ОБМІН ПРАКТИЧНИМ ДОСВІДОМ ТА ТЕХНОЛОГІЯМИ

УДК 621.391

O.V. TYKHONOVA
O.S. Popov Odessa National Academy of Telecommunications

THE PRINCIPLES OF MAXFLOW TASK STUDY FOR MULTI-POLE SOFTWARE DEFINED NETWORK

In this paper, a brief survey made on network max flow task problem and commonly known algorithms for max flow task solution. Concluded, that no exhaustive solution of max flow task exists in general case of an undirected multi-pole network. It is noticed, that known applied network graph models do not actually meet real networking technologies and network dynamic reconfiguration abilities. To overcome these issues, three basic principles of max flow task statement introduced for undirected multi-pole weighted graph of a software defined network. The first principle supposes to consider communication link as a "flexibly reconfigurable channel" where the overall channel capacity is an invariant for dynamic scheduling between the forward and backward link directions. The second principle is that the communication network is viewed as a kind of a shell which divides and unites two open network environments – outer and inner neighborhoods. These neighborhoods are simulated by special abstract vertices on the graph, and the network nodes connected with these vertices are considered as poles opened "outside" and/or "inside" respectively. In this approach, there is no need on special nodes like "source" or "destination" node. Each of the network nodes can be the flow source and/or the flow destination. According to the third principle, three types of elementary network activities distinguished in multi-pole network: "inner local activity" (ILA), "outer transit activity" (OTA), and "inner-to-outer activity" (IOA). The ILA activity describes relationships between the inner neighborhood nodes through the considered network nodes. The OTA activity describes transit flows of outer neighborhood via the given network as a transporting communication infrastructure. The IOA activity describes the communication between the inner and outer neighborhoods via the given network infrastructure. Based on these principles, the max flow tasks classified on seven basic max flow indexes. In this classification, the known two-pole network transporting task is a part of the "outer transit activity" class.
Keywords: max flow task, multi pole graph, software defined network.

О.В. ТИХОНОВА
Одеська національна академія зв'язку ім. О.С.Попова

ПРИНЦИПИ РОЗВ'ЯЗАННЯ ЗАДАЧІ ПРО МАКСИМАЛЬНИЙ ПОТІК ДЛЯ БАГАТОРОЛЮСНОЇ ПРОГРАМНО КОНФІГУРОВАНОЇ МЕРЕЖІ

Наведено стислий огляд відомих методів розв'язання задачі про максимальний потік у мережі. Зроблено висновок, що задача пошуку максимального потоку мережі не може вважатися вирішеною повністю у загальному випадку для багатополюсної мережі. Запропоновано базові принципи розв'язання задачі про максимальний потік у багатополюсному зваженому графі програмно конфігурованої мережі.
Ключові слова: задача максимального потоку, багатополюсний граф, програмно конфігурована мережа.

1. Introduction

The max flow task (denote it MFT) commonly implies finding a feasible "product flow" through a network graph with solely one *source node S* (assumed to generate an unlimited product flow) and solely one *target node T* (assumed to absorb an unlimited product flow). The "product" term (denote it *P*) means any measurable substance (water, gas, goods etc.), [1–2]. In telecommunication networks, the "product" usually means "information" measured in bits, bytes or other units. The related "product flow" in telecommunication network (denote it F) is measured in bit per second (bps), Kilobit per second (Kbps), Megabit per second (Mbps) etc.

The typical prerequisites in MFT study are:

a) network topology and metrics defined by directed weighted network graph *G*;

b) an indication of two distinguished network nodes *S* and *T*;

c) hypothesis that none of graph vertices (except *S* and *T*) can accumulate, generate or consume the product *P*;

d) an assumption of *S* and *T* nodes "product power" generation/consumption ability (conventionally supposed to infinity), [3–4].

The MFT intends to find the maximal possible product flow (MPF) between the source node *S* and the target node *T* on the given graph *G*, fig.1.

In the MFT statement on fig.1, the maximal product flow MPF satisfies the well known Ford-Fulkerson theorem [3] which says that max product flow MPF equals the minimal network graph cut C_{min} which divides the graph *G* in two parts $G_S \in S$ and $G_T \in T$; here C_{min} is the minimal overall capacity of directed edges which cross the cut from $G_S \in S$ to $G_T \in T$.

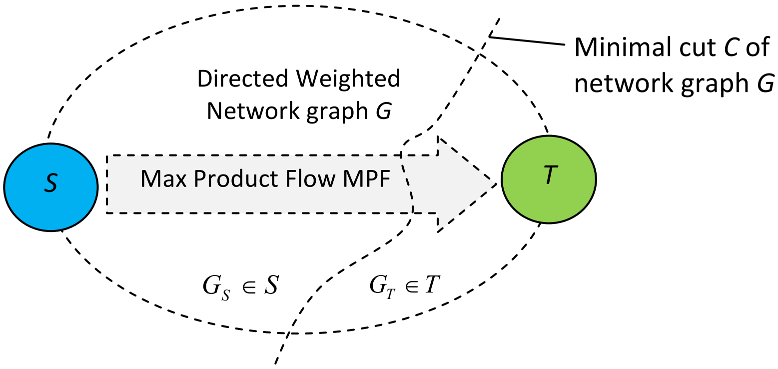


Fig.1. The typical max flow task statement

The widely known solution of the MFP is Ford–Fulkerson algorithm (FFA), also mentioned as "method", because it solely finds augmenting paths in a residual graph without full specification, which typically needed in an accurate algorithm (1956, [3]). A directed graph (or digraph) is often defined as that, where all the edges are unidirectional (i.e. directed from one vertex to another), in contrast to undirected graph (or non-digraph), where all the edges are bidirectional [2]. Sometimes, however, a digraph is understood as that, where at least one edge is either unidirectional or bidirectional, whereas a graph with all unidirectional edges is called "oriented graph"; a graph with no one directed edge is considered undirected, [5]. To make easy terms, we will use "ark" (or "simplex") for unidirectional edge; in contrast to bidirectional edge noted "bi-ark" (or "complex"). So far the "undirected weighted graph" (UWG) is not uniformly perceived by the specialists; once, UWG is taken as everywhere symmetrically weighted bidirectional graph, or skew-symmetric graph.

The main idea behind the FFA is scheduling the graph traversing flows until the residual throughput from the source S to the target T exhausted. Any path having positive capacity is called "augmenting path". To explore actual paths traversing the graph in the FFA process, the depth first search (DFS) technique is applied: start at the root and follow one of the tree branches, until the seeking vertex (sink) reached or a leaf node met (with no children more). The FFA has run time complexity $O(E \cdot f)$, where E is the number of graph edges, f is the graph max flow index, $O(\ )$ is commonly used descriptor for the run time complexity of an algorithm. Over the years, enhanced solutions to the MFP have been conceived, notably the blocking flow algorithm of Dinitz (DFA, 1970, [4]); the Edmonds–Karp algorithm (EKA) of the shortest augmenting path (1972, [6]); the push–relabel max flow algorithm of Goldberg and Tarjan (1987, [7]); the binary blocking flow algorithm of Goldberg and Rao (1998, [8]); the electrical flow algorithm of Christiano, Kelner, Madry and Spielman (2011, [9]). The MFT can be also observed as a particular case of more complex graph circulation task (GCT), formulated along with additional constraints such as fixed lower bound on the edge flows, excluded conservative/emanative vertices, (i.e. no special vertices required more), multiple flow commodities, flow cost accounting, [10].

The Dinitz's DFA algorithm runs in $O(V^2 \cdot E)$ time and is similar to EKA in usage the shortest augmenting paths. However, Dinitz's concepts of the level graph and blocking flow makes DFA to achieve better performance vs. $O(E \cdot f)$ in FFA. On the other hand, the FFA is often referred to as EKA; though, the latter provides full specification of the FFA approach. Besides, the EKA claims that breadth-first search (BFS) but not DFS must be applied in the algorithm for graph exploration, in order to find the shortest paths. Also, the EKA improves the runtime complexity of FFA from $O(E \cdot f)$ to $O(V \cdot E^2)$ index, where V is the number of graph vertices. This improvement is vigorously important, as EKA runtime turns independent on the graph max flow. In contrast to DFS style of graph searching, the BFS, when started at arbitrary graph vertex (aka "search key"), goes on exploring the neighbor nodes first, before moving to the next level neighbors. In literature, also known generalizations of the max flow task for multi-pole directed graph, e.g. the bipartite graph framework with multiple sources and targets [11], or the particular case of multi-pole directed balanced graph (1999, [12]); directed weighted graph is balanced, if any vertex of the graph satisfies the following property: the overall input arks' capacity equals the overall output arks' capacity.

By and large, we conclude, that max flow task is not exhaustively studied yet in general case for undirected multiple-pole weighted graph. Besides, commonly applied graph models do not actually meet real networking technologies; for instance, the known mapping of a link among two network nodes by the fixed directed or undirected "edge" of a network graph ignores the flexible reconfiguration ability of modern optical channels in software defined network architecture. Considering aforesaid observation, *this work aims to substantiate basic principles of max flow task study for multi-pole software defined network with reconfigurable undirected channels*.

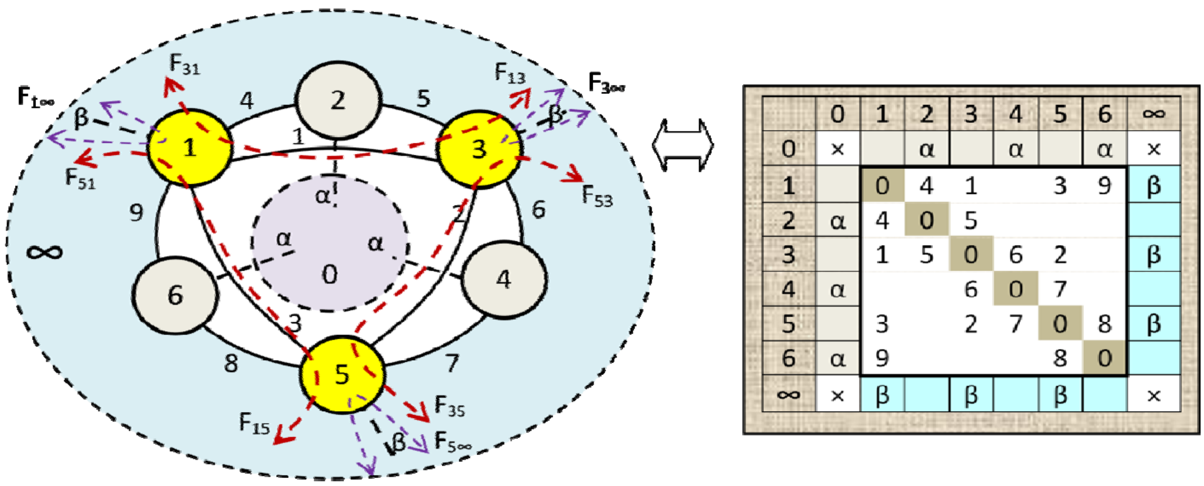
2. The principles of max flow task study for a multi-pole network with reconfigurable undirected channels

To achieve declared above goal, the following three principles of max flow task study formulated for a multi-pole software defined network with reconfigurable undirected channels [13].

1) *Cognitive insight* on communication link as "*flexible reconfigurable channel*" (FRC) to connect adjacent nodes in a network graph model instead of commonly used fix directed linkage [14, 15]. The FRC-link encompasses known types of graph edges (like unidirectional and bidirectional edges, as well as diverse interpretations of what is declared as "undirected edge"); but herewith, the FRC concept of flexible nodes linkage brings new opportunities towards network performance optimization.

2) *Unified view* on network graph model as a *dual open shell* (DOS) [16], which has its own transfer infrastructure, as well as two open neighborhoods: *inner local area network* (LAN) and *outer wide area network* (WAN). The DOS shell has inner and outer opened poles to communicate with LAN and WAN; some poles can be dual inner/outer opened. In a graph presentation, let LAN entity denote as an abstract *null-vertex* V_0 , and WAN entity as an abstract *infinity-vertex* V_∞ .

Figure 2 shows an example of a visual topology/metrics graph and correspondent matrix topology/metrics graph for a network which has three outer poles (vertices number 1, 3, 5) and three inner poles (vertices number 2, 4, 6); α and β are inner and outer throughput indexes. This point of view eliminates necessity of exclusive network nodes traditionally observed in max flow task (such as "source node" *S* emanating flow and "target node" *T* consuming flow). The DOS shell framework is formed by closed bilateral links which connect the shell vertices according given topology and metrics. Inner/outer shell transparency provided by open inner/outer links.



a) Visual graph b) Matrix graph

Fig.2 Visual and correspondent matrix topology/metrics graphs for a "3 outer/3 inner-pole" network

All the closed and open links in a network graph understood as flexible reconfigurable channels (FRCs) having individual capacities.

In general case of max flow task consideration, all the graph vertices (incl. network poles, LAN and WAN) may simultaneously generate and/or consume product flows. However, in this observation, the pure transporting network will be discussed furthermore with no generation or consumption in network nodes.

3) *Comprehensive vision* on max flow task in multi-pole network, where three types of network activities distinguished: a) "inner local activity" (ILA), "inner-to-outer activity" (IOA), "outer transit activity" (OTA), fig. 3. Respectively, three independent types of max flow indexes observed in a DOS: ILA max flow (F_{ILA}^{max}), IOA max flow (F_{IOA}^{max}) and OTA max flow (F_{OTA}^{max}).

Each of these three max flow indexes can be estimated separately. From these observations, the following generalization can be formalized for network max flow task

$$f\left(F_{ILA}, F_{IOA}, F_{OTA}\right) = max \quad . \tag{1}$$

a)

	0	1	2	3	4	5	6	∞
0	×	×	f_{020}	×	f_{040}	×	f_{060}	×
1	×	×	×	×	×	×	×	×
2	f_{020}	×	×	×	f_{24}	×	f_{26}	×
3	×	×	×	×	×	×	×	×
4	f_{040}	×	f_{42}	×	×	×	f_{46}	×
5	×	×	×	×	×	×	×	×
6	f_{060}	×	f_{62}	×	f_{64}	×	×	×
∞	×	×	×	×	×	×	×	×

b)

	0	1	2	3	4	5	6	∞
0	×	×	×	×	×	×	×	×
1	×	×	f_{12}	×	f_{14}	×	f_{16}	×
2	×	f_{21}	×	f_{23}	×	f_{25}	×	×
3	×	×	f_{32}	×	f_{34}	×	f_{36}	×
4	×	f_{41}	×	f_{43}	×	f_{45}	×	×
5	×	×	f_{52}	×	f_{54}	×	f_{56}	×
6	×	f_{61}	×	f_{63}	×	f_{65}	×	×
∞	×	×	×	×	×	×	×	×

c)

	0	1	2	3	4	5	6	∞
0	×	×	×	×	×	×	×	×
1	×	×	×	f_{13}	×	f_{15}	×	$f_{\infty 1\infty}$
2	×	×	×	×	×	×	×	×
3	×	f_{31}	×	×	×	f_{35}	×	$f_{\infty 3\infty}$
4	×	×	×	×	×	×	×	×
5	×	f_{51}	×	f_{53}	×	×	×	$f_{\infty 5\infty}$
6	×	×	×	×	×	×	×	×
∞	×	$f_{\infty 1\infty}$	×	$f_{\infty 3\infty}$	×	$f_{\infty 5\infty}$	×	×

Fig.3 Three types of network flow activities: a) inner local activity (ILA); b) inner-to-outer activity (IOA); c) outer transit activity (OTA)

We note that in contrast to the two-pole *S-T* directed graph, the term "network flow" in multi-pole network with flexible reconfigurable channels may cause some difficulties in understanding. To clear up this term, we consider distinct max flow indexes F_{ILA}^{max}, F_{IOA}^{max} and F_{OTA}^{max}. The mostly similar to known approaches for max flow task for a transport network is the F_{OTA}^{max} index which means the maximal possible flow sum related to outer network environment (∞) and each of three outer poles 1, 3 and 5 in fig.2. There are two types of outer transport flows, fig.3, c:

a) transit flows which traverse different outer poles: $F_{13}, F_{31}, F_{15}, F_{51}, F_{35}, F_{53}$;

b) reflected flows each solely traversing a single outer pole: $F_{\infty 1\infty}, F_{\infty 3\infty}, F_{\infty 5\infty}$.

Now, the max flow index F_{OTA}^{max} is defined as $f\left(F_{ILA}, F_{IOA}, F_{OTA}\right) = F_{OTA} = max$:

$$F_{OTA} = \left[\left(\left|F_{13}\right| + \left|F_{31}\right|\right) + \left(\left|F_{15}\right| + \left|F_{51}\right|\right) + \left(\left|F_{35}\right| + \left|F_{53}\right|\right)\right] + \left(\left|F_{\infty 1\infty}\right| + \left|F_{\infty 3\infty}\right| + \left|F_{\infty 5\infty}\right|\right) = max \quad . \qquad (2)$$

Similarly, a symmetric transport max flow task arises $f\left(F_{ILA}, F_{IOA}, F_{OTA}\right) = F_{ILA} = max$ when inner local activity F_{ILA}^{max} considered:

$$F_{ILA} = \left[\left(\left|F_{24}\right| + \left|F_{42}\right|\right) + \left(\left|F_{26}\right| + \left|F_{62}\right|\right) + \left(\left|F_{46}\right| + \left|F_{64}\right|\right)\right] + \left(\left|F_{020}\right| + \left|F_{040}\right| + \left|F_{060}\right|\right) = max \quad . \qquad (3)$$

This means that network transporting infrastructure solely used for local product transportation.

The inner-to-outer max flow index F_{IOA}^{max} is defined as $f\left(F_{ILA}, F_{IOA}, F_{OTA}\right) = F_{IOA} = max$:

$$F_{IOA} = \left(\left|F_{12}\right| + \left|F_{21}\right|\right) + \left(\left|F_{14}\right| + \left|F_{41}\right|\right) + \left(\left|F_{16}\right| + \left|F_{61}\right|\right) + \left(\left|F_{23}\right| + \left|F_{32}\right|\right) + \left(\left|F_{25}\right| + \left|F_{52}\right|\right) +$$
$$+ \left(\left|F_{34}\right| + \left|F_{43}\right|\right) + \left(\left|F_{36}\right| + \left|F_{63}\right|\right) + \left(\left|F_{45}\right| + \left|F_{54}\right|\right) + \left(\left|F_{56}\right| + \left|F_{65}\right|\right) = max \quad . \qquad (4)$$

The unified presentation of network flow activities is shown in fig. 4 for "3 outer/3 inner-pole" 6-node graph, with respect to max flow task statement (1).

	0	1	2	3	4	5	6	∞
0	×	×	f_{020}	×	f_{040}	×	f_{060}	×
1	×	×	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	$f_{\infty 1\infty}$
2	f_{020}	f_{21}	×	f_{23}	f_{24}	f_{25}	f_{26}	×
3	×	f_{31}	f_{32}	×	f_{34}	f_{35}	f_{36}	$f_{\infty 3\infty}$
4	f_{040}	f_{41}	f_{42}	f_{43}	×	f_{45}	f_{46}	×
5	×	f_{51}	f_{52}	f_{53}	f_{54}	×	f_{56}	$f_{\infty 5\infty}$
6	f_{060}	f_{61}	f_{62}	f_{63}	f_{64}	f_{65}	×	×
∞	×	$f_{\infty 1\infty}$	×	$f_{\infty 3\infty}$	×	$f_{\infty 5\infty}$	×	×

Fig. 4 The unified presentation of network flow activities in "3 outer/3 inner-pole" 6-node graph

Based on three elementary max flow indexes F_{ILA}^{max}, F_{IOA}^{max} and F_{OTA}^{max}, the seven different max flow tasks can be formulated, Tab.1.

The seven types of max flow tasks in Tab.1 can be multiplied by four categories of inner and outer channels properties:

1) inner and outer channels have infinite product generation/consumption capacity;
2) inner and outer channels have limited product generation/consumption capacity;
3) inner channel has infinite and outer channel has limited product generation/consumption capacity;
4) inner channel has limited and outer channel has infinite product generation/consumption capacity.

The multiplication of seven classes in Tab. 1 by 4 aforesaid categories resulted in 28 diverse max flow task statements.

Table 1

Classification of max flow tasks

Nr.	Max Flow Task
1	$f_1(F_{ILA}, F_{IOA}, F_{OTA}) = F_{OTA} = max$
2	$f_2(F_{ILA}, F_{IOA}, F_{OTA}) = F_{ILA} = max$
3	$f_3(F_{ILA}, F_{IOA}, F_{OTA}) = F_{IOA} = max$
4	$f_4(F_{ILA}, F_{IOA}, F_{OTA}) = f_4(F_{ILA}, F_{IOA}) = max$
5	$f_5(F_{ILA}, F_{IOA}, F_{OTA}) = f_5(F_{ILA}, F_{OTA}) = max$
6	$f_6(F_{ILA}, F_{IOA}, F_{OTA}) = f_6(F_{IOA}, F_{OTA}) = max$
7	$f_7(F_{ILA}, F_{IOA}, F_{OTA}) = max$

3. Conclusion

In this paper, a brief survey given on known approaches to network max flow task study. Concluded that max flow task is not exhaustively studied yet in general case for flexibly reconfigured multiple-pole weighted graph. Besides, commonly applied graph models do not actually meet real networking technologies. To overcome these issues, basic principles introduced to study the max flow task for an open multi-pole software defined network with flexibly reconfigured channels.

References

1. Ford L.R. Flows in Networks / L.R.Ford, D.R.Fulkerson // A report prepared for United States air force project RAND. – 1962. – 332 p.

2. Swamy M. N. S. Graphs, Networks, and Algorithms / M. N. S. Swamy, K. Thulasiraman. – Wiley, 1981. – 592 p.

3. Ford L. R. Maximal flow through a network / L. R.Ford, D. R.Fulkerson // Canadian Journal of mathematics. – 1956. – N 8(1956). – p. 399-404.

4. Yefim Dinitz . Algorithm for solution of a problem of maximum flow in a network with power estimation: doklady Akademii nauk SSSR. – 1970. – Tom 194, vol. 11. – p. 1277–1280.

5. Math Insight: Undirected graph definition. – Availailable at: http://mathinsight.org/definition/undirected_graph.

6. Edmonds, Jack; Karp, Richard M. (1972). "Theoretical improvements in algorithmic efficiency for network flow problems". Journal of the ACM. Association for Computing Machinery.

7. Goldberg, A. V., Tarjan, R. E. Solving minimum-cost flow problems by successive approximation: proc. of the nineteenth Annual ACM Symp. on Theory of Computing. – New York,1987. – p.7-18.

8. Goldberg A. V. Beyond the flow decomposition barrier / A. V.Goldberg, S.Rao // Journal of the ACM. – 1998. – vol. 45 (issue 5). – p. 783-797. – doi:10.1145/290179.290181.

9. Cameron Musco, Advisor: Dan Spielman. Fast Approximation of Maximum Flow using Electrical Flows – April 18, 2011. – Available at: http://www.cameronmusco.com/personal_site/pdfs/CNMSpringThesis.pdf.

10. Planar graphs (DIMACS series in discrete mathematics theoretical computer science, Volume 9) / Editor W.T. Trotter. – American Mathematical Society, 1993. – 152p.

11. Bela Bollobas. Modern Graph Theory. – New York: Springer Science & Business Media, 1998. – 394 p.

12. Christian Fremuth-Paeger. Balanced Network Flows. I. A Unifying Framework for Design and Analysis of Matching Algorithms / Christian Fremuth-Paeger, Dieter Jungnickel // Networks. – 1999. – vol.3, issue 1. – p.1-28.

13. Tykhonova O.V. The maxflow problem statement for multi-pole software defined network / O.V. Tykhonova, V.I.Tikhonov // XVII international scientific-technical conference "Measuring and computing devices in technological processes". – June 8-13, 2017. – p. 161-164.

14. Tikhonov V.I. Quantum field model of the digital telecommunication channel / V.I. Tikhonov // Tsyfrovi tekhnolohii. – 2011. – N 9. – p.7-19.

15. Tikhonov V.I. Tensor model of digital duplex channel in the integrated telecommunication technology / V.I. Tikhonov // Zviazok. – 2012. – N 3. – p.45–49.

16. Tikhonov V.I. Network structure analysis in topological space / V.I. Tikhonov // Tsyfrovi tekhnolohii. – 2015. – N 18. – p. 56-62.

Рецензія/Peer review : 11.4.2017 р. Надрукована/Printed :22.6.2017 р.
Стаття рецензована редакційною колегією