UDC 681.3

S. Yukhymchuk, Dr. Sc. (Eng.), Prof.; D. Bilous

DETERMINATION OF MAXIMUM FLOW IN DECENTRALIZED NETWORKS

The paper presents the methods and algorithms for determination of maximum flow in decentralized networks, based on the Goldberg – Rao iterative algorithm. There had been given the main assumptions and requirements to the bringing of the maximum flow search task in the decentralized net to the task in the oriented net with the unique source and outflow. To determine the maximum flow there had been suggested the iterated algorithm with the application of binary functions.

Keywords: *decentralized Networks, Goldberg – Rao algorithm, maximum flow.*

Introduction

Decentralized (peer-to-peer or P2P) computer network is the network that is based on equality of components [1]. A pure peer-to-peer network does not have the notion of clients or servers, and each peer node simultaneously functions both as "clients" and "servers" regarding other nodes of the network. Unlike client-server architecture, such arrangement enables the network to keep on functioning at any number and any combination of nodes. Decentralized computer networks have found wide application for organization of file-sharing networks (E-Donkey, BitTorreent), Internet-TV, Internet-radio, multimedia networks etc.

Taking into consideration the rapid growth of the number of P2P in various spheres, determination of P2P of certain structure and configuration functional efficiency becomes scientific problem of paramount importance. Determination of P2P efficiency enables already of the stage of design or in the process of usage to reveal “weak points” of P2P and ways of their elimination, as well as correspondence of the given structure to fulfill the suggested tasks [1, 2].

One of the most efficient technologies used for P2P application is the implementation of distributed computations [1], which allow to carry out enormous volume of computations in relatively short period, that, even in case of using supercomputers would require, depending on the complexity of the problem, tens or hundreds of years [2]. Such performance can be achieved since the global task is divided into great number of blocks, which are solved by hundreds of thousands of computers participating in the project. In such cases the problem to be solved is determination of the efficiency of P2P operation, taking into account its configuration and characteristics of its components. Nowadays there is no generally accepted list of efficiency indices for evaluation of computer network operation. This can be explained, mainly, by various approaches regarding the aim of network creation. Basic efficiency indices are the following [1, 2, 3, 4]:

– traffic-carrying capacity;
– time of message delivery ;
– probability of in-time message delivering;
– reliability and survivability;
– profitableness;
– cost;
– control efficiency.

Taking into consideration the necessity of in-time delivery of information while performing distributed computations, one of the main criteria of P2P operation efficiency as computer network, we may consider the value of maximum data flow, that can be circulated in P2P [2]. Thus, among the problems dealing with determination of efficiency rate and operation efficiency, the most actual one is the problem dealing with determination of maximum flow in P2P already at the stage of design or in the process of operation. The value of maximum flow depends on traffic-carrying capacity of communication channels, hardware resources of units, etc., so it can be considered as a complex index of P2P operation efficiency. Besides, determination of maximum flow value enables to determine the efficiency and expending of usage of communication channels and blocks, that, in its turn, can be considered as the efficiency of the given structure and configuration of P2P.

The problem dealing with calculation of maximum flow in the network is classical problem of optimization; algorithms of its solution have been elaborated during last 50-60 years. Solution methods are applied in transport, communication, electric networks as well as for modeling of various physical and chemical processes. Initial approach to the solution of the problem intended for maximum flow determination was the application of linear programming simplex-method. In 1956 Ford and Fulkerson suggested the applications of oriented network and iterative pseudo-polynomial algorithm for the solution of the given problem that enabled to improve evaluation of algorithm of maximum flow determination from $O(n^2mU)$ (where $n=|V|$ - number of vertexes, $m=|E|$ - number of ribs, $U=\max(C_{ij})$ - maximum value of traffic capacity) to $O(nm\cdot U)$. In the given case $O(g(n))$ is the upper asymptotical evaluation of algorithm complexity and determines the rate of its performance in the worst case. Algorithm performance time $T(n)$ has evaluation $O(g(n))$ if $\exists c>0, n_0>0: T(n)\le cg(n), \forall n>n_0$ [5]. Further investigations allowed to decrease evaluation of the algorithm of maximum flow determination. Chronology of main achievements regarding he solution of maximum flow problem [6] is given in Table.

Table

Chronological table of achievements in solution of maximum flow problem

Year	Author	Running Time
1951	Danzig	$O(n^2mU)$
1956	Ford-Fulkerson	$O(nm^2\cdot U)$
1970	Edmonds-Karp	$O(nm^2)$
1970	Dinic	$O(n^2m)$
1972	Edmonds-Carp	$O(m^2\cdot \log U)$
1973	Diniz-Gabow	$O(nm\cdot \log U)$
1974	Karzanov	$O(n^3)$
1978	Malhotra-Kumar-Malacshwart	$O(n^3)$
1978	Galil-Naamad	$O(nm\cdot \log^2 n)$
1980	Sleator-Tarjan	$O(nm\cdot \log n)$
1986	Goldberg-Tarjan	$O(nm\cdot \log \frac{n^2}{m})$
1989	Cheriyan-Hagerup-Mehlhorn	$O(nm+n^2\log^2 n)$
1992	King-Rao-Tarjan	$O(nm+n^{2+e})$
1997	Goldberg-Rao	$O(\min\left\{m^{\frac{2}{3}}, n^{\frac{1}{2}}\right\} m\cdot\log\left(\frac{n^2}{m}\right)\log U)$

In 1997 Goldberg and Rao proposed algorithm that was giving non-single arc length. Nowadays this algorithm is the most modern and effective. Its asymptotical evaluation exceeded $O(nm)$. Most of the current research are directed on studying and improvement of Goldberg – Rao algorithm.[6,7].

Distributed computing P2P network's structure may constantly change because of nodes' disconnections or accidental faults without losing coherency. That is why there exists the necessity to determine possible value of maximum flow for P2P current configuration.

The aim of the given paper is the elaboration of the algorithm intended for determination of maximum flow in case when, unlike classical algorithms of Goldberg-Roa and Ford-Fulkerson, nodes have limited bandwidth.

Maximum Flow Problem in P2P Networks

P2P Mathematical Model

Oriented weighted graph $G(V,E)$ is suggested to use as P2P mathematical model on the level of morphological description. Graph vertexes V correspond to network nodes, and branches E correspond to communication lines.

Every vertex $v_i \in V$ corresponds to the following set of indices and values: [3, 4]:

– capacity $c(v_i)$;

– probability of node failure $q(v_i)$;

– pool memory $r(v_i)$;

– node cost $s(v_i)$;

– service algorithm U_i.

Every branch $e_j \in E$, that connects node v and node w has such set of characteristics and corresponding values [3,4]:

– length $l(e_j)$;

– measure of unreliability $q(k)$ of k channel(unreliabilities are accepted as independent);

– capacity $t(e_j)$;

– carrying capacity $c(e_j)$;

– probability of error $\xi(k)$ in message when delivering;

– price $s(e_j)$.

Unreliabilities of peers and lines of the P2P network are accepted as independent. Unreliabilities are the results of physical reliability of the components and are independent on rules and level of their usage.

Proposed model describes only primary network as network of communication lines. Secondary network could be built on basis of primary network's channel using crossing on some nodes. In that case it's necessary to supplement primary model with corresponding crossing scheme taking into account more complex dependences *{q(V);V}*. For determination of max flow in network primary model is necessary and sufficient [3]. Primary model considers all necessary network's characteristics in full measure.

Reduction of the max flow problem in P2P network to the classic max flow problem in directed network with single source node and single sink node

In order to use algorithms for determination of max flow in P2P network it is necessary to reduce current problem to the classic max flow problem in directed network with single source node and single sink node. P2P network's features differ from the classical. They are:

1. Several source nodes and several sink nodes

Quantity of nodes of oriented on calculation P2P network could be divided into three groups: S – quantity of source nodes, T – quantity of sink nodes, R – quantity of intermediate nodes. Flow could be oriented from any source node into any sink node. In that case it is necessary to supplement graph model of the network with new nodes S^* - super-source and T^*-super-sink. S^* is connected with all source nodes. T^* is connected with all sink nodes. Capacity of all of the new links are ∞. In that case determination of the max flow from S to T is identical to determination of the max flow from S^* to T^* [6].

2. Limited nodes' capacity

There is possibility of limited capacity of nodes in the P2P network. Such possibility is explained by limited hardware resources in nodes. In that case it's necessary to replace each node $v_i \in V$ with limited capacity $c(v_i)$ with two nodes $v_i^{'}$ and $v_i^{''}$. Two new nodes are connected with link having capacity of $c(v_i)$.

3. Undirected or mixed P2P network

In the case of existence of undirected links in the P2P network max flow on that link (v, w) with capacity $c(v, w)$ should meet the following requirements:

$$\begin{cases} f(v,w) \le c(v,w), \\ f(w,v) \le c(v,w), \\ f(v,w) \cdot f(w,v) = 0. \end{cases}$$

Replacing undirected link with two directed links having equal capacity $c(v, w)$ we reduce the problem to the same problem on the directed graph [7].

Algorithm for the Maximum Flow Problem in the P2P network

For arbitrary case of P2P network the number of links $m = |E|$ greatly exceeds the number of nodes $n = |V|$: $m >> n$. In the borderline case $m = n^2$. Therefore most successful way is using of Goldberg – Rao Algorithm with running time of $O(n^{2\frac{1}{2}} \cdot \log U)$. It's better that running time of Karzanov algorithm ($O(n^3)$) and both modification of Tarjan algorithms($O(n^3 \cdot \log n)$ and $O(n^3)$) [5].

We use the following notation: the network is a directed graph G = (V,E) with a source node s and a sink node t. The capacity u(a) of every arc a is an integer between 1 and U. The network has n nodes and m arcs. Consideration must be given to differences between P2P network's structure and classical network's structure that is used in the classical Goldberg – Rao algorithm. Therefore it's necessary to add to the algorithm some preliminary stage that allows to reduce current max flow problem to the classical max flow problem in the oriented network. Such preliminary stage realization consists of the following steps:

1. Add super-source S^* and super-sink T^* to the model.
2. Replace each node $v_i \in V$ with limited capacity $c(v_i)$ with pair of nodes $v_i^{'}$ and $v_i^{''}$, that are connected with link having capacity $c(v_i)$.
3. Replace each undirected link $e_j \in E$ having capacity $c(e_j)$ with pair of oppositely directed links having capacities $c(e_j)$.
4. As a result modified graph model $G^{'}(V^{'}, E^{'})$ of P2P network is obtained. Modified model contains source node S^* and sink node T^*.

Let function $c(e): E^{'} \to \{1,...,C\}$ determine carrying capacities of communication channels. Function $f(e): E^{'} \to \{1,...,C\}$ will be labeled as flow. The given function satisfies the condition $f(e) \le c(e)$ and principle of flow saving. Remaining carrying capacity will be labeled as $c_f(i, j) = f(i, j) + c(j,i) - c(i, j)$. Arks having positive c_f represent the set E_f. Remaining flow is the difference between a certain optimal flow in P2P f^* and current flow f^R : $\left| f^*(e) - f^R(e) \right|$. Function, where there is no route from S^* to T^* T in graph E_f will be labeled as line end flow.

Function $l: E^{'} \to R^+$ will be labeled as function of arc length. Function $d: V^{'} \to R^+$ will be considered as distance label relatively l if $d(t) = 0$ and for all arcs $e \in E^{'}$ remaining cost is nonnegative. Remaining cost is determined as $l_d(i,j) = l(i,j) + d(j) - d(i)$. Remaining cost on arcs of the route from S^* to T^* can only increase. Let $d_l(i)$ be distance of the vertex i to t relatively l. For any d : $\forall d, d \le d_l$ [6, 8].

Using arc length function we can obtain the upper limit of remaining flow. Link from E_f can be considered as the channel of $c(e)$ width, $l(e)$ length and $Ar(e) = c(e)l(e)$ area. The area of the network Ar_{Ef} will be the sum of areas of all links with positive remaining capacity [6].

Let each unit of the remaining flow require $d(s)$ units of area, than the upper limit of remaining flow's value is $\frac{Ar_{Ef}}{d(s)}$. Let the link (i,j) from E_f be called as admissible, if $d(i) > d(j)$ or $d(i) = d(j)$ when $l(,i,j) = 0$. The set $A(f,d,l)$ is formed from admissible links and graph $G_A(V', A(f,d,l))$ is composed.

Goldberg – Rao algorithm is based on binary function of arc l' length, which equals 0 on links with "large" remaining carrying capacity and equals 1 – on others. This allows decreasing Ar_{Ef} and obtaining better estimation for remaining flow as compared with single function of length. Determination of "large" remaining carrying capacities depends on current value of remaining flow.

At the every iteration of the algorithm current value of remaining flow F and corresponding value ΔF (the value by which F is to be increased at the given step of the algorithm) are determined. Binary length function $l(e)$ equals 0 if $c_f(e) \ge 3\Delta F$ and 1 – in opposite case. The length of some links changes after flow increase in the network by means of line end flow [8].

At the beginning of each iteration all components of strong connectivity, formed by links of zero length, are eliminated from graph G_A and the flow increases in obtained non-cycle graph for which either line and flow or flow of ΔF value is found.

Unlike classical Goldberg – Rao algorithm the given paper suggests to use algorithm of maximum flow search for determination of ΔF instead of binary search using network section, described in [6]. Such replacement enables to increase operation speed of the algorithm due to reduction of the number of computational operations.

In general case operation of the algorithm at *i-th* iteration can be presented in the form of the following sequence:

While $F \ge 1$

1. Parameter ΔF, binary function of l' length and labels of distances d are updated.
2. Components of string connectivity, formed by links of zero length, are eliminated. Elimination is performed only within the limits of current iteration, at the next iteration links are restored.
3. Graph $G_A(V', A(f,d,l))$ of admissible links is determined.
4. Flow in G_A which is either line end or equals ΔF is determined.
5. Current flow is increased at the expense of the flow that was determined at the previous step
6. F is updated.

End.

The search of line end flow is performed by any of the existents methods. If the value of obtained flow $H > \Delta F$, then the excess is eliminated [6].

The suggested modifications of Goldberg – Rao algorithm mean its adaptation to specific structure of P2P as compared with conventional network having a single source and a single sink and optimization of ΔF variation determining with using network section instead of binary search. Suggested modifications don't influence the convergence, thus the convergence of modified algorithm can be proved in much the same way as the convergence of the original Goldberg – Rao algorithm, considered in [8].

Conclusions

The given paper considers the method intended for evaluation of maximum network flow in P2P. Algorithm of P2P maximum flow problem solution based on Goldberg-Roa algorithm is suggested. Asymptotic estimation of the suggested algorithm is $O(n^{2\frac{1}{2}} \cdot \log U)$.

Improvements of the algorithm are possible due to the application of parallel computations theat enables to decrease the algorithm estimation to $O(n)$.

Authors plan to elaborate the program package intended for realization of P2P behavior modeling and allowed to determine numerical values of P2P maximum flow. Proceeding from intermediate values of algorithm operation it is possible to reveal “weak” points of P2P, cases of non-efficient usage of nodes resources or channels etc.

REFERENCES

1. Cisco Systems. Руководство по технологиям объединенных сетей. – М: Вильямс, 2004 – 1039 с.
2. Менаске Д., Алмейда В. Производительность web-служб. Анализ, оценка и планирование. – М: Diasoft, 2005 – 467 с.
3. Ченцов В.М. Системы распределения информации. Синтез структуры и управления – М:Связь, 1980 – 144 с.
4. Білоус Д.А., Юхимчук С.В. Комплексний критерій оцінювання ефективності функціонування обчислювальних мереж. // Труды восьмой международной научно-практической конференции «Современные информационные и электронные технологии, Одеса, 2007 – с.117.
5. Кормен Т. Алгоритмы: построение и анализ. – М: МЦНМО, 2002 – 957 с.
6. Hudson I. The Maximum Flow Problem – Denison University, 2004 – 19 с.
7. Касьянов В.Н., Евстигнеев В.А. Графы в программировании: обработка, визуализация и применение. – Санкт-Петербург: «БХВ-Петербург», 2003 – 1104 с.
8. Papp D. The Goldberg – Rao Algorithm for the Maximum Flow Problem. – COS, 2006 – 5 с.

Serhiy Yukhymchuk **–** Head of the Department;

Dmytro Bilous **–** Master student of the Department.
The Department for Intellectual Systems, Vinnytsia National Technical University