International Journal of Statistics and Applied Mathematics 2022; 7(4): 267-270

International Journal of Statistics and Applied Mathematics

ISSN: 2456-1452
Maths 2022; 7(4): 267-270

www.mathsjournal.com
Received: 27-05-2022
Accepted: 02-07-2022

Anjana Devi Bhandari
Kantipur Engineering College, Tribhuvan University, Kathmandu, Nepal

Madan Bahadura Chand
Gokuleshwor Multiple Campus, Far-western University, Mahendranagar, Nepal

Corresponding Author:
Madan Bahadura Chand
Gokuleshwor Multiple Campus, Far-western University, Mahendranagar, Nepal

A review on the quickest flow problem

Anjana Devi Bhandari and Madan Bahadura Chand

DOI: https://doi.org/10.22271/maths.2022.v7.i4c.870

Abstract
Path problems were basically studied to find an alternate path that is, finding a second shortest route, if the route is blocked. The shortest path problem is mainly focused on finding the shortest paths between the vertices of a given network. A new variant of shortest path problem is quickest path problem, where the predetermined data is sent from the source to the sink. The quickest flow problem relaxes the limitations of single path to multiple paths. In this paper we reviewed the shortest path problem, quickest path problem and quickest flow problem. Later on, each problem is clarified with certain examples.

Keywords: Path, shortest path, quickest path, quickest flow

Introduction

Without viable distribution and coordinate networks, we are unable to have present day goods and services at moderate costs. In mathematics, networks are often called graphs, and the field of mathematics dealing with the study of graphs is called graph theory. The Seven Bridges Konigsberg is generally an eminent network problem in mathematics. Its negative goal by Leonard Euler in 1736 establishes the framework of graph theory and symbolizes the possibility of topology. The scope of network flow problems is the application of graph theory mounted with optimization theory to practical problems. A network flow problem related to the Euler tour is the famous Chinese Postman problem which attempts to find a tour with minimal repetition of the edges.

Going back to the past of evacuation planning, path problems were studied at the beginning of 1950's. One can imagine that even in very ancient (even animal) societies, finding a short path (for instance, to food) is necessary. Dijkstra's algorithm is considered as the first shortest path algorithm for finding the shortest path between two nodes in a graph, which may represent, for example, a road network. Dijkstra's algorithm solves the single-source shortest path problem with non-negative edge weight and runs in time O (n), where n is the number of nodes. A couple of notable algorithms to solve shortest paths are Bellman Ford algorithm and Floyd Warshall algorithm. The Bellman Ford algorithm solves the single-source problem if the edge weight is negative (edge weight negative means a weighted graph in which the total weight of an edge is negative). The Floyd Warshall solves all pairs for shortest paths.

Further, Chen and Chin (1990) [5] proposed a new variant of shortest path problem as a quickest path problem; they developed several algorithms to find a quickest path to send a certain amount of data from the source to the sink. Later, Rosen *et al*. (1991) [17] developed the algorithms for the quickest path problem and also they proposed a figure on corresponding quickest paths. Further, they developed an efficient algorithm for the quickest path problem to enumerate the first p quickest paths to send a given amount of data from one node to another with time complexity O (pr(n.m+n^2 □log log n)),where m is the number of arcs, n is the number of vertices, r is the number of distinct capacity values of a given network N.

In the quickest path problem, the solution is mainly dependent on travel time and the number of units to be sent along a single path. In most of the practical situations our network model is of multiple paths. In this case, the quickest path problem fails to meet the solution. Therefore, a new problem mainly known as quickest flow problem which represents the realistic situation, was introduced by the authors Burkard *et al*. (1993) [2]. Unlike the quickest path problem, the quickest flow problem relaxes the limitation of single path to multiple paths. In the quickest path problem the amount of flow can be transmitted through only one path. Later on Burkard *et al.* (1993) [2] investigated the generalization that data can be sent simultaneously along two disjoint paths from the same starting node to the end node in network N. Their work showed that the quickest flow problem is closely related to the maximum dynamic flow problem and to linear fractional programming problems, Ibraki (1983) [9]. Further these authors developed several polynomial time algorithms and strongly polynomial algorithm for this quickest flow problem using continuous version of flow value $v(T)$ in case of single source and a single sink of the best time complexity.

$$O(m^2 log^3(n(m + n \log \log n)).$$

The maximum dynamic flow is closely related to the min-cost flow problem. Therefore, it can be solved by using various minimum cost flow algorithms in polynomial time. Authors in Burkard *et al*. (1993) [2] developed an improved algorithm by using a parametric search method in polynomial time. These algorithms were mainly based on binary search techniques and maximum dynamic flow algorithms. A new example of this approach can be found in Bhandari & Dhamala (2020) [1]. In this paper, we study the formulation of shortest path problem, quickest path problem and quickest flow problem.

The structure of this paper is organized as follows: Shortest path problem is formulated in section 2. In section 3, we elaborated the formulation of the quickest path problem. The shortest path may not always be the fastest one, as demonstrated by one example. In section4, mathematical formulation of the quickest flow problem is explained. Finally, the paper is closed with some concluding remarks in section 5.

2. Shortest Path Problem (SPP)

The shortest path problems (SPPs) are perhaps the most important problems in combinatorial operations research. SPP is about finding a path between two vertices in a graph such that the total sum of the edges weights is minimum.

Definition 1. Consider a directed network $N = (V, A, c_{ij}, \tau_e)$, where V and A represent finite sets of nodes and arcs, respectively. For each arc (i, j), x_{ij} be a flow on it. Let x_{ij} be 1 if an arc (i, j) is in the path and 0 otherwise.

$$Min \sum_{(i,j)\epsilon A} c_{ij}\ x_{ij}$$

Subject to

$$\sum_{(i,j)\epsilon A} x_{ik} - \sum_{(i,j)\epsilon A} x_{kj} = \begin{cases} 1\ k = s \\ -1\ k = d \\ 0\ k \in A \setminus \{s, d\} \end{cases}$$

The most important algorithms and its time complexity are listed below in table: 1

Table 1: *Shortest path Algorithm and its time complexity*, Magnanti *et al*. (1993) [12]

Shortest Path Algorithms	Time Complexity
Dijkstra's algorithm	$O(m + n \log \log n)$
Bellman-Ford	$O(mn)$
A* Search	$O(m)$
Floyd- Warshall	$O(n^3)$
Johnson's algorithm	$O(n^2 \log \log n + nm)$

3. Quickest Path Problem (QPP)

Let v be a given amount of data that has to be sent from the source s to the sink d in a given network$N$. Assume thatτ_e, is the lead time for the transmission of data from node $e = (i, j)$ and c_{ij} be the capacity of arc (i, j).

Definition 2. Let P be the set of paths from s to d in network N Calvete *et al.* (2012) [3]. We assume that $P \neq \emptyset$. The quickest path problem can be formulated as finding a path $p \in P$ so that

$$Min\ T(v, p)$$

Such that

$$p \in P$$

Example 1. Let us see an important relation between the quickest path and the shortest path by considering a network model in $Figure$ 1 with two terminals. Node A is the unsafe place (source) that contains evacuee and node D is the safe place (sink) with sufficient capacity and the remaining nodes are intermediate. The first and second quantity attached with each arc is the capacity and transit time of an arc, respectively. For instance, an arc directed (A, B) between the node A and B has its capacity 2 and transit time 3. Suppose we want to send 1000 evacuees from node A to node D.

Consider all possible paths starting from source node A to sink node D and denote the corresponding paths as $P_1 = (ABD), P_2 = (ACBD), P_3 = (ACD)$. The quickest path is P_3 with egress time $T(1000, P_3) = \lambda(P_3) + \frac{v}{c(P_3)} = 260$. But the shortest path is P_1. Therefore, the quickest path may not always be a shortest path. Furthermore quickest path violates concatenated property, i.e., a sub-path of a quickest path may not be a quickest path.

On the other hand, it is worth mentioning at this point the importance of the number of items to be sent in the computation of the quickest path. In the same Figure 1, if we send 4 items from node A to node D, the quickest path is P_1 i.e., $A - B - D$. However, the quickest path to send 1000 items is $A - C - D$. Generally speaking, if v is rather small, the quickest path tends to be the shortest path according to lead time. However, if v is large, the quickest path leads to the path with the largest capacity.

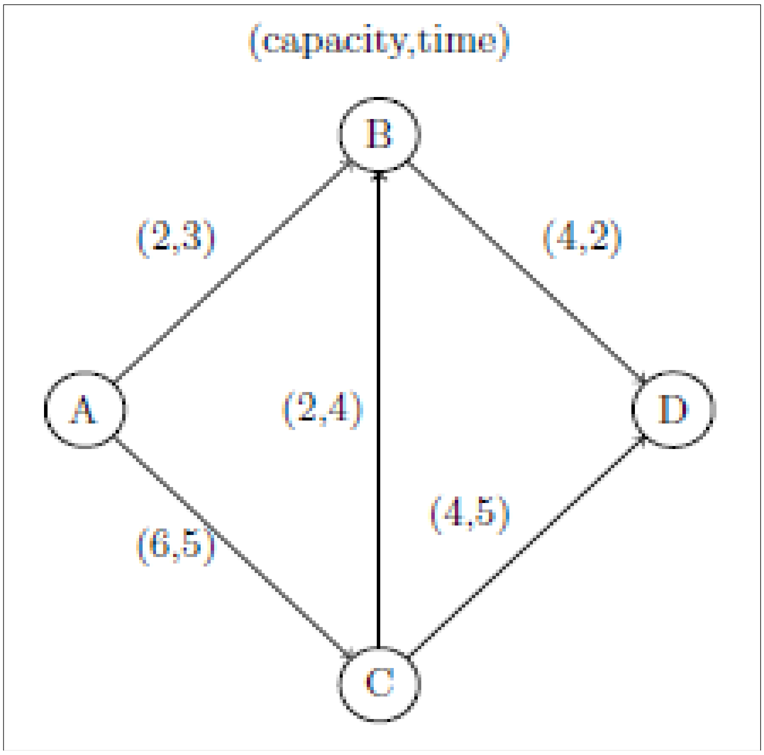

Fig 1: Network model

4. Quickest Flow Problem (QFP)

The quickest flow problem (QFP) is a dynamic transshipment problem which modifies the restriction of a single path to multiple paths that clear the network in minimum possible time. In the QFP, we consider integer capacities c_{ij} and non-negative integer transmission time τ_e. Let $I_T = \{0,1,\ldots,T\}$ be a discrete time interval with $T \in N_0$ and let N_s be the set of nodes $V \setminus \{s,d\}$.

Definition 3. Quickest Flow Problem

Consider a mapping $x: A \times I_T \rightarrow N_0$, a quickest flow in a network N from the source to the sink for a given flow value v is defined as follows:

Min f_0T=T (9)

and it satisfies the following properties

$$\sum_{t=0}^{T} \Big(\sum_{(s,j)\epsilon A} x_{sj}(t) \; - \sum_{(i,j)\epsilon A} x_{is}(t-\tau_{is}) \Big) = \; v$$

$$\sum_{(i,j)\epsilon A} x_{ij}(t) \; - \sum_{(k,i)\epsilon A} x_{ki}(t-\tau_{ki}) = \; 0 \qquad \forall\, i \in N_s, \quad \forall\; t \in I_T,$$

$$\sum_{t=0}^{T} \sum_{(d,j)\epsilon A} x_{dj}(t) \; - \sum_{(i,d)\epsilon A} x_{id}(t-\tau_{id}) \; = \; -v$$

$$0 \leq x_{ij}(t) \leq c_{ij} \;\; \forall (i,j) \in A, \quad \forall\, t \in I_T$$

Where Where $x_{ij}(t)$ denotes the flow from i to j at time t and the value of $x_{ij}(t)$ is always zero wshen $t < 0$.

Equation (2) explains that the flow which leaves the source s in the time interval I_T totals v. This flow reaches during this time in the sink d, described in (4). Flow conservation is described in equation (3). Equation (5) represents the capacity constraints at any time t.

Table 2: An overview of literature flow in QPP and QFP

Contributions	Contributed by
QPP was first initiated by Moore to model flows of convoy-type traffic through networks in which the whole volume of traffic, initially located at the source node, must reach at the sink node in as little time as possible when all the traffic must flow along the same path and the rates of flow along the arcs limited by flow rate constraints.	Moore (1976) [14]
Proposed new variant of shortest path problem, known as quickest path problem and developed algorithms for the single pair QPP with time complexity $O(m^2 + nm\, log\, log\, m)$ where $m = \lvert V \rvert, n = \lvert A \rvert$ when the amount of data to be shipped is given.	Chen and Chin (1990) [5]
Developed an algorithm to enumerate the first $p$quickest paths to send a given amount of data from one node to another with time complexity $O(pr(mn + m^2\, log\, log\, m\,))$, where r is the distinct capacity of network N.	Rosen *et al.* (1991) [17]
QPP is considered as a bicriteria problem and also developed efficient algorithm of time complexity $O(mn\, log\, log\, m\, + m^2)$.	Martins and Dos Santos (1997) [13]
First polynomial time algorithm for the integral quickest transshipment problem with multiple sources and sinks.	Hoppe and Tardos (2000) [8]
The problem of finding the first p quickest paths in non- decreasing order of transmission time $(p - QPP)$ is analyzed.	Chen (1994) [4], Pascoal *et al.* (2006) [15]

Modified the algorithm of Chen and Chin also analyzed the problem of identifying the quickest path whose reliability is not lower than a given threshold.	Calvete *et al*. (2012) [3]
Revealed that QFP is closely connected to the maximum dynamic flow problem and to linear fractional programming problems and developed several polynomial and strongly polynomial algorithms to solve the quickest flow problem.	Burkard *et al*. (1993) [2]
Developed a new cost-scaling algorithm that holds the parametric nature of the problem. The algorithm solves the quickest flow problem with integer arc costs in the same time complexity as that of the minimum cost flow algorithm.	Lin and Jaillet (2015) [11]
Modified the algorithm of Lin and Jaillet to a strongly polynomial time algorithm	Saho and Shigeno (2017) [18]
Solved the quickest contraflow problem with constant transit time on arcs in strongly polynomial time.	Pyakurel *et al*. (2018) [16]

5. Conclusion
In this paper a detailed analysis is presented to the formulation of shortest path problem, quickest path problem & quickest flow problem. The relation between shortest paths, quickest path and quickest flow is discussed. Later, an overview of the literature flow of the mentioned problem is highlighted.

References

1. Bhandari AD, Dhamala TN. On quickest flow problem using improved binary search algorithm. International Journal of Innovative Science, Engineering & Technology. 2020;7:12.
2. Burkard RE, Dlaska K, Klinz B. The quickest flow problem. Zeitschrift fur Operations Research. 1993;37(1):31-58.
3. Calvete HI, Del Pozo L, Iranzo JA, Algorithms for the quickest path problem and the reliable quickest path problem. Computational Management Science. 2012;9(2):255-272.
4. Chen Y. Finding the k quickest simple paths in a network. Information Processing Letters. 1994;50(2):89-92.
5. Chen Y, Chin Y. The quickest path problem. Computers & Operations Research. 1990;17(2):153-161.
6. Dhamala TN. A survey on models and algorithms for discrete evacuation planning network problems. Journal of Industrial & Management Optimization. 2015;11(1):265-289.
7. Fulkerson D, Ford L. Flows in networks. Princeton University Press; c1962.
8. Hoppe B, Tardos E. The quickest transshipment problem. Mathematics of Operations Research. 2000;25(1):36-62.
9. Ibaraki T. Parametric approaches to fractional programs. Mathematical Programming. 1983;26(3):345-362.
10. Khadka SR, Bhandari PP. Dynamic network contraflow evacuation planning problem with continuous time approach. International Journal of Operations Research. 2017;14(1):27-34.
11. Lin M, Jaillet P. On the quickest flow problem in dynamic networks: a parametric min-cost flow approach. In Proceedings of the twenty-sixth annual ACMSIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics; c2015. p. 1343-1356.
12. Magnanti T, Ahuja R, Orlin J. Network flows: theory, algorithms, and applications. PrenticeHall, Upper Saddle River, NJ; c1993.
13. Martins EDQV, Dos Santos JLE. An algorithm for the quickest path problem. Operations Research Letters. 1997;20(4):195-198.
14. Moore MH. On the fastest route for convoy-type traffic in flowrate-constrained networks. Transportation Science. 1976;10(2):113-124.
15. Pascoal MM, Captivo MEV, Cl´ımaco JC. A comprehensives survey on the quickest path problem. Annals of Operations Research. 2006;147(1):5-21.
16. Pyakurel U, Dhamala TN. Continuous time dynamic contraflow models and algorithms. Advanced in Operations Research; c2016. p. 1-6.
Pyakurel, U., Nath, H. N., and Dhamala, T. N., Efficient contraflow algorithms for quickest evacuation planning. Science China Mathematics. 2018;61(11):2079-2100.
17. Rosen JB, Sun SZ, Xue GL. Algorithms for the quickest path problem and the enumeration of quickest paths. Computers & Operations Research. 1991;18(6):579-584.
18. Saho M, Shigeno M. Cancel-and-tighten algorithm for quickest flow problems. Networks. 2017;69(2):179-188.