Theoretical and Applied Informatics
ISSN 1896–5334
Vol.23 (2011), no. 3-4
pp. 163–176
DOI: 10.2478/v10179-011-0011-y

Optimization of Links Cost for Unicast and Anycasttraffic

JAKUB GŁADYSZ

Chair of Systems and Networks Computer Networks
Wroclaw University of Technology
ul. WybrzeżeWyspianskiego27, Wroclaw, Poland
jakub.gladysz@pwr.wroc.pl

Received 5 September 2011, Revised 28 September 2011, Accepted 1 October 2011

Abstract: This work presents optimization model and computational results of Capacity and Flow Assignment Problem for multilayer networks with unicast and anycast traffic. Capacity of each channel is expressed in a set of link proposal. Anycast is a network addressing and routing methodology in which datagrams from a single sender are routed to the topologically nearest node in a group of potential receivers all identified by the same destination address. We propose two heuristic algorithms based on Flow Deviation and Tabu Search method. The results of algorithms will be compared with optimal solution obtained using CPLEX package. To improve execution time of exact algorithm we introduce cut inequalities. Cut inequalities are added to the optimization problem, enabling the branching phase to use this information in calculation of more effective bounds. Next, we want to examine testing networks depend on different percentage of anycast traffic, number of distribution centers (servers or replicas) and the different size of network (number of nodes, links, routes).

Keywords: unicast, anycast, capacity, CFA, heuristic algorithms

1. Introduction

Due to increasing number of people using the Internet and due to growing flows of Information, the mechanism of designing computer networks is a matter of great importance. Nowadays effective and efficient transmission of information is vital to economy and society. Additionally, telecoms every year invest tens to hundreds of millions dollars in network infrastructure, research opportunities in network theory, algorithms and applications.

The paper centers around the Capacity and Flow assignment problem for simultaneous unicast and anycast kind of traffic. This problem arises from the growing popularity of network services using anycast flows (e.g. Domain Name Service (DNS), peer-to-peer (P 2P) systems, Content Delivery Networks (CDN), video streaming, software distribution, sensor networks) [3], [6], [10]. Anycast paradigm is a new technique to deliver packets in computer networks which was implemented in Internet Protocol version 6.0. Itis the point-to-point flow of packets between a single client and the "nearest" destination server. The idea behind anycast is that a client wants to download or send packets to any one of several possible servers offering a particular service or application [5]. Since the same information is replicated in many replica servers located in the network, the user can select one of these servers according to some criteria including also QoS parameters. Consequently, the anycasting can reduce the network traffic and avoid congestion causing big delays in data delivery. One of the most popular caching technology is the Content Delivery Network (CDN) defined as mechanisms to deliver a range of content to end users on behalf of origin Web servers. The original content is offloaded from source sites to other content servers located in different locations in the network. Each request is redirected to one of CND replica servers offering the requested Web page. The CDN delivers the content from the origin server to the replicas that are much closer to end-users. Thus, the CDNs' servers can approach the hit ratio of 100%. It means that almost all request to replicated servers are satisfied. Akamai is an example of a CDN system used in the Internet.

In this paper we consider Capacity and Flow Assignment Problem Simultanously for unicast and anycast flows for multicommodity, non-bifurcated flow. As commodity we understand a set of packets, which have the same origin and destination nodes. Non-bifurcated flow routing between two location is exactly along one way. In the following paper we use unicast and anycast flow, which were described in work [9]. Unicast connections are defined as triple: origin node, destination node and bandwidth requirement. Anycast is defined as client node and bandwidth requirement to and from server. In anycast flow dates are kept in many locations in the network (servers or replicas). To calculate routes for particular user first we have to select server and then choose connection between user and server and vice versa. Two connections associated with the same demand are denoted as associated connection. If a is a connection from user to server, then $\tau(a)$ is connection from server to user and vice versa.

Multi-layer modeling across the network layers (at least 2) consists of different technologies, protocols and functionalities (e.g. IP oved DWDM, IP over ATM over SONET etc.). In the literature there are many research works touching optimization of single-layer networks. Because in single-layer method each layer is optimized separately, global optimality of the solution cannot be guaranteed.

As a model of multi-layer network we propose two-layers, based on MPLS over DWDM architecture. In our work we are not given capacity of physical links and the main goal of our paper is to minimize cost function of operating upper and lower layer.

In the upper MPLS layer we consider two kinds of demands: unicast and anycast. The MPLS layer consists of nodes – MPLS routers and logical links which are formed from lightpaths. The idea of multi-layer network modeling is as follows. The links of an upper layer are constructed using paths of the lower layer, and this approach repeats going down the resources hierarchy. Logical links given by paths allocated to demands of the upper layer represent the demand pattern of the lower layer. Wavelength Division Multiplexing (WDM) is a technology, which multiplexes multiple optical carrier signals on a single optical fibre by using various wavelengths (colours) of laser light to carry different signals. Since a single fiber using WDM can carry many (e.g. 128) wavelengths 40 Gb/s each, the overall capacity of a fiber grows to terabits per second. In our work the lower WDM layer is responsible for physical transmission –in this layer we assume only one-to-one connection which are realized logical links in upper layer.

To find optimal solution of our problem we use CPLEX linear programming package [4]. Because the considered problem is NP-complete - with a growing network structure, finding an optimal solution may be difficult in sensible time. Therefore in this work we focus cut inequalities that can be applied in construction of branch-and-cut algorithm. We suppose that introducing cut-inequalities to our model can reduce the execution time of branch-and-cut algorithm included in CPLEX solver. Because these problems are NP-hard, therefore optimal solutions of branch-and-bounds or branch-and-cut methods can be generated for relatively small networks. Consequently, we propose new heuristic algorithms based on Flow Deviation, TabuSearch method.

Above algorithms for modular link cost in each layer was formulated in work [2]. In our problem we are given a set of link proposal and we must select only one of them. This formulation denotes the situation that many telecom operators offer price list including links with decreasing costs of unit capacity with the increase of overall link capacity. For instance, link 10 Mb/s costs 1000 USD, while link of capacity 30 Mb/s costs 2000 USD. Note that in the model with modular links the cost of link 30Mb/s would be 3000 USD. The decreasing link cost is introduced in MPLS upper layer, in DWDM layer the physical layer is realized using modular link cost.

The remainder of this paper is organized as follows. In the next section we present optimization models of CFA problem. In section 3 we formulate cut inequalities. Section 4 includes description of the Flow Deviation and Tabu Search heuristic algorithm. Section 5 includes results of numerical experiments. The last section concludes this work.

2. Optimization model

The network model addressed in this paper is a two layer model: MPLS over DWDM.The network is modeled as a graph consisting of nodes and links. Nodes represent network devices like routers, switches in upper layer or optical cross-connects in lower layer. The lower layer – optical transport layer applying DWDM – consists of nodes represented by optical cross-connects (OXCs) that perform wavelength routing operations and optical links – fibers. The upper layer – MPLS layer – includes nodes represented by MPLS routers, namely label switching routers (LSR).A set of lightpaths (wavelengths) provisioned by DWDM layer forms a logical topology for the MPLS routers. i.e. lightpaths represent in MPLS layer. In Fig. 1 we show a simple example to illustrate MPLS over DWDM architecture.The logical link between LSR1 and LSR4 consists of two lightpaths (wavelengths). However, these two lightpaths are routed in two various paths in the DWDM layer: OXC4-OXC5-OXC1 and OXC4-OXC3-OXC6-OXC2-OXC1.

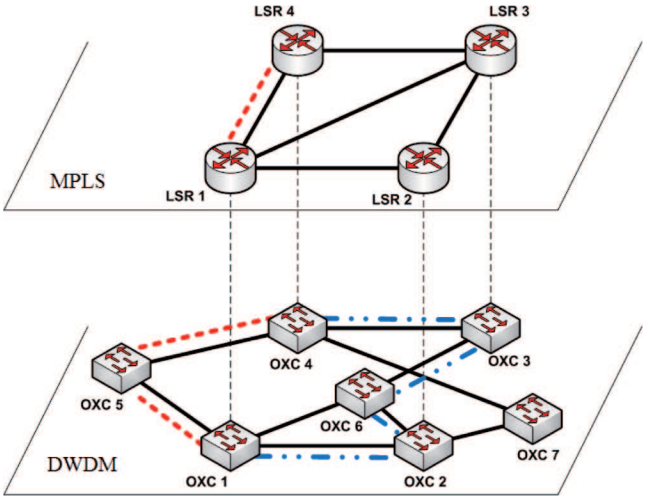

Fig. 1. MPLS over DWDM architecture

Links e = 1,2,$\dots$,E denote logical links of MPLS network or physical links g = 1,2,$\dots$,G of lower layer, e.g. fibers. We are given the cost of link type for link$e$ in upper layer, cost of one module capacity on link g in lower layer and set of connections denoted by index d = 1,2,$\dots$,D. We assume that connections a = 1,2,$\dots$,A are anycast connections including both upstream and downstream connections. If connection a is an upstream (downstream) connection, connection $\tau(a)$ denotes the associated downstream (upstream) connection. Connections u = A + 1,$\dots$,D are unicast. Notice that to distinguish anycast and unicast connection we use indices a and u, respectively. Let h_d denote the volume of connection d = 1,2,$\dots$,D.

The considered problem can be formulated as follows:

given:	network topology of both layers, localization of servers, set of anycast and unicast demands,set of routes,
minimize:	function of cost
over:	flow and capacity
subject to:	all connections are established, summary flow in each arc can not be bigger than its capacity

To mathematically represent the problem we use notation which was proposed in [7]. Our problem can be recorded following:

indices

d = 1,2,...,D	demands (both unicast and anycast)
a = 1,2,...,A	anycast demands
u = A + 1,...,D	unicast demands
p = 1,2,...,P_d	candidate paths for demand d
e = 1,2,...,E	network links
q = 1,2,...,Q_e	candidate paths for link $e$in lower layer
g = 1,2,...,G	network links in upper layer

constants

δ_{edp}	= 1 if link e belongs to path p realizing demand d, 0 otherwise
γ_{geq}	= 1 if link g belongs to path q realizing capacity of link e, 0 otherwise
h_d	volume of unicast demand d
k_e	cost of one capacity module
k_{el}	cost of link type k for link e
M	size of the link capacity module
N	size of the link capacity module in lower layer
$o(a,p)$	origin node of path x_{ap}
$t(a,p)$	destination node of path x_{ap}
$\tau(a)$	index of the anycast upstream (downstream) demand associated with anycast downstream (upstream) demand a

variables:

x_{dp} = 1 if route $p$is chosen to realize demand d, 0 otherwise (binary)
y_e = 1 if link of type l is selected for link e, 0 otherwise (binary)
z_{eq} number of paths $q$selected to realize capacity of link e, 0 otherwise (integer, non-negative)
u_g number of modules N to be installed on link $g$in the lower layer (integer, non-negative)

Our problem can be formulated below:

objective:

min $\sum_e \sum_l k_{el} y_{el} + \sum_g \kappa_g u_g$

constraints:

$$\sum_p x_{dp} \;=\; 1 \quad d \;=\; 1, 2, \ldots, D \tag{1}$$

$$\sum_{\mathrm{d}} \sum_{\mathrm{p}} \delta_{\mathrm{edp}} \mathrm{x}_{\mathrm{dp}} \mathrm{h}_{\mathrm{d}} \leq \sum_{\mathrm{l}} \mathrm{c}_{\mathrm{el}} \mathrm{y}_{\mathrm{el}} \mathrm{e} = 1, 2, \ldots, \mathrm{E} \tag{2}$$

$$\sum_l y_{el} = 1 e = 1, 2, \ldots, E \tag{3}$$

$$\sum_p x_{ap} o(a,p) = \sum_p x_{\tau(a)p} t(a,p) a = 1, 2, \ldots, A \tag{4}$$

$$\sum_q z_{eq} = \sum_l c_{el} y_{el} e = 1, 2, \ldots, E \tag{5}$$

$$\sum_e \sum_q \gamma_{geq} z_{eq} \leq N u_g g = 1, 2, \ldots, G \tag{6}$$

The objective is to minimize the cost of operating two layer. Constraint (1) guarantees that only one route can be chosen for one connection (one for unicast connection, one for anycast connection upstream and one for anycast connection downstream) in upper layer. Condition (2) states that flow in each arc can't be bigger than its capacity. Condition (3) assures that only one value of capacity can be selected for each link. Equation (4) guarantees that two routes associated with the same anycast demand connect the same pair of nodes. (5) is a demand flow constraint, i.e. the upper link is realized by a set of lower layer paths. Condition (6) states that flow in each link in lower layer can't be bigger than its capacity.

3. Exact algorithm

To find optimal solution linear programming package CPLEX 10.2.0 will be used[4]. It calculates feasible initial selection and then uses Branch-and-Cut method improves founding solution. Because considered Problem is NP-complete in not all cases finding optimal solution in acceptable through time 3600 s is possible. The solution will be treated as feasible and will be compared with heuristic algorithms. in this point we focus cut inequalities which are added in the root node of the solution tree. Our cut inequalities uses different properties of unicast and anycast flows and the upper bound of function of cost. The cut inequalities for linear function of cost in single-layer network first was formulated in [1]. In this work we introduce cut inequalities in MPLS over DWDM network and candidate link type of each channel.

Let E denotes the set of all links in the network, V is a set of all nodes in the network, *VR* denote the set consisting all nodes that have a replica server and *VC* denote the set of all other nodes, i.e. V = *VR*$\cup$*VC*, $o(e)$ and $t(e)$ denote the origin and destination node of link e and $o(d)$ and $t(d)$ denote the origin and destination node of demand d. Notice that in the context of anycast demands the origin (destination) node of upstream (downstream) connection is a single node. The destination (origin) nodes of upstream (downstream) connection is defined as a set of replica nodes *VR*. We assume that σ (*V′*) is a set of all links included in the cut between sets *V′* and $V \setminus$ *V′*

$$\sigma(V') = \{e \in E : o(e) \in V', t(e) \in V \setminus V'\}$$

Let h(*V′*,V_{J}) is the amount of flow related to demands originating in one of nodes of *V′* and terminating in one of nodes of V_{J}.

$$h(V', V_{\text{J}}) = \sum_{d:o(d) \in V', t(d) \in V_{\text{J}}} h_d$$

For instance, h(*VC*,*VR*) denotes the overall demand from client nodes to replica nodes taking into account both unicast and anycast demands.

To formulate the first pair of cut inequalities connected with anycast paradigm we propose following inequalities (one for upstream and one for downstream traffic):

$$\sum_{e \in \sigma(VR)} \sum_{l} k_{el} y_{el} \geq h(VR,\, VC) \qquad (7)$$

$$\sum_{p} x_{dp} = 1 d = 1, 2, \ldots, D \qquad (8)$$

If we consider inequality (7) we can notice that the cut σ (*VR*) must carry the whole anycast upstream traffic, since the one of replica servers included in *VC* must be selected for each anycast demand. Furthermore, we also take into account unicast traffic. The left-hand side of (7) is the overall link capacity necessary to be installed on links included in σ (*VR*) to satisfy the traffic coming from nodes included in *VR* to nodes included in *VC* (right-hand side). Constraints (7) and (8) are versions of inequality for anycast paradigm. We refer to (7) as to *upstreamreplica inequality* (URI) and (8) as *downstreamreplica inequality* (DRI).

Next inequalities including summary demand outgoing and incoming node v we can formulate following:

Let $\varepsilon^{+}(v) = \{e \in E : o(e) = v\}$ denote the links outgoing node v and $\varepsilon^{-}(v) = \{e \in E : t(e) = v\}$ denote the links incoming node v.

Let $h^{+}(v) = \sum_{d\,:\,o(d)\,=\,v} h_d$ and $h^{-}(v) = \sum_{d\,:\,t(d)\,=\,v} h_d$ denote the summary demand outgoing and incoming node v, respectively.

We consider for each node $v \in V$ the outgoing and incoming demand. The final inequalities for NPI (*node partitioninequality*) we can record following:

$$\sum_{e \in (\varepsilon+(v))} \sum_{l} k_{el} y_{el} \geq \lceil h^{+}(v) \rceil v \in V \tag{9}$$

$$\sum_{e \in (\varepsilon\ -(v))} \sum_{l} k_{el} y_{el} \geq \lceil h^{-}(v) \rceil v \in V \tag{10}$$

The last inequality uses heuristic algorithm based on Flow Deviation or Tabu Search method. The cut inequalities, referred to as *upper bound* (UB), is formulated following:

$$\sum_{e} k_e y_e \leq HEU \tag{11}$$

where *HEU* denotes the value of objective function given by the heuristic.

4. Heuristics algorithms

In this point we show two heuristic algorithmsdeveloped for the our optimization problem. The main idea of the algorithms are based on the Flow Deviation and Tabu Search method. Both of them for modular links were presented in work [2]. In our optimization problem we change modular links to candidate lint type in upper layer.

4.1. Flow Deviation Algorithm

Let r be the algorithm's iteration number. Let $\mathbf{f}_r = [f_{1r}, f_{2r}, \ldots f_{Er}]$ denote a vector of link flows calculated according to variables x_{dp} obtained in the iteration r of the algorithm. Furthermore, we assume that $\mathbf{g}$ and $\mathbf{v}$ denote vectors of link flows, $K(\mathbf{f}_r)$ denote the first part of objective function $(\sum_e \sum_l k_{el} y_{el})$ and $k_e(\mathbf{f}_r)$ denote metric of link e in r iteration, B denote set including connections (unicast or anycast)that have not been already processed and $SR(\mathbf{f}_r)$ denote set of shortest path for each connection under metric $k_e(\mathbf{f}_r)$ in upper layer. We assume that s is the algorithm's iteration number in lower layer. For each links of upper layer we have given set of lightpaths in lower layer. Let s be in $\mathbf{h}_s = [h_{1s}, h_{2s}, \ldots h_{Gs}]$ be a vector of link flows calculated according to variables z_{eg} obtained in the iteration s of the algorithm for capacity of links in upper layer. Let $\mathbf{c}$ and $\mathbf{n}$ denote vectors of link flows in lower layer and $|\mathbf{c}| = |\mathbf{v}| = G$, $K(\mathbf{h}_s)$ denote the second part of objective function ($\sum_g \kappa_g u_g$) and $\kappa_g(\mathbf{h}_s)$ denote metric of link g in s iteration, D denote set including lightpaths that have not been already processed and $SR(\mathbf{h}_s)$ denote set of shortest path for each connection under metric $\kappa_g(\mathbf{h}_s)$. We can write our algorithm using the following steps:

Step1 Set $r = 1$. Compute initial selection $\mathbf{f}_1$ and metric $k_1(\mathbf{f}_1)$. For each link e set $k_1(\mathbf{f}_1) = k_{e1}$. Then for unicast and anycast connection compute set $SR(\mathbf{f}_1)$ using metric $k_1(\mathbf{f}_1)$. For anycast demands, condition (4) must be satisfied. Then, for vector $\mathbf{f}_1$ find capacity of links in upper layer. Condition $c_{el} \geq f_{e1}$ must be satisfied. Go to step (3).

Step2 Calculate metric $k_e(\mathbf{f}_r)$. For each link e set $k_e(\mathbf{f}_r) = \min\{k_{el}\}$ where $c_{el} \geq f_{er}$. Then for unicast and anycast connection compute set $SR(\mathbf{f}_r)$ using metric $k_e(\mathbf{f}_r)$. For anycast demands, condition (4) must be satisfied.

Step3 Let $\mathbf{g} = \mathbf{f}_r$

a) Find $\mathbf{v}$ from $\mathbf{g}$ by deviating flow of connection unicast $d = A + 1, \ldots, D$ or anycast $d = 1, 2, \ldots, A$ and $\tau(d)$ to the shortest path included in$SR(\mathbf{f}_r)$. Path for other connections remain unchanged. Compute $K(\mathbf{v})$. Go to Step (5).

b) If $K(\mathbf{v}) < K(\mathbf{g})$ then set $\mathbf{g} = \mathbf{v}$.

c) If $B = \emptyset$o to step 4. Otherwise set $B = B -- \{d\}$ for unicast or $B = B - \{d, \tau(d)\}$ for anycast connection and go to step 3a.

Step4 If $\mathbf{g} = \mathbf{f}_r$ stop Algorithm. Since the solution cannot be improved. Otherwise set $r = r + 1$ and $\mathbf{f}_r = \mathbf{g}$ and go to step (2).

Step5 Compute initial selection $\mathbf{h}_1$ and metric $\kappa_1(\mathbf{h}_1)$. For each link g set $\kappa_1(\mathbf{h}_1) = \kappa_g u_g$. Condition $\mathrm{N}u_g \geq h_{g1}$must be satisfied. Then for each link of upper layer compute set $SR(\mathbf{h}_1)$ using metric$\kappa_1(\mathbf{h}_1)$. Go to Step 7.

Step6 Calculate metric$\kappa_g(\mathbf{h}_s)$. For each link g set $\kappa_g(\mathbf{h}_s) = \min\{_g\}$ where $\mathrm{N}u_g \geq h_{gs}$. Then for each links of upper layer compute set $SR(\mathbf{h}_s)$ using metric$\kappa_g(\mathbf{h}_s)$.

Step7 Let $\mathbf{c} = \mathbf{h}_s$

a) Find $\mathbf{n}$ from $\mathbf{c}$ by deviating flow of link of upper layer e to the shortest lightpath included in$SR(\mathbf{h}_s)$. Path for other links remain unchanged. Compute $K(\mathbf{n})$.

b) If $K(\mathbf{n}) < K(\mathbf{c})$ then set $\mathbf{c} = \mathbf{n}$.

c) If $D = \emptyset$o to step 7. Otherwise set $D = D -- \{e\}$ and go to step 7a.

Step8 If $\mathbf{c} = \mathbf{h}_s$ set $K(\mathbf{v})$=$K(\mathbf{v})$+$K(\mathbf{c})$ and go to Step 3b. Since the solution cannot be improved. Otherwise set $s = s + 1$ and $\mathbf{h}_s = \mathbf{c}$ and go to step 6.

4.2. Tabu Search Algorithm

In this point we propose a new Tabu Search (TS) heuristic algorithm applied for the CFA problem with simultaneous unicast and anycast flows in two-layer, MPLS over DWDM architecture. In this algorithm we extended problem about optimization of lower layer. The algorithm for lower layer uses FD Algorithm, which was described in previous point.

Our Tabu Search method uses the provided initial solution as a starting point and tries to improve this solution. First, using Top-Down and SRP Algorithm we obtain an initial, feasible solution. We assume that r denotes number of algorithm's iteration. Let *IS* denote the initial solution, *CS* denotes the current solution, *BS* denotes the best solution and let *NS* denote the neighborhood which is a set of neighbors N. As a neighbor we assume the solution similar to the current solution – it means that for each current solution the neighborhood is defined as a set of solution which has different selected paths for unicast or anycast demands. Let $K(CS)$ denote a value of the objective (1) for the current solution *CS*. Our algorithm has input three parameters: number of iteration R, size of tabu-list L and maximal number of iterations with no improvement to the best solution K. The length of the tabu-list is an important parameter of our algorithm and it states for how many iteration given move (selected in last iteration unicast or anycast demand) is forbidden. To implement the tabu-list we use short-term memory. To evaluate each move we use the objective (1), i.e. if we switch the path of a selected demand and calculate the new objective function, which is compared against the previous value of the $K(Z)$. The algorithm terminates according to one of two stopping conditions: (1) R iterations were run, (2) in subsequent K iterations there was no improvement of the solution.

The main idea of Tabu Search algorithm can be described using the following steps.

Step. 1. Find an initial solution *IS*. Set *CS* = *IS* and *BS* = *IS*, $r = 0$. Calculate $K(BS)$ as a cost of upper and lower layer (1). Go to step 2.

Step. 2. Set $r = r + 1$. Calculate *NS*: for each unicast and anycast demand which are not on the tabu-list change path included in *CS* (for anycast both upstream and downstream connection). Then for each change of path calculate $\sum_l k_{el} y_{el}$ for each link e Then for each link in upper layer select light-path in lower layer using FD Algorithms (Steps 5 – 8 in previous section). Calculate $\sum_g \kappa_g u_g$ and objective (1). Go to Step 3.

Step 3. Select N from *NS* with minimal value of objective (1). If $K(N) < K(BS)$, then *BS*=N, $K(BS) = K(N)$, $k = 0$ and put demand in selected N on tabu-list. Otherwise set $k = k + 1$. Go to Step 4.

Step. 4. If $r = R$ or $k = K$ then stop algorithm. Otherwise go to Step 2.

5. Computational results

In this section we present some results illustrating our approach. We run tests on the network topology Polska, Germany, Atlanta [8]. For example Germany network has 17 nodes and 26 bidirectional links. We generate 119 unicast and 13 anycast demands. In Table 1 we show main parameters of the network and number of demands.

Network	**Nodes**	**Links**	**Unicast**	**Anycast**
Upper Layer				
Polska	9	20	50	9
Germany	17	52	119	13
Atlanta	26	82	234	22
Lower Layer				
Polska	12	36	–	–
Germany	20	74	–	–
Atlanta	32	122	–	–

Tab 1. Parameters of tested networks

To verify the performance of proposed cuts we run experiments on the networks using the CPLEX solver. In Table 2 we report the execution time of the CPLEX solver algorithm for 6 variants (columns 3-8): without cuts, with the URP inequality (7), with the DRP inequality (8), with the NP cut (9)-(10), with the upper bound on objective function (11) and with all cuts. The last column shows value of the objective function. For each network was carried out tests for different number of replicas and for different proportion of unicast and anycast.

We can watch that the application of all cuts reduces the execution time by 9.15% on average. However, in the case of the upper bound cut the average gap is even larger, i.e. 14.91%.

	Execution time [s]							
Number of replicas	Unicast /anycast proportion	No CUT	URP CUT (7)	DRP CUT (8)	NP CUT (9)-(10)	UB CUT (11)	All CUTS	Value
2	70%/30%	832.17	798.47	767.59	763.27	564.59	590.98	65 832
3	70%/30%	956.50	941.03	967.05	857.50	811.64	870.72	62 014
4	70%/30%	1154.53	1132.43	1098.34	1054.32	911.54	943.23	60 085
2	80%/20%	857.50	835.56	845.64	716.97	760.14	724.19	67 536
3	80%/20%	1232.76	1102.54	1135.22	1219.69	1144.59	1177.78	64 926
4	80%/20%	1282.55	1312.35	1202.99	1294.73	1108.74	1216.88	62 796

Tab 2. Performance of cuts inequalities

The next goal of experiments was to evaluate the Flow Deviation and Tabu Search approach against optimal solutions. In Table 3 we report the comparison between optimal results and Flow Deviation and Tabu Search for various scenarios in terms of the number of replicas and unicast/anycast proportion. The average gap of TS to optimal results is 1.74% and average gap of FD to optimal is 8,16%. We can notice that execution time of CPLEX package is between 1200 – 3600 seconds, execution time of FD is between 5 and 10 seconds and for TS time is about 30-60 seconds.

Number of replicas	Optimal	FD	Gap	TS	Gap
70% unicast/30% anycast					
2	65832	70 664	7,34%	67 221	2,11%
3	62 014	67223	8,40%	62 900	1,42%
4	60 085	65570	9,13%	61515	2,38%
80% unicast/20% anycast					
2	67 536	72223	6,94%	68879	1,99%
3	64 926	70 223	8,16%	66 438	2,33%
4	62 796	67 681	7,78%	64 372	2,51%

Tab 3. Average gaps of FD and TS to optimal results

6. Conclusions

In this work we have focused on the problem of joint optimization of flows and link capacity for networks with simultaneous anycast and unicast flows for two-layers, based on MPLS over DWDM architecture. First we have developed and tested cut inequalities for the problem. Second, we have presented and verified an heuristic algorithms. The results of cut inequalities and heuristic algorithms were presented in section 3. In the future we plan to improve performance the heuristic algorithm and propose new cut inequalities.

Acknowledgements

This work is supported by the European Social Fund according to the Operational Programme Human Capital‘National Cohesion Strategy‘.

References

1. J. Gładysz, K. Walkowiak: *Analysis of cut inequalities for the uncapacitated network design problem with simultaneous unicast and anycast flows*, Advanced simulation of systems, Proceedings of the [31st International] Autumn Colloquium, MARQ, 2009. s. 47-52.
2. J. Gładysz, K. Walkowiak: *Design of MPLS over DWDM architecture for unicast and anycastflows*, Lecture Notes in Computer Science. 2010, vol. 6294, s. 172-183.
3. M. Hofmann, L. Beaumont: *Content networking :architecture, protocols, and practice*, Morgan Kaufmann, San Francisco, 2005.
4. ILOG CPLEX 11.0 User's Manual, France, 2007.
5. Ch. Metz:*IP Anycast Point-to-(Any) Point Communication*, IEEE Internet Computing, March-April 2002, pp. 94-98.
6. G. Peng: *CDN: Content Distribution Network, Technical Report,* sunysb.edu/tr/rpe13. ps.gz, 2003.
7. M. Pioro, D. Medhi: *Routing, Flow, and Capacity Design in Communication and Computer Networks,* Morgan Kaufmann Publishers, 2004.
8. SNDLIB Library of test instances for Survivable fixed telecommunication Network Design. http://sndlib.zib.de/home.action (online) 2008.
9. K. Walkowiak: *Anycast Communication – A new Approach to Survivability of Connection-Oriented Networks*, Communications in Computer and Information Science, Springer Verlag, pp. 378-389.
10. J. Wu: *Theoretical and Algorithmic Aspects of Sensor*, Ad Hoc Wireless and Peer-to-Peer Networks, Auerbach Publications 2006.

Optymalizacja kosztu łączy kandydujących dla połączeń unicast oraz anycast

Streszczenie

Poniższa praca prezentuje model optymalizacyjny oraz eksperymenty obliczeniowe dla problemu jednoczesnego wyznaczania przepustowości kanałów oraz przepływów unicast oraz anycast. Jako przepustowości kanałów użyte zostaną tzw. przepustowości kandydujące – spośród dostępnych przepustowości w danym kanale wybieramy dokładnie jedną. Takie rozwiązanie przyjęte zostanie w górnej warstwie. W dolnej warstwie będziemy rozważać przepustowości modularne – przepustowość kanału wyrażona jest w ilości modułów potrzebnych do zainstalowania w łączu. Anycast jest nowym rodzajem przepływów w sieciach komputerowych, możliwym do zastosowania w szóstej wersji protokołu IP. Jest to transmisja jeden do wielu, w której użytkownik może wysłać/pobrać dane do jednego spośród serwerów w sieci oferujących daną

usługę. W pracy zaproponowane zostały dwa algorytmy heurystyczne. Pierwszy oparty jest o metodę FlowDeviation, drugi na zaproponowanej przez Glovera metodzie Tabu Search. Oba algorytmy zostały wcześniej zaproponowane i opisane przez autora dla przepustowości modularnych. Do znalezienia rozwiązań optymalnych zostanie użyty pakiet programowania liniowego CPLEX. Rozważany problem jest problemem NP.-zupełnym. Oznacza to iż dla dużych sieci komputerowych znalezienie rozwiązania optymalnego może okazać się niemożliwe. Z tego powodu do badanego problemu wprowadzone zostały tzw. funkcje odcinające. Zadaniem funkcji odcinających jest zmniejszenie przestrzeni dopuszczalnych rozwiązań, a co za tym idzie skrócenie czasu poszukiwania rozwiązania optymalnego. Do konstrukcji odpowiednich funkcji odcinających wykorzystywane są właściwości badanego problemu. Zaproponowane funkcje odcinające oraz algorytmy heurystyczne zostały przebadane dla trzech sieci komputerowych. Są to sieci komputerowe o różnej topologii, różnej liczby węzłów oraz połączeń pomiędzy węzłami. Badania miały na celu zbadanie wpływu ruchu anycast w sieci, porównanie czasu rozwiązań optymalnych z zastosowaniem funkcji odcinających oraz ocenę algorytmów heurystycznych. Wyniki przeprowadzonych eksperymentów pokazują, iż zastosowanie przepływów anycast (kosztem unicast) zmniejsza sumaryczny przepływ w sieci przy takim samym strumieniu danych wprowadzanych do sieci. Można to zaobserwować porównując proporcje przepływów unicast oraz anycast. W przypadku badań dotyczących funkcji odcinających można zaobserwować zmniejszenie czasu poszukiwania rozwiązania po dodaniu ograniczenia dotyczącego górnego ograniczenia funkcji kryterialnej. Wartość ta pochodzi z algorytmów heurystycznych. Jest to kolejny powód do dalszych prac nad tymi algorytmami. W badaniach dotyczących algorytmów heurystycznych można zaobserwować iż algorytm FlowDevation znajduje rozwiązanie dopuszczalne w czasie rzędu kilku sekund, jednak jest ono odległe od rozwiązania optymalnego o ok. 7-9%. W przypadku algorytmu Tabu Search otrzymujemy rozwiązanie dopuszczalne odległe od optymalnego o 1-3%, niemniej jednak czas działania algorytmu jest dłuższy i wynosi kilkanaście do kilkudziesięciu sekund. Należy zatem odpowiednio dobrać parametry algorytmy Tabu Search – długość listy tabu oraz liczba iteracji. W pracy dotyczącej przepustowości modularnych znajdują się szczegółowe badania dotyczące tych dwóch parametrów.