algorithm. Currently, an efficient polynomial time algorithm for the MCF problem was proposed by Orlin [12] and Vygen [13]. Some theoretical results and algorithms are given in [14–18] for solving the MCF problem or its variants.
Recently, Vieira et al. [19] evaluated the practical performance of 4 algorithms on real networks. And Vernet et al. [20] used a success shortest path algorithm for dynamic graph. A recent survey on the MCF problem is given in [21].

Our purpose in this paper is to present a new algorithm for solving the MCF problem. We consider the dual problem of MCF problem and derive the optimality conditions. The algorithm can find an augmenting path directly in a directed network when some dual variables (nodes potentials) of the MCF problem are updated iteratively in finite steps. In contrast to other algorithms, we do not need to define any residual network.

This paper is organized as follows: Section 2 derives optimality conditions of MCF problem with duality theory. Section 3 gives an iterative algorithm for the MCF problem. Section 4 discusses validity of the presented algorithm. Numerical experiment results and analysis of computational experiments are reported in Section 5. Section 6 contains some concluding remarks on the proposed algorithm for solving the MCF problem.

2. Optimality Conditions

Let \( G = (N, A, C, U) \) be a directed network with a node set \( N \), an arc set \( A \), a cost set \( C \), and a capacity set \( U \). Furthermore, a cost \( c_{ij} \in C \) and a nonnegative capacity \( u_{ij} \in U \) associated with each arc \( (i, j) \in A, \forall i, j \in N \). And we associate with each node \( i \in N \) a number \( b_i \) which indicates its supply or demand depending on whether \( b_i < 0 \) or \( b_i > 0 \). Let node \( s \) and node \( t \) be the source node and sink node, respectively. And we assume that \( \sum_{i \in N, j \neq s} b_i = 0 \). The MCF problem can be stated as follows:

\[
\begin{align*}
\text{min } & \quad z(x) = \sum_{(i,j) \in A} c_{ij}x_{ij} \\
\text{MCF} & \quad \sum_{j: (j,i) \in A} x_{ij} - \sum_{j: (i,j) \in A} x_{ji} = \nu^s, \quad i = s, \\
& \quad \sum_{j: (j,i) \in A} x_{ij} - \sum_{j: (i,j) \in A} x_{ji} = -\nu^t, \quad i = t, \\
& \quad \sum_{j: (j,i) \in A} x_{ij} - \sum_{j: (i,j) \in A} x_{ji} = 0, \quad \forall i \in N, i \neq s, t, \\
& \quad 0 \leq x_{ij} \leq u_{ij}, \quad \forall (i, j) \in A.
\end{align*}
\]

The dual problem of MCF is

\[
\begin{align*}
\text{max } & \quad \omega = \nu^s p_s - \nu^t p_t + \sum_{(i,j) \in A} u_{ij}p_{ij} \\
\text{DMCF} & \quad p_t - p_j + p_{ij} \leq c_{ij}, \quad \forall (i, j) \in A, \\
& \quad p_t \text{ is free}, \quad \forall i \in N, \\
& \quad p_{ij} \leq 0, \quad \forall (i, j) \in A, i \in N,
\end{align*}
\]

where \( p_t \) and \( p_{ij} \) are the dual variables of the MCF problem. And we refer to \( p_t \) and \( p_{ij} \) as the potential of node \( i \) and arc \( (i, j), \forall i \in N, \forall (i, j) \in A \), respectively.

Let \( x = \{x_{ij}\} \) and \( p = \{p_t, p_{ij}\} \) be a feasible solution of MCF and DMCF, respectively. If feasible solution \( x = \{x_{ij}\} \) and \( p = \{p_t, p_{ij}\} \) are optimal solutions, according to theorem of commentary slackness, the following conditions will be satisfied:

\[
\begin{align*}
x_{ij} = 0, & \quad \text{if } p_t - p_j + p_{ij} < c_{ij}, \forall i \in N, (i, j) \in A, \quad (3) \\
x_{ij} = u_{ij}, & \quad \text{if } p_{ij} < 0, \forall (i, j) \in A. \quad (4)
\end{align*}
\]

Since \( p_t \) is free \( \forall i \in N \), let \( p_{ij} = \min \{0, c_{ij} - p_t - p_j\} \), then we have \( p_{ij} \leq 0 \). Therefore, we can find a feasible solution \( p = \{p_t, p_{ij}\} \) of DMCF where the value of \( p_{ij} \) is defined above \( \forall i \in N, \forall (i, j) \in A \). Thus, the complementary slackness conditions (3) and (4) are equal to the following conditions:

\[
\begin{align*}
x_{ij} = 0, & \quad \text{if } p_t - p_j < c_{ij}, \forall i \in N, (i, j) \in A, \quad (5) \\
x_{ij} = u_{ij}, & \quad \text{if } p_t - p_j > c_{ij}, \forall i \in N, (i, j) \in A. \quad (6)
\end{align*}
\]

Therefore, if feasible solution \( x = \{x_{ij}\} \) is an optimal solution of MCF, then \( x \) will satisfy conditions (5) and (6). On the contrary, for the arc \((i, j) \in A \) satisfying \( p_t - p_j = c_{ij} \), we have \( 0 \leq x_{ij} \leq u_{ij} \). In other words, we can find an augmenting path where each arc \((i, j) \in A \) on the path such that \( p_t - p_j = c_{ij} \) when the value of flow is not the maximum flow in a network. We refer to arc \((i, j) \in A \) satisfying \( p_t - p_j = c_{ij} \) as the admissible arc, and let \( G' = (N', A', C', U') \) be an admissible network satisfying \( N' \subseteq N, A' \subseteq A, C' \subseteq C, U' \subseteq U, p_t - p_j = c_{ij}, \forall (i, j) \in A' \).

3. An Iterative Algorithm for Solving MCF

It is clear that \( x_{ij} = 0, \forall (i, j) \in A \) is an optimal solution with a flow valued 0. In order to get an optimal solution of MCF with a given amount of flow, we iteratively augment flow holding the complementary slackness condition (5) and (6) at each iteration. And flow can be augmented after we find an augmenting path within the admissible network. Moreover, a new admissible network can be given by updating the potential of node when we cannot find an augmenting path and flow is not the maximal flow. Let \( V \) be a number of given amount of flow. We now describe formally our algorithm for solving the MCF problem (Algorithm 1).

Next, we will give the validity of the proposed algorithm.

4. Validity of the Proposed Algorithm

The validity of the above algorithm depends mainly from the following results:

Lemma 1. When the current flow is not the maximal flow in a network, at least one admissible arc can be found after updating node potential.

Proof. It is clear that there is at least one augmenting path if current flow is not the maximal flow in a directed network. Therefore, we have \( x_{ij} < u_{ij} \) for some forward arc \((i, j) \) or \( x_{ji} > 0 \) for some backward arc \((j, i) \), \( \forall i \in S, j \in \overline{S} \). Otherwise, node \( j \) will be labeled and \( j \in S \) according to the presented
Initialization: initialize node potential $p_i = 0$, $\forall i \in N$. Let $x_{ij} = 0$, $(i, j) \in A$, be an initial optimal solution with a flow valued 0. Label source node (0, +∞). Set $S = \{s\}$, $\overline{S} = N - S$.

While $\sum_{(i,j) \in A} x_{ij} \geq V$ or $\theta$ cannot be found

While $t \notin S$

For $i \in S$

$\theta_i = \min \left\{ \min_{j \in (S \cup \overline{S}) \setminus A} (c_{ij} - p_i + p_j), \min_{j \in (S \cup \overline{S}) \setminus A} (p_j - p_i - c_{ij}) \right\}$

End

$\theta = \min \{\theta_i\} \forall i \in S$

$p_i = \begin{cases} p_i + \theta, & \text{if } i \in S, \\ p_i, & \text{if } i \notin S. \end{cases}$

If $p_i - p_j = c_{ij}$ then

$(i, j) \in R$

End

If $t \in S$ then

For $\forall i \in S, j \in \overline{S}$

If $(i, j) \in R$ and $x_{ij} < u_{ij}$

Label node $j$ with a label $(i, \delta_j)$, where $\delta_j = \min \{\delta_i, u_{ij} - x_{ij}\}$, $S = S \cup \{j\}$.

End

If $(j, i) \in R$ and $x_{ji} > 0$

Label node $j$ with a label $(-i, \delta_j)$, where $\delta_j = \min \{\delta_i, x_{ji}\}$, $S = S \cup \{j\}$.

End

$x_{ij} = \begin{cases} x_{ij} + \delta_i, & \text{if } x_{ij} < \mu^i, \\ x_{ij} - \delta_i, & \text{if } x_{ij} > \mu^i, \\ x_{ij}, & \text{else} \end{cases}$

End

End

End

Algorithm 1

5. Numerical Experiment

In this section, the computational experimental results are presented in order to verify the performance of the presented algorithm. The presented algorithm has been coded with Matlab and run on PC with Intel® Core™ i5-7300HQ CPU (2.5 GHz) and RAM 8 GB. The tested problem is generated by the well-known Waxman network topology generator with the method proposed by Waxman [22]. All nodes of the generated problem are a Poisson process in the plane with scaled. Nodes $i$ and $j$ are connected with probability $p(i, j) = \alpha e^{-d(i,j)/\beta L}$, where $\alpha$ is the maximal link probability, $\beta$ is a parameter to control length of the arcs, $d(i, j)$ is the Euclidean distance between node $i$ and $j$, and $L$ is the maximal distance between any two nodes. For the network generated by the Waxman generator, we randomly give a $c_{ij}$ and $u_{ij}$ for each arc $(i, j) \in A$. The computational experiments are performed on a network with different scales ($n$ varied from 10 to 200). Let $n(m)$ be the number of nodes (arcs) of the generated network.

And for a fixed $n$ and $m$, a set of 10 instances are solved by our proposed algorithm and the successive shortest path algorithm (SSPA) [9] in order to make computational comparison. Furthermore, for a fixed $n$, our algorithm and SSPA are performed on a network with different number of arcs and $c_{ij}$, $i, j = 1, 2, \ldots, n$. For the sake of simplicity, the algorithm proposed in [9] is named as Algorithm A1 and the algorithm.
In this paper, we proposed an algorithm for solving the MCF problem with a dual approach which holds a complementary slackness conditions at each iteration. The key step of our algorithm is updating node potential, and flow can be augmented after updating node potentials in finite steps.

Table 1: CPU time on network with \( n \) varying from 10 to 200 and two different intervals of \( c_{ij} \).

<table>
<thead>
<tr>
<th>No. of nodes</th>
<th>No. of arcs (average)</th>
<th>AVER. CPU time (s), ( c \in [1, 10] )</th>
<th>AVER. CPU time (s), ( c \in [1, 1000] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>32</td>
<td>7.964 0.050</td>
<td>8.002 0.054</td>
</tr>
<tr>
<td>15</td>
<td>62</td>
<td>8.430 0.060</td>
<td>8.517 0.078</td>
</tr>
<tr>
<td>20</td>
<td>86</td>
<td>8.642 0.079</td>
<td>8.726 0.085</td>
</tr>
<tr>
<td>30</td>
<td>210</td>
<td>8.954 0.087</td>
<td>8.936 0.164</td>
</tr>
<tr>
<td>40</td>
<td>418</td>
<td>9.014 0.166</td>
<td>9.201 0.258</td>
</tr>
<tr>
<td>50</td>
<td>371</td>
<td>8.976 0.092</td>
<td>9.005 0.255</td>
</tr>
<tr>
<td>60</td>
<td>763</td>
<td>9.356 0.278</td>
<td>9.428 0.423</td>
</tr>
<tr>
<td>80</td>
<td>1186</td>
<td>9.628 0.364</td>
<td>9.617 0.673</td>
</tr>
<tr>
<td>100</td>
<td>1570</td>
<td>9.725 0.247</td>
<td>9.829 1.056</td>
</tr>
<tr>
<td>200</td>
<td>2000</td>
<td>10.320 0.344</td>
<td>10.431 1.734</td>
</tr>
<tr>
<td>400</td>
<td>4857</td>
<td>10.569 1.550</td>
<td>10.726 6.438</td>
</tr>
<tr>
<td>800</td>
<td>9488</td>
<td>18.351 2.069</td>
<td>19.251 37.181</td>
</tr>
<tr>
<td>1600</td>
<td>19482</td>
<td>210.358 23.429</td>
<td>221.852 143.759</td>
</tr>
</tbody>
</table>

Unlike existing algorithms such as successive shortest path algorithm and primal-dual algorithm, our algorithm does not find the residual network, nor find the shortest path iteratively. We conduct a numerical experiment to investigate the performance of the proposed algorithm. The results of the numerical experiment show that our algorithm reduces CPU time greatly compared with SSPA.

In this paper, the cost of flow is fixed, and we plan to develop an algorithm for solving the MCF problem with fuzzy cost and constraints. In practice, it is worthwhile to study the case that we have a close-to-optimal solution for the MCF problem. It is fundamentally important to transform the multi-object assignment problem and multi-object transportation problem into the MCF problem, which could be our further study.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by the Hubei Superior and Distinctive Discipline Group of “Mechatronics and Automobiles” under grant XKQ2019005, the Youth Project of Humanities and Social Sciences financed by the Ministry of Education under grant 17YJC630084, and the Starting Research Fund from the Hubei University of Arts and Science. The authors sincerely appreciate all the supports.

References


