View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

Paper

Optimization algorithm for reconfiguration process of the IP over optical networks

Nghia Le Hoang

Abstract — The IP over optical (IPO) network is becoming one of the most interesting among all proposed models of transport networks nowadays. In an IPO network, the reconfiguration capability of the network could be used in order to balance the load of its network elements (NEs). Reconfiguration operations (i.e., switching in OXC nodes and rerouting in IP routers) take place in real-time. Consequently, intensive changes in NEs settings might cause failures in the existing connections in the network. For that reason, changes in NEs settings should be coordinated in a reconfiguration process. In this paper, the author has proposed an optimization method for such reconfiguration process. The mathematical model of the method including computation results has been presented.

Keywords — network reconfiguration, network optimization, IP over optical network.

1. Introduction

IP networks are to transport traffic of various services and with different characteristics [1, 2, 9, 10]. IP traffic, in comparison with traditional telephony traffic, is characterized by greater dynamics and irregularity [13–15]. In order to cope with changing traffic conditions, either the capacity of NEs should be over-provisioned, or NEs should elastically adapt to the changes of traffic conditions. Considering the huge annual growth of IP traffic, the second solution seems to be more desirable. However, the reconfiguration capabilities of NEs are limited. For that reason, changes in NEs settings should be optimally coordinated in a reconfiguration process. A model of such reconfiguration process has been proposed in this paper. The mechanism of the reconfiguration process is based on the optimization method described in the further sections.

1.1. The architecture of an IPO network

An IPO network consists of two layers – the *optical layer* and the *IP layer*. Components of each layer are grouped into three generic classes: nodes, links and paths. In each layer, a link connects two nodes, whilst a path is defined as a sequence of links.

In the optical layer, an *optical node* represents an optical cross connect (OXC). An *optical link* is constructed of one or several parallel optical fiber trails terminated at its nodes. An *optical path* transports one or several optical signals (lambdas). The frequency (wave) of an optical signal should be the same along its path, unless some O/E/O conversions are used on its way. Each optical signal carries a bit-stream. Consequently, an optical path, after E/O-O/E adaptation, is seen by the IP layer as a bundle of electric bit-streams, which has certain rate.

In an IPO network, an IP router is connected to one or several OXCs by means of *inter-layer connectors*. An inter-layer connector could be implemented with an optical fiber or a copper pair. In the first case, the IP router should have E/O-O/E conversion capability. Otherwise, such a capability should be located in the OXC.

In the IP layer, an *IP node* represents an IP router. An *IP link* is constructed of the bit-streams transported by one or several optical paths. These optical paths should start at an optical node connected to an IP node of the IP link, and end at an optical node connected to the other IP node of the IP link. An *IP path* (i.e., a sequence of IP links) transports IP packets from its source node to its destination node.

A simplified illustration of the architecture of an IPO network is shown in Fig. 1.

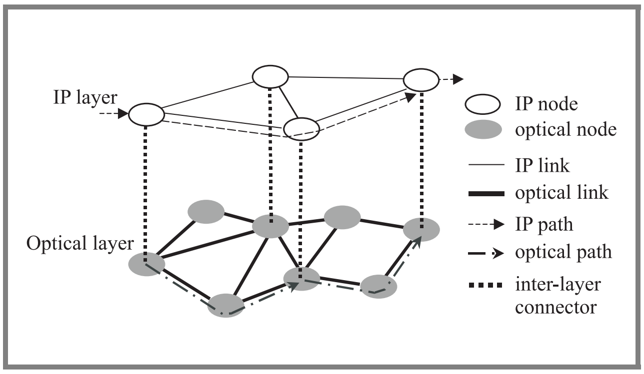

Fig. 1. The architecture of an IPO network.

The services offered by an IPO network are grouped into *service classes*. Between some pair of IP nodes, in each service class, a *demand* is defined in order to represent the offered traffic generated by the users connected to these nodes. The bandwidth required by a demand is called the volume of the demand, and is realized by means of one or several IP paths.

1.2. The description of a reconfiguration process

A *network configuration* corresponds to a feasible combination of NEs settings. The process, in which NEs settings evolve to adapt the network configuration to the traffic conditions, is called the *reconfiguration process* of the network (Fig. 2). The effect of changing traffic conditions is that the network configuration could not be the optimal one in a long time. The consequence of the non-optimal configuration is a low quality of service (QoS) level for connections in the network. In order to prevent such a situation, the reconfiguration process should be active each time when a fall in QoS level is forecasted. Each activation of the reconfiguration process is called a *reconfiguration procedure*. The result of the reconfiguration procedure should be an optimal configuration with regard to the estimated traffic conditions.

Because generally the network configuration cannot be changed to the optimal one at once, the reconfiguration procedure should be decomposed into a sequence of *reconfiguration operations*. Each reconfiguration operation represents an indivisible action that modifies the network configuration.

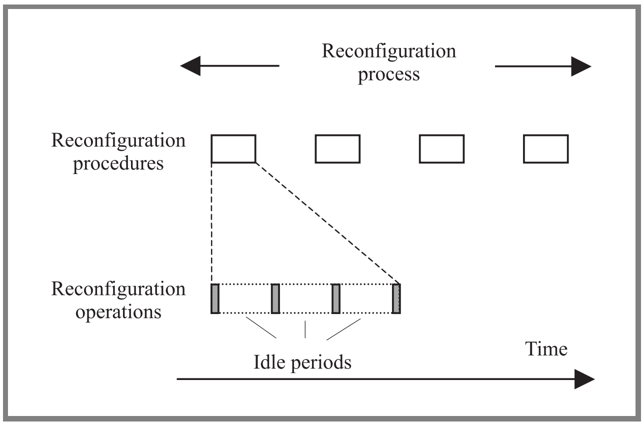

Fig. 2. The structure of a reconfiguration process.

When a reconfiguration operation takes place, some NEs settings are in an unstable state. It is one of the reasons why the next reconfiguration operation should not start immediately after the preceding reconfiguration operation ends. Furthermore, a period is necessery in order to propagate the information about changes of NEs settings throughout the network. Therefore, each reconfiguration operation should be followed by an idle period considerably larger than its duration.

The configuration between two consecutive reconfiguration operations is called an *intermediate configuration*. In a reconfiguration procedure, a series consisting of intermediate configurations and the optimal configuration determines how the network configuration should be changed. In the further part of this paper, such series is called the *configuration series* of a reconfiguration procedure.

The configuration series of a reconfiguration procedure would be computed by means of an optimization algorithm. Such an algorithm has been proposed in this paper. The principle of the algorithm is based on the *optimization problem of reconfiguration* (OPR) that expresses the relations among NEs and the suitability degree of a network configuration to traffic conditions.

2. The simplified description of OPR

The formulation of OPR consists of a group of constraints and an objective function. The constraints of OPR result from the relations among NEs, whilst the objective function of OPR corresponds to the suitability degree of a network configuration to traffic conditions.

2.1. The relations among the NEs – the constraints of OPR

The main constraints of OPR are below:

1) the conservation of flows – the Kirchhoff's law,

2) the budget constraint,

3) the network inertia constraint.

Other possible constraints, e.g. the continuity of an optical wave along its path [5], etc., are neglected in order to simplify the OPR algorithm. It is possible to introduce them into the model in future works.

The result of constraint (1) is that, among the links of a path, the amount of capacity reserved for the path in each link should be equal. Such amount is defined as the capacity of a path. In the IP layer, considering the possibility of packets loss, the Kirchhoff's law should only be a simplifying assumption.

Constraint (2) states that, the total capacity of the paths going across a link (or a node) is restricted by the capacity of the link (or the node).

Constraint (3) is a specific constraint of OPR. Network inertia results from various aspects. Firstly, the switching time in an OXC is finite ($\sim$ 50 ms). During the switching time, a stream of 10 Gbit/s rate suffers a loss of 0.5 Gbit. Such loss causes the number of simultaneous switching operations to be limited [3, 8, 16]. Furthermore, a node (an OXC or an IP router), in order to modify its paths, should negotiate with other nodes. The duration time of such procedures is in the order of seconds or dozens of seconds. With a large number of modified paths, commands between nodes have to be queued, and consequently, the unstable period might be minutes or tens of minutes [4].

The full model of network inertia including all its parameters might be quite complicated. In this paper, only the main parameters of network inertia are considered:

– the maximal feasible number of switching operations of a node, within a reconfiguration operation;

– the minimal feasible interval between two consecutive reconfiguration operations.

The network inertia constraint is used in computing an intermediate configuration. In contrary, the optimal configuration is computed by solving the OPR without this constraint.

2.2. The suitability degree of a network configuration to traffic conditions – the objective function of OPR

The suitability degree of a network configuration to traffic conditions is determined with the level of QoS in the network. The level of QoS for connections of a demand depends on the ratio of the bandwidth dedicated for the demand to the volume of the demand. Such ratio is called the *dedicated-requested ratio*. Because network resources are shared for all demands, the level of QoS in the network as a whole should be constructed according to a fairness principle [7, 11, 12]. In this paper, the simplest one has been considered. The level of QoS in the network is defined as the lower bound of the dedicated-requested ratio for each demand.

Because the optimization purpose is to increase the level of QoS in the network, maximization is optimizing the direction. In the basic version, the formulation of the objective function is:

$$u = \min_d \left(a_d / z_d\right),$$

where a_d denotes the bandwidth dedicated for demand d, and z_d denotes the volume of demand d.

2.3. A simplified scenario of a reconfiguration procedure

One of the constraints defined in OPR results from network inertia. Network inertia is the reason why a reconfiguration procedure should be decomposed into reconfiguration operations. In a reconfiguration procedure, a configuration series determines how the network configuration should be changed. Each element of such series could be computed by solving a mixed integer programming (MIP) problem with the objective and the constraints based on OPR.

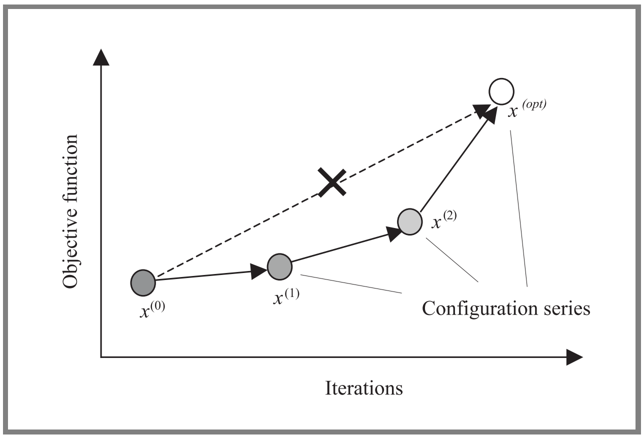

Fig. 3. A reconfiguration procedure.

A configuration series could be computed as follows.

Suppose the current configuration is $x^{(0)}$, whilst the optimal one is $x^{(\mathrm{opt})}$. Let us assume further that $x^{(\mathrm{opt})}$ is not directly reachable from $x^{(0)}$ due to the inertia constraints; while the best reachable configuration (constricted by the network inertia) is $x^{(1)}$ with an objective value much worse than $x^{(\mathrm{opt})}$'s. In this case, we could compute a configuration $x^{(2)}$ modified from $x^{(1)}$ and better than $x^{(1)}$. After a finite number of such iterations, we can achieve the optimal configuration (Fig. 3).

It was only a simplified illustration. Detailed problems (e.g. the shortest way to reach the optimal configuration, the degeneration effect, etc) have been discussed in further sections of this paper.

3. The mathematical model of OPR

3.1. The objects of an IPO network

An IPO network is modeled with generic objects (nodes, links, paths) of both layers and demands. In each configuration, the absence of such object as an optical path, an IP link or an IP path is denoted with a zero capacity of this object. Hence, a creation or a deletion of such object is expressed by a change of capacity from zero to a nonzero value or *vice versa*.

The IP layer is modeled as a directed graph $\boldsymbol{G}_{\mathrm{IP}} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{P})$, where $\boldsymbol{V} = \{v\}_{v=1}^{V}$, $\boldsymbol{E} = \{e\}_{e=1}^{E}$ and $\boldsymbol{P} = \{p\}_{p=1}^{P}$ denote the set of IP nodes, the set of IP links and the set of IP paths, respectively.

Similarly, the optical layer is modeled as a directed graph $\boldsymbol{G}_{\mathrm{opt}} = (\boldsymbol{W}, \boldsymbol{F}, \boldsymbol{Q})$, where $\boldsymbol{W} = \{w\}_{w=1}^{W}$, ,$\boldsymbol{F} = \{f\}_{f=1}^{F}$ and $\boldsymbol{Q} = \{q\}_{q=1}^{Q}$ denote the set of optical nodes, the set of optical links and the set of optical paths, respectively. An optical path serves an IP link by transporting its bit-streams. The set of optical paths that serve IP link $e \in \boldsymbol{E}$ is denoted by $\boldsymbol{Q}(e)$.

Symbols $\boldsymbol{S} = \{s\}_{s=1}^{S}$ and $\boldsymbol{D} = \{d\}_{d=1}^{D}$ denote the set of service classes and the set of demands, respectively. The service class that demand $d \in \boldsymbol{D}$ belongs to is denoted with $s(d)$. The set of IP paths that serve demand $d \in \boldsymbol{D}$ is denoted by $\boldsymbol{P}(d)$.

Each object has some attributes representing e.g. capacity, switching capability etc. The value of such attribute is a scalar quantity. In each object set, each attribute could be denoted with a vector in the $\mathfrak{I}^{+N}$ or $\mathfrak{R}^{+N}$ space (N denotes the number of members of the set). Such characteristic vectors have been described in the next subsections.

3.2. The characteristic vectors of IPO network objects

The capacity of an IP link or an IP path is measured in bit/s. Symbols $b = (b_e)_{e \in E} \in \mathfrak{R}^{+E}$ and $\xi = (\xi_p)_{p \in P} \in \mathfrak{R}^{+P}$ denote the capacity vectors of IP links and IP paths, respectively. The relation between vectors b and ξ, the budget constraint, is given by

$$\forall e \in \boldsymbol{E}: \sum_{p: p \in P \wedge p \ni e} \xi_p \leq b_e\,, \qquad (1)$$

where $p \ni e$ denotes that path p goes across link e.

The capacity of an optical link or an optical path is measured in the number of lambdas. Symbols $c = (c_f)_{f \in F} \in \mathfrak{J}^{+F}$ and $\eta = (\eta_q)_{q \in Q} \in \mathfrak{J}^{+Q}$ denote the capacity vectors of optical links and optical paths, respectively. The relation between vectors c and η, the budget constraint, is presented below:

$$\forall f \in \boldsymbol{F}: \sum_{q: q \in Q \wedge q \ni f} \eta_q \leq c_f \,. \tag{2}$$

In an IPO network, an IP link is constructed by the bitstreams of one or several optical paths. Hence, the value of vector b could be directly derived by the value of vector η by the following formula:

$$\forall e \in \boldsymbol{E}: b_e = C_\lambda \cdot \sum_{q \in Q(e)} \eta_q \,. \tag{3}$$

Here C_λ is a constant that denotes the bandwidth of the bitstream transported by an optical signal. Besides the link capacity and path capacity, the node capacity should also be considered. The limited capacity of an optical node results from the bounded space in the switching matrix of an OXC and causes a limited number of its input and output ports. In each optical node, the number of input ports and the number of output ports are usually equal. Thus, the capacity vector of optical nodes is denoted consistently with $n = (n_w)_{w \in W} \in \mathfrak{J}^{+W}$. The ports of an OXC are used in connecting the OXC to other OXCs and IP routers. The number of connections in the first type (OXC–OXCs) is equal the total capacity of optical links ended in the OXC. The number of connections in the last type (OXC–IP routers) is equal the total capacity of optical paths ended in the OXC. Consequently, the number of busy ports in an optical node could be accounted by the total capacity of optical links and optical paths ended at the node. In particular, the number of busy input ports of node $w \in W$ is

$$\sum_{f: eg(f) = w} c_f + \sum_{q: ing(q) = w} \eta_q \,,$$

and its number of busy output ports is

$$\sum_{f: ing(f) = w} c_f + \sum_{q: eg(q) = w} \eta_q \,.$$

The budget constraint is formulated in this case as follows:

$$\forall w \in \boldsymbol{W}: \sum_{f: eg(f) = w} c_f + \sum_{q: ing(q) = w} \eta_q \leq n_w \tag{4}$$

$$\forall w \in \boldsymbol{W}: \sum_{f: ing(f) = w} c_f + \sum_{q: eg(q) = w} \eta_q \leq n_w \,, \tag{5}$$

where $ing(f)$ and $eg(f)$ denote, respectively, the ingress node and the egress node of link f. Similarly, $ing(q)$ and $eg(q)$ denote the ingress node and the egress node, respectively, of path ‘q.

The limited capacity of an IP node results from the bounded computational capability of an IP router. The capacity of an IP node $v \in \boldsymbol{V}$ represents the maximal total capacity of IP links ended at the node. The capacity vector of IP nodes is denoted by $m = (m_v)_{v \in V} \in \mathfrak{R}^{+V}$. The budget constraint is formulated in this case as follows:

$$\forall v \in \boldsymbol{V}: \sum_{eg(e) = v} b_e \leq m_v \tag{6}$$

$$\forall v \in \boldsymbol{V}: \sum_{ing(e) = v} b_e \leq m_v \,. \tag{7}$$

Vectors m, n and c are parameters that do not depend on NEs settings. By contrast, vectors ξ and η depend on them. On the other hand, the pairs (ξ, η) from different network configurations should not be equal. Hence, a configuration of an IPO network could be defined with a pair (ξ, η). In this paper, the pair (ξ, η) has sometimes been denoted in a convenient form as a mixed vector $x = (\xi, \eta) \in \mathfrak{R}^{+P} \times \mathfrak{J}^{+Q}$.

In an IPO network with known m, n and c, vector $x \in \mathfrak{R}^{+P} \times \mathfrak{J}^{+Q}$ represents a network configuration if and only if it satisfies the conditions from (1) to (7). The set of all such vectors is called the set of feasible configurations, and denoted by $\boldsymbol{K}_{mnc}$.

3.3. The formulation of network inertia

Network inertia causes some configurations not to be directly reachable (i.e., reachable within a reconfiguration operation) from the current configuration. Besides, it prevents reconfiguration operations from taking place too frequently.

Such aspects of the network inertia are represented by following parameters:

- the maximal feasible number of switching operations of a node, within a reconfiguration operation;
- the minimal feasible interval between two consecutive reconfiguration operations.

The switching capability of an IP node is represented by the maximal feasible number of switching operations of the node within a reconfiguration operation. The switching capability vector of IP nodes is denoted with $\alpha = (\alpha_v)_{v \in V} \in \mathfrak{J}^{+V}$. Each switching operation causes a change in the capacity of an IP path. In particular, such change could be a growth from zero or a fall to zero that practically means a creation or a deletion of an IP path. Thus, in each node, the limited switching capability causes the bounded number of changes in the capacity of IP paths that go across the node. Denoting by ξ and ξ' the path capacity vectors of the current and the next configuration, respectively, the following inequality holds:

$$\forall v \in \boldsymbol{V}: \sum_{p: p \ni v} \left| \operatorname{sgn}(\xi'_p - \xi_p) \right| \leq \alpha_v \,, \tag{8}$$

where sgn denotes the sign-num function (sgn(r) returns the sign of real number r). The inequality (8) could be transformed into a mixed integer form by using a binary indicator $\delta = (\delta_p)_{v\in V} \in \{0,1\}^P$:

$$\forall p \in \boldsymbol{P} : -\delta_p \leq M \cdot \left(\xi_p' - \xi_p\right) \leq \delta_p \tag{9}$$

$$\forall v \in \boldsymbol{V} : \sum_{p:p\ni v} \delta_p \leq \alpha_v \,. \tag{10}$$

In the inequality (9), M stands for a big constant (e.g. 10^6). Because of the large value of $M, \delta_p = 0$ only if $\xi_p = \xi_p'$, so δ_p could represent the value of $\left|\mathrm{sgn}(\xi_p' - \xi_p)\right|$.
The switching capability of an optical node is represented by the maximal feasible number of switching operations of the node within a reconfiguration operation. The switching capability vector of optical nodes is denoted with $\beta = (\beta_w)_{w\in W} \in \mathfrak{J}^{+W}$. The effect of a switching operation is an increment or a decrement of one optical wave in an optical path. Thus, in the optical node, the limited switching capability causes the bounded sum of capacity change of optical paths that go across the node. Denoting by η and η' the path capacity vectors of the current and the next configuration, respectively, we have the following inequality:

$$\forall w \in \boldsymbol{W} : \sum_{q:q\ni w} \left|\eta_q - \eta_q'\right| \leq \beta_w \,. \tag{11}$$

The reader could notice some dissimilarity between the inequalities (8) and (11). The dissimilarity results from the different ways to change the path capacity in the layers of an IPO network. A change in the capacity of an optical path requires a number of physical switching operations. The more a capacity changes, the larger number of such operations is needed. In the contrary, the number of operations for a change in the capacity of an IP path generally does not depend on the amount of capacity to be changed. Hence, the number of operations executed by an IP router to modify capacity of some IP paths depends on the number of changed paths (detected by the signnum function), not on the amount of capacity to be changed.

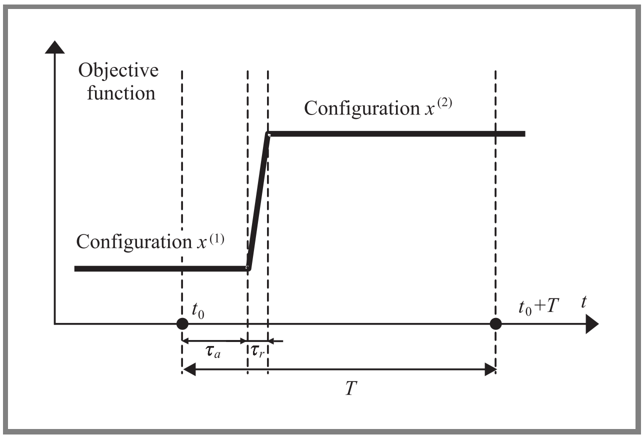

Fig. 4. Time parameters.

In an IPO network of known parameters m, n, c, α and β, a configuration $x' = (\xi', \eta')$ belong to $\boldsymbol{K}_{mnc}$ is directly reachable from the current configuration $x = (\xi, \eta)$ if and only if the inequalities (8) and (11) hold. The set of such configurations is denoted by $\boldsymbol{H}_{mnc\alpha\beta}(x)$, and called the neighborhood of configuration x. In further part of this paper, assuming the parameters m, n, c, α and β to be known and constant, the notations $\boldsymbol{K}_{mnc}$ and $\boldsymbol{H}_{mnc\alpha\beta}(x)$ have been replaced by shortened forms $\boldsymbol{K}$ and $\boldsymbol{H}(x)$.
Another effect of the network inertia is the need of an idle period between consecutive reconfiguration operations. When a reconfiguration operation takes place, some NEs are in an unstable state. Furthermore, a period of time is needed to propagate the information about changes of NEs settings throughout the network. Consequently, each reconfiguration operation should be followed by an idle period considerably larger (e.g. 10^3 times) than its duration. The duration τ_r of a reconfiguration operation is in the order of 100 ms, so interval between reconfiguration operations T should be at least tens of seconds.
Besides the effect of network inertia, another factor delaying a reconfiguration procedure is the duration of traffic analysis and optimization computing. Such duration, denoted by τ_a, should not exceed 10% of T.
The relation among τ_r, τ_a and T is illustrated in Fig. 4. Furthermore, this figure shows how the objective function changes when a reconfiguration operation takes place.

3.4. The demand models

In the preceding subsections, the model of an IPO network has been described. The volume of a demand changes in time and is a quantity independent of the network configuration. Furthermore, it is generally difficult to determine the volume of a demand exactly at each moment. In this paper, two demand models have been proposed:

1) the deterministic model, based on the effective bandwidth;

2) the probabilistic model.

In the first model, the volume of a demand is represented by the effective bandwidth of the demand. The effective bandwidth of a demand is defined as the amount of bandwidth enough to serve the demand with an acceptable QoS. By this way, traffic conditions in the network are represented by the effective bandwidth vector $z = (z_d)_{d\in D} \in \mathfrak{R}^{+D}$.
In the second model, traffic conditions in the network are represented by random vector $Y = (Y_d)_{d\in D}$ of known distribution function $F_Y : \mathfrak{R}^{+D} \to [0\ldots1]$. Each element of the random vector Y represents the probabilistic behavior of the volume of a demand.
Because of the dynamics and irregularity of IP traffic, an effective bandwidth defined for a long period (e.g. for all day) causes to be useful. Such a quantity might be too large in comparison with the mean used bandwidth. Similarly, a random vector Y defined for such a long period might have too wide variation. Since the NEs are able to adapt to the changes of traffic conditions, it is not necessary to define z or Y for such a long period. This period should

be chosen in the order of the duration of a reconfiguration process.

3.5. Optimization objective models

The formulation of an optimization objective function depends on the demand model. For the deterministic demand model, the objective function, denoted by $u_1(z,x)$, has the basic formulation as follows:

$$u_1(z,x) = \min_{d\in D}(a_d/z_d)\,. \tag{12}$$

In this formulation, $a_d = \sum_{p:p\in P(d)} \xi_p$ denotes the bandwidth dedicated for demand $d \in D$, which equals the total capacity of the IP paths serving this demand.

The advanced formulation of the objective function, which includes the service class weights, is described below:

$$u_1(z,x) = \min_{d\in D}(\rho_{s(d)} \cdot a_d/z_d)\,. \tag{13}$$

The weights of service classes are denoted by vector $\rho = (\rho_s)_{s\in S} \in \mathfrak{R}^{+S}$.

The objective function for the probabilistic demand model is defined with the help of function u_1. In this case, the value of function $u_1(y,x)$ represents the QoS level of configuration x for an instance y of the random vector Y. Thus, the statistical QoS level of configuration x, which is denoted by $u_2(Y,x)$, could be derived with a Lebesgue's integral:

$$u_2(Y,x) = \int_{\mathfrak{R}^{+D}} u_1(y,x)dF_Y(y)\,. \tag{14}$$

Further in this paper, the objective functions $u_1(z,x)$ and $u_2(Y,x)$ have sometimes been denoted by common form $u(x)$ in such situation when the formulations do not depend on the demand model.

4. The optimization algorithm proposed for reconfiguration procedures

Having the model of OPR, we could now determine how a reconfiguration procedure should take place. The configuration series of a reconfiguration procedure should satisfy the following requirements:

- The number of reconfiguration operations in the reconfiguration procedure should be minimized.
- The value of the objective function of a configuration should not be worse than its preceding configuration.
- The last element in the configuration series should be the optimal configuration.

In this paper, three algorithms for determining the configuration series have been proposed. The first exactly obeys the requirements mentioned above. The others are some heuristic ones.

4.1. Algorithm 1 – by defining a master problem

Algorithm 1 is to solve the master problem defined below: Minimize N, subjects to:

$$\forall k = 1\ldots N : x^{(k)} \in \boldsymbol{H}\left(x^{(k-1)}\right) \tag{15}$$

$$\forall k = 1\ldots N : u\left(x^{(k)}\right) \geq u\left(x^{(k-1)}\right) \tag{16}$$

$$u\left(x^{(N)}\right) = \max_{x\in K} u(x)\,, \tag{17}$$

where $x^{(0)}$ represents the current configuration; $x^{(1)}\ldots x^{(N-1)}$ the intermediate configurations and $x^{(N)}$ the optimal configuration. Given is $x^{(0)}$, whilst $x^{(1)}\ldots x^{(N)}$ are variables to find.

The procedure for resolving the master problem is following:

1. Let $N \leftarrow 1$.
2. Find $x^{(1)}\ldots x^{(N)}$ satisfying (15)–(17). If there exist such vectors, then **finish**, the series $\left[x^{(1)}\ldots x^{(N)}\right]$ found is the solution we look for.
3. Otherwise, let $N \leftarrow N+1$, and **go to** Step 2.

In each time at Step 2, a MIP problem should be resolved. The size of such MIP problem increases in proportion with N. Therefor, at present; the algorithm could be used only for small networks. In future works, the efficiency of the algorithm could be improved by using a decomposition technique.

4.2. Algorithm 2 – by finding the local optimum

The idea of Algorithm 2 has been described in the simplified illustration in Section 1. This algorithm is based on repeating the local optimization computing in order to reach the global optimal configuration. The description of this algorithm is below:

1. Compute $u_{\text{opt}} = \max_{x\in K} u(x)$.
2. Let $N \leftarrow 1$.
3. Compute such configuration $x^{(N)} \in \boldsymbol{H}\left(x^{(N-1)}\right)$ that $u\left(x^{(N)}\right) = \max_{x\in H\left(x^{(N-1)}\right)} u(x)$. If $u\left(x^{(N)}\right) = u_{\text{opt}}$, then **finish**, the series $\left[x^{(1)}\ldots x^{(N)}\right]$ found is the solution we look for.
4. Otherwise, let $N \leftarrow N+1$, and **go to** Step 3.

Algorithm 2 has a critical drawback. The consequence of the drawback might occur when $u\left(x^{(N)}\right) = u\left(x^{(N-1)}\right)$ in Step 3. In such a situation, we have no guarantee that the next configuration must not be the same as some earlier one. Then, the computation procedure might fall into a perpetual loop, and never reach the optimal configuration. The phenomenon is known as a degeneration effect. In order to repair the drawback, some anti-degeneration techniques should be used, e.g. lexicographic ordering of the configurations.

4.3. Algorithm 3 – by finding the shortest distance to the global optimum

Algorithm 3 is an improved version of Algorithm 2. The description of this algorithm is below:

1. Compute the global optimal configuration $x^{(\mathrm{opt})}$, so $u\left(x^{(\mathrm{opt})}\right) = \max_{x\in K} u(x)$.
2. Let $N \leftarrow 1$.
3. Find such configuration $x^{(N)} \in \boldsymbol{H}\left(x^{(N-1)}\right)$ that $u\left(x^{(N)}\right) = \max_{x\in H\left(x^{(N-1)}\right)} u(x)$ and $\sum_{k=1}^{P+Q} \left|x_k^{(N)} - x_k^{(\mathrm{opt})}\right|$ to be minimized. If $u\left(x^{(N)}\right) = u_{\mathrm{opt}}$, then **finish**, the series $\left[x^{(1)} \ldots x^{(N)}\right]$ found is the solution we look for.
4. Otherwise, let $N \leftarrow N+1$, and **go to** Step 3.

In comparison with Algorithm 2, in Step 3, the configuration $x^{(N)}$ should not only be the best in the neighborhood of the configuration $x^{(N-1)}$, but should also be the nearest to the optimal configuration $x^{(\mathrm{opt})}$. The introduction of this condition avoids the degeneration effect in most practical cases.

5. Computation examples

The method has been implemented as a C program with the use of the CPLEX optimization package [6]. Two examples have been considered. In Example 1, a small size network is used for illustrating the method. In Example 2, a medium size network is used for testing the convergence of the method. In these examples, the deterministic demand model has been used.

5.1. Example 1 – a small size network for illustrating the method

The topology of the network in this example is shown in Fig. 5. In this figure, an edge (without an arrow) represents a pair of links in both directions between two nodes.

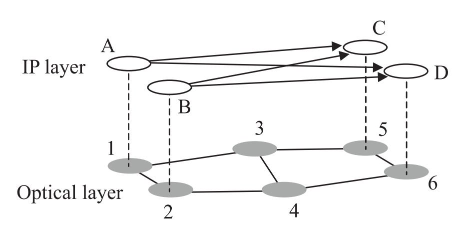

Fig. 5. The topology of the network in Example 1.

The optical layer consists of 6 nodes and 14 links. The capacity of an optical node equals 10 lambdas. The capacity of an optical link equals 5 lambdas. Each lambda carries a bit-stream with 1 Gbit/s. Four optical nodes are connected to IP nodes.

In order to simplify the illustration, only the optical layer is considered, while the IP layer is reduced as much as possible.

The IP layer consists of 4 nodes and 4 links. Each demand is served by only one IP path consisting of only one IP link. The demands are AC, AD, BC and BD. The traffic conditions in the network are represented with effective bandwidth vector z.

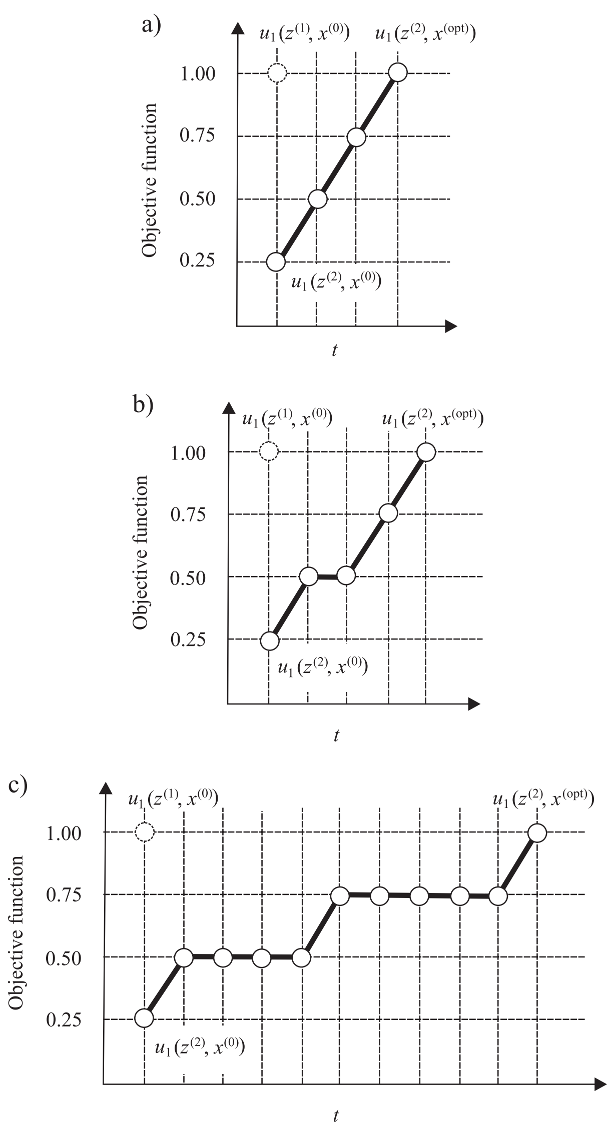

Fig. 6. The configuration series of the reconfiguration procedure in different cases: (a) $\beta_0 = 3$; (b) $\beta_0 = 2$; (c) $\beta_0 = 1$.

Network inertia is represented by the maximal feasible number of switching operations of an optical node within a reconfiguration operation. Such number is denoted by β_0. As regards IP nodes, the network inertia constraint is neglected.

We suppose vector z changes from $z^{(1)} = (1,4,1,4)$ to $z^{(2)} = (4,1,4,1)$. As vector z changes from $z^{(1)}$ to $z^{(2)}$, the objective function falls from 1 to 0.25. We consider three cases: when β_0 equals 3, 2 or 1. Figure 6 shows how the reconfiguration procedure should take place in each case.

When $\beta_0 = 3$, only 3 reconfiguration operations are needed. The objective function successively grows from 0.25 to 1. In such case, both Algorithms 2 and 3 could be used.

When $\beta_0 = 2$, a problem occurs in the second reconfiguration operation: the objective function of $x^{(1)}$ and $x^{(2)}$ are equal. In this case, Algorithm 2 cannot be used because of the degeneration effect. However, Algorithm 3, having an anti-degeneration technique, copes with the problem and succeeds. In this case, 4 reconfiguration operations are needed.

When $\beta_0 = 1$, the network inertia constraint becomes too severe and causes the reconfiguration procedure to take place in 10 reconfiguration operations. Algorithm 3 is the only one that could be used.

In this example, the relation between the reconfiguration capability and the duration of a reconfiguration procedure has been observed. It seems that for a not too severe network inertia constraint, the length of configuration series is acceptable. In the next example, a medium size network is used for testing the convergence of the method.

5.2. Example 2 – a medium size network for testing the convergence of the method

The topology of the network of this example is shown in Fig. 7. In this figure, an edge represents a pair of links in both directions between two nodes.

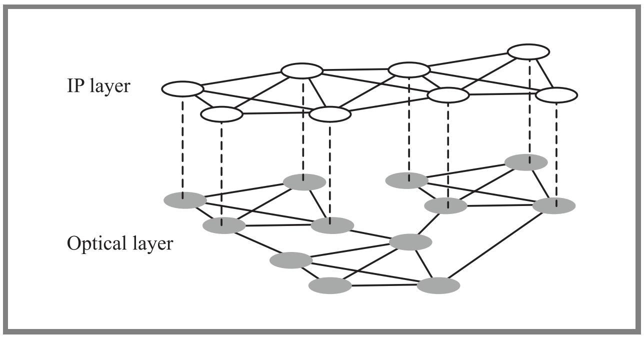

Fig. 7. The topology of the network in Example 2.

The optical layer consists of 12 nodes and 44 links. The capacity of an optical node equals 100 lambdas. The capacity of an optical link equals 10 lambdas. Each lambda carries a bit-stream with 1 Gbit/s. Eight optical nodes are connected to IP nodes. Between each pair of such nodes, because of the enormous number of possible paths, only 4 optical paths are chosen as the admissible paths.

The IP layer consists of 8 nodes and 24 links. The capacity of an IP node equals 100 Gbit/s. Between each pair of IP nodes, 3 IP paths are chosen as admissible paths. The traffic conditions in the network are represented with effective bandwidth vector z. Each element of vector z should be a quantity from 0 to 1 Gbit/s.

Network inertia is represented by the maximal feasible number of switching operations of an optical node within a reconfiguration operation. Such number is chosen to be 4 for each optical node. As regards IP nodes, the network inertia constraint is neglected.

In order to test the method, a random series $\left[z^{(1)} \dots z^{(K)}\right]$ has been generated. Each pair $\left(z^{(k)},\, z^{(k+1)}\right)$ represents a change of traffic conditions in the network. Each time when traffic conditions change, a reconfiguration procedure should take place. In order to determine how the network configuration changes in a reconfiguration procedure, a configuration series is computed. Algorithm 3 has been used.

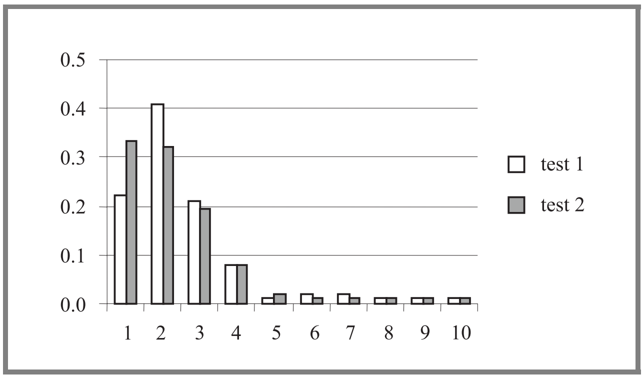

Fig. 8. The test results.

Two tests have been realized. In the first test, the elements of random series $\left[z^{(1)} \dots z^{(K)}\right]$ are probabilistically independent. In the second test, the elements of random series $\left[z^{(1)} \dots z^{(K)}\right]$ are correlated by the following rule: for a demand, the probability, that the change of its effective bandwidth is above 50%, is 20%. This rule is denoted as follows:

$$\forall k = 1 \dots K-1,\ \forall d \in \boldsymbol{D} : \Pr\left\{ \left| z_d^{(k+1)} - z_d^{(k)} \right| \geq 0.5 \cdot z_d^{(k)} \right\} = 0.2.$$

Figure 8 presents the distribution of the number of reconfiguration operations in a reconfiguration procedure. We see that, in this example, the number varies just from 1 to 6.

References

[1] B. T. Doshi, “Future WAN architecture driven by services, traffic volume, and technology trends”, *Bell Labs Tech. J.*, vol. 6, no. 1, 2001.

[2] H. Epstein, “Managing optical networks”, *Bell Labs Tech. J.*, vol. 6, no. 1, 2001.

[3] D. Banerjee, "Wavelength-routed optical networks: linear formulation, resource budgeting tradeoffs, and a reconfiguration study", *IEEE/ACM Trans. Netw.*, vol. 7, 2000.

[4] B. T. Doshi, "Optical network design and restoration", *Bell Labs Tech. J.*, vol. 4, no. 1, 1999.

[5] R. Dutta, "A survey of virtual topology design algorithms for wavelength routed optical networks", *Opt. Netw. Mag.*, vol. 10, 2000.

[6] ILOG, "User's Manual of ILOG CPLEX 6.5", 1999.

[7] W. Ogryczak, "Multiple criteria optimization and decisions under risk", *Control Cybern.*, vol. 31, 2002.

[8] J. F. P. Labourdette, "Branch-exchange sequences for reconfiguration of lightwave networks", *IEEE Commun.*, vol. 37, 1994.

[9] B. Rajagopalan, "IP over optical networks: a framework", Internet Draft, draft-ietf-ipo-framework-00.txt

[10] S. Seetharamna, "IP over optical networks: a summary of issues", Internet Draft, draft-osu-ipo-mpls-issue-02.txt

[11] W. Ogryczak and T. Śliwiński, "On solving liner programs with the ordered weighted averaging objective", *Eur. J. Oper. Res.*, vol. 148, pp. 80–91, 2003.

[12] A. Tomaszewski, "A polynomial algorithm for max-min fairness problem", in *Second Polish-German Teletraffic Symp.*, Gdańsk, Poland, 2002.

[13] Z. Papir, *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*. Warszawa: WKŁ, 2001.

[14] A. Adas, "Traffic models in broadband networks", *IEEE Commun. Mag.*, July 1997.

[15] V. S. Frost, "Traffic modeling for telecommunications networks", *IEEE Commun. Mag.*, March 1994.

[16] G. N. Rouskas and M. H. Ammar, "Dynamic reconfiguration in multihop WDM networks", *J. High Speed Netw.*, vol. 4, no. 3, 1995.

Nghia Le Hoang was born in Vietnam in 1974. He received his M.Sc. degree in telecommunications from Warsaw University of Technology, in 1998. He is now preparing his Ph.D. theses, concerning reconfiguration methods for telecommunications networks, at Warsaw University of Technology, in Institute of Telecommunications.

e-mail: nhoang@tele.pw.edu.pl
Institute of Telecommunications
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland