Paper

End-to-end service survivability under attacks on networks

Wojciech Molisz and Jacek Rak

Abstract— Network survivability is a capability of a networked system to provide its services despite failures or attacks. Attacks, e.g., due to acts of war, being potentially damaging events, were basically considered in the historical definitions of a survivability phenomenon. The meaning of the term: "network survivability" evolved in the last decade. Recently, attacks replayed the important role again. Their nature, however, including intrusions, probes, denials of service, differs from the old one. Survivability is strongly related to other fields of study. In particular, quality of service depends on network survivability. We investigate these dependencies in scale-free networks. Many networks are scale-free, i.e., their node degree distribution follows the power law. Nodes of the highest degrees, called centers, are highly vulnerable to attacks. Elimination of these nodes seriously degrades the overall performance of network services. In this paper we propose a model, which, based on traffic parameters of a demand, like delay or bit rate, allows to establish the survivable and attack proof end-to-end connections. The key idea of this model is that for the significant traffic, it establishes paths, which omit centers. The important connections become more resistant to attacks. We show that in the best case, obtained for the highest class of service, the number of broken connections is reduced even by factor 3. Example results are compared to those for the standard distance metrics. Our model is applicable to many network architectures and many classes of service.

Keywords— survivable data networks, attacks on networks, scale-free networks, routing, resource allocation.

1. Introduction

Network survivability is a capability of a networked system to provide its services despite failures or attacks. Attacks, e.g., due to acts of war, being potentially damaging events, were basically considered in the historical definitions of a survivability phenomenon. In the last decade, focus was rather on protecting systems against failures, due to software defects, hardware faults or human errors. Recently, attacks replayed the important role again. Their nature, however, including intrusions, probes, denials of service, differs from the old one.

Survivability is strongly related to other fields of study, including fault-tolerance, reliability, safety or performance evaluation. In particular, quality of service depends on network survivability. In this paper we investigate these dependencies in scale-free networks (SFNs). Many networks are scale-free, i.e., their node degree distribution follows the power law. Nodes of the highest degrees, called centers, are highly vulnerable to attacks. Elimination of these nodes seriously degrades the overall performance of network services.

Redundancy is the key to provide services in the face of attacks or failures. Survivability, based on the experience of fault tolerance, assumes various techniques of protection and restoration. Protection in our model is based on the pre-planned backup path for each active (working) end-to-end path [7, 10]. Depending on allowable costs, protecting paths may be either dedicated or shared. This is typical for all survivable networks. The scale-free networks, however, need a special treatment.

Centers in such networks are connected to many other nodes by links of high capacities, switch large amount of data and are of great degree. They are excellent goals of malicious attacks, performed by intruders getting the maximum destructive effect at minimum cost.

An example of a scale-free network is shown in Fig. 1.

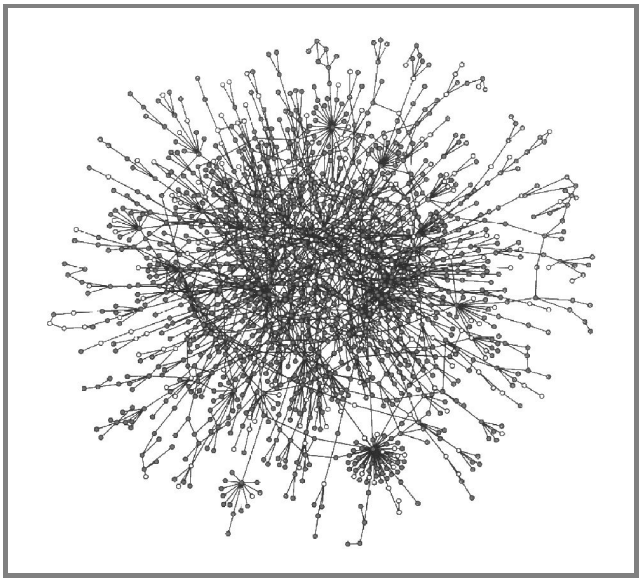


Fig. 1. An example of a scale-free network.

Centers exist in many networks. It has been proved, that the uncontrolled growth of a network leads to a power law distribution of node degrees ($P(k) \sim k^{-\gamma}$) [2]. Figure 2 presents the degree distribution $P(k)$ of scale-free networks compared to random topologies (having the Poisson degree distribution).

By uncontrolled growth we mean adding the new elements according to the preferential attachment rule. Following such a rule, network nodes are mostly being attached to the already highly connected ones. This phenomenon, often

referred to as the *rich get richer process*, causes networks to become scale-free.

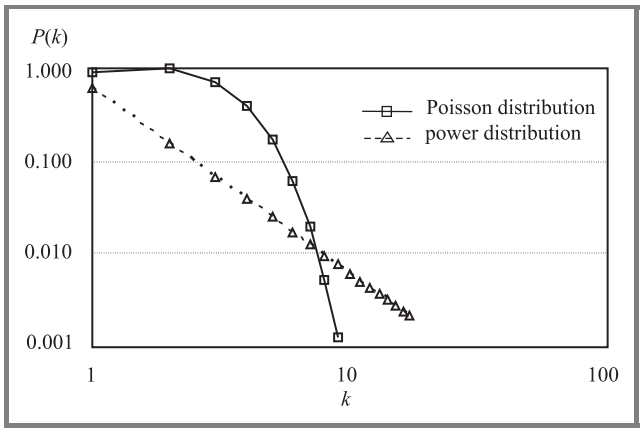


Fig. 2. Comparison of Poisson and power law distributions.

It has been proved that the topology of many wide-area networks is scale-free [1, 4, 6, 8]. Such topologies can be also found in other fields of science, for instance in chemistry or biology. Some examples of power law distribution that are worth mentioning can be found for the stock-price fluctuations, movie actor collaboration networks, biological cellular networks, scientific citation networks and many others.

If connections are established with help of the shortest path algorithm and the standard distance metrics, nodes of extraordinary high degree are often the transit elements of the installed paths. We define the parameter named *betweenness centrality* with regard to a certain node k, denoted $BC(k)$ as [5]:

$$BC(k) = \sum_{p \neq q} \frac{\delta_k(p,q)}{\delta(p,q)}, \tag{1}$$

where:

$\delta(p,q)$ is the number of shortest paths between nodes p and q,

$\delta_k(p,q)$ is the number of shortest paths between nodes p and q that run through a vertex k.

Each central node has a high value of BC, because many shortest paths established between various pairs of nodes (p,q) traverse such nodes. Attacks on centers may thus cause many connections to fail.

In this paper we propose a model, which, based on traffic parameters of a demand, allows to establish the survivable end-to-end paths in scale-free networks. Specific properties of scale-free networks imply the choice of a new metrics of the cost of a link and new algorithms finding paths, reducing the number of connections broken due to an attack.

The rest of the paper is organized as follows. Section 2 presents the model of establishing survivable connections with regard to the requested level of continuity of service in scale-free networks. Section 3 presents the integer linear programming (ILP) formulation of the problem and the corresponding heuristic algorithm. Modeling results, obtained for an artificial scale-free network, are shown in Section 4 and include: the probability of demand rejection, length of active path, the number of broken connections and the values of restoration time.

2. Model description

The main objective of our model is to establish survivable connections in the way that the number of connections broken due to an attack on a node is reduced. The decrease in number of broken connections is to be greater with the increase of the required level of service. It can be achieved with help of a metrics that, depending on the requested class of service, forces active paths to bypass nodes of high degree.

We consider the network consisting of N nodes and A directed arcs. Nodes and arcs are numbered:

$n-1, \ldots, N$;

$a = 1, \ldots, A$.

We assume $K \leq N(N-1)$ demands. We denote:

$p_k(q_k)$	source (destination) of a demand k; $k = 1, 2, \ldots, K$
c_a	capacity of an arc a
Λ_a	number of channels of an arc a
$\prod_{p,q}$	an active path from p_k to q_k
$\overline{\prod}_{p,q}$	a backup path from p_k to q_k
$c\left[\prod_{p,q}\right]$	the requested capacity for a connection from p_k to q_k
$\overline{C}_a$	spare capacity on an arc a

M classes of service are assumed, numbered from 0 to $M-1$. Class 0 represents demands for which the probability of breaking the connections must be minimized. With the increase of a class number, the requested level of service continuity gets lower. Various types of traffic of many network architectures (e.g., ATM) can be mapped onto the proposed classes, which makes the approach widely applicable.

In order to find active paths on a shortest-path basis, we propose the metrics that determines the cost κ_a a of an arc a, as shown in Eq. (2):

$$\kappa_a^m = \begin{cases} \dfrac{m}{M-1} d_a^* + \dfrac{(M-1)-m}{M-1} BC^*(n) & \text{if } c\left[\prod_{p,q}\right] \leq \overline{C}_a \\ \infty & \text{otherwise} \end{cases}, \tag{2}$$

where:

m is the current class of service; $m = 0, 1, 2, \ldots, M-1$,

d_a^* is the normalized length of an arc a:

$$d_a^* = \frac{d_a}{\max\limits_a (d_a)}. \tag{3}$$

$BC^*(n)$ is the normalized value of the betweenness centrality of a node n:

$$BC^*(n) = \frac{BC(n)}{\max\limits_{n}\left(BC(n)\right)}. \tag{4}$$

If there is not enough spare capacity on an arc a to install the active path (according to the requested capacity $c[\prod_{p,q}]$), then the cost of using this arc is set to infinity. Otherwise the cost is the weighted sum of the normalized length of an arc (d_a^*) and the value of the normalized betweenness centrality coefficient $BC^*(n)$ of the end node n of an arc a. Two boundary cases are worth explanation:

- Class 0 for demands of the highest quality of service. Here the cost of the arc is calculated only on the basis of the value of $BC^*(n)$. This results in installing paths that omit central nodes. This causes the connections of class 0 to have a low probability of breaking. However, the installed active paths are not the shortest ones.

- Class $M-1$ for connections that do not require the guarantee of continuity. For them the cost of each arc is determined by Eq. (2) and is thus equal to:

$$\kappa_a^{M-1} = \begin{cases} d_a^* & \text{if } c\left[\prod_{p,q}\right] \le \overline{C}_a \\ \infty & \text{otherwise} \end{cases}. \tag{5}$$

  Here, the active paths of connections are the shortest ones in the sense of distance but often run through central nodes and are thus exposed to attacks.

For classes $1,2,\ldots,M-2$ characteristics of active paths with regard to their length and vulnerability to attacks are expected to be the compromise of the corresponding features of classes 0 and $M-1$, respectively.
Backup paths are found using the standard distance metrics. In order to allow fast restoration of connections of each class in case of a failure, the backups are computed as the shortest ones.

3. Methods used to compute active and backup paths

In the following part of the paper, the protection against a single node failure is assumed. All backup paths are dedicated (not shared). There is only one backup path for each connection (path protection model).

3.1. The ILP formulation of the problem

We propose using the following nomenclature:

N set of nodes of a network

A set of directed link (arcs) in a network

K number of source-destination (demand) pairs of nodes, $K \le N(N-1)$

p_k source node of a demand k

q_k destination node of a demand k

$\alpha_{k,a}^{\lambda}$ takes value of 1 if a channel λ of an arc a is used by an active path of a demand k; 0 – otherwise

$\beta_{k,a}^{\lambda}$ takes value of 1 if a channel λ of an arc a is used by a backup path of a demand k; 0 – otherwise

λ_a a capacity of an arc a represented by the number of channels: $\forall_a \lambda_a = \Lambda$

$\kappa_{k,a}^{m}$ the channel cost in an arc a calculated by considering the class m of a demand k and the length of an arc according to Eq. (2) (for active paths)

$s_{k,a}$ the channel cost in an arc a calculated for a demand k with regard to its length (for backup paths)

x vector of all components of flows (variables)

It is to find paths transporting required flows from sources to destinations, protecting them against a single node failure and minimizing the linear cost:

$$\varphi(x) = \sum_{k=1}^{K}\sum_{a=1}^{A}\sum_{\lambda=1}^{\lambda_a}\left(\kappa_{k,a}^{\lambda}\cdot\alpha_{k,a}^{\lambda} + s_{k,a}\cdot\beta_{k,a}^{\lambda}\right) \tag{6}$$

subject to constraints given in Eqs. (7)–(15).

a) *Capacity constraints (on the number of the available wavelengths on an arc a)*:

$$\sum_{\lambda=1}^{\lambda_a}\sum_{k=1}^{K}\left(\alpha_{k,a}^{\lambda} + \beta_{k,a}^{\lambda}\right) \le \lambda_a. \tag{7}$$

b) *The flow balance constraints for each wavelength λ and for each demand k*:

For a source node of an active path:

$$\sum_{\{a:a\equiv(p_k,j);j=1,2,K,N;j\neq p_k\}}\alpha_{k,a}^{\lambda} - \sum_{\{a:a\equiv(i,p_k);i=1,2,K,N;i\neq p_k\}}\alpha_{k,a}^{\lambda} = 1. \tag{8}$$

For a destination node of an active path:

$$\sum_{\{a:a\equiv(q_k,j);j=1,2,K,N;j\neq q_k\}}\alpha_{k,a}^{\lambda} - \sum_{\{a:a\equiv(i,q_k);i=1,2,K,N;i\neq q_k\}}\alpha_{k,a}^{\lambda} = -1. \tag{9}$$

For transit nodes of an active path:

$$\sum_{\{a:a\equiv(i,j);j=1,2,K,N;i,j\neq p_k,i,j\neq q_k\}}\alpha_{k,a}^{\lambda} - \sum_{\{a:a\equiv(i,j);i=1,2,K,N;i,j\neq p_k,i,j\neq q_k\}}\alpha_{k,a}^{\lambda} = 0. \tag{10}$$

For a source node of a backup path:

$$\sum_{\{a:a\equiv(p_k,j);j=1,2,K,N;j\neq p_k\}}\beta_{k,a}^{\lambda} - \sum_{\{a:a\equiv(i,p_k);i=1,2,K,N;i\neq p_k\}}\beta_{k,a}^{\lambda} = 1. \tag{11}$$

For a destination node of a backup path:

$$\sum_{\{a:a\equiv(q_k,j);j=1,2,K,N;j\neq q_k\}} \beta_{k,a}^{\lambda} - \sum_{\{a:a\equiv(i,q_k);i=1,2,K,N;i\neq q_k\}} \beta_{k,a}^{\lambda} = -1. \tag{12}$$

For transit nodes of a backup path:

$$\sum_{\{a:a\equiv(i,j);j=1,2,K,N;i,j\neq p_k,i,j\neq q_k\}} \beta_{k,a}^{\lambda} - \sum_{\{a:a\equiv(i,j);i=1,2,K,N;i,j\neq p_k,i,j\neq q_k\}} \beta_{k,a}^{\lambda} = 0. \tag{13}$$

c) *Constraints assuring nodal-disjointness of active and backup paths*:

$$\sum_{\lambda=1}^{\lambda_a} \sum_{\{a:a\equiv(i,j);j=1,2,K,N;j\neq i;i\neq p_k\}} \left(\alpha_{k,a}^{\lambda} + \beta_{k,a}^{\lambda}\right) \leq 1, \tag{14}$$

$$\sum_{\lambda=1}^{\lambda_a} \sum_{\{a:a\equiv(i,j);i=1,2,K,N;i\neq j;j\neq q_k\}} \left(\alpha_{k,a}^{\lambda} + \beta_{k,a}^{\lambda}\right) \leq 1. \tag{15}$$

Constraint, given in Eq. (7), assures that the total number of channels, reserved for survivable connections on an arc a, will not exceed the capacity of this arc. For each channel and each demand, flow balance for the active paths is assured by Eqs. (8)–(10). For instance, Eq. (8) guarantees that there is only one active path outgoing from the source node s_k. Equation (10) simply states that transit nodes do not store traffic. Equations (11)–(13) describe the flow balance constraints for backup paths, respectively.

3.2. The SACC heuristic algorithm

SACC algorithm of establishing survivable and attack-compliant connections

Input:

- A pair of source and destination nodes $[p_k, q_k]$
- Requested capacity $c\,[\prod_{p,q}]$
- Requested class of service m
- Number of classes of service M

1. Find the active path of a connection:
   - 1.1. For each arc a calculate its cost as defined in Eq. (2)
   - 1.2. Find the shortest path between nodes p_k and q_k, using the matrix of costs C, evaluated in 1.1
   - 1.3. If the active path is found then install it, else go to Step 3
2. Find the backup path:
   - 2.1. For each arc a evaluate its cost as defined in Eq. (5)
   - 2.2. In order to assure the nodal disjointness of active and backup paths of a connection, set the costs of active path's arcs as well as arcs incident to active path's nodes to infinity
   - 2.3. Find the shortest path between nodes p_k and q_k, using the matrix of costs C, evaluated in 2.1
   - 2.4. If the backup path is found then install the path
3. If any of the two paths cannot be found due to the lack of spare resources, then reject the demand and remove the active path (if installed), else establish the connection

The main advantage of using heuristic methods over ILP approach is their polynomial complexity. SACC algorithm, described below, uses a Dijkstra's algorithm to find a shortest path between a pair of a source and destination nodes.

4. Modeling results

In this section we focus our research on measuring the probability of demand rejection, the average length of active path, the number of broken connections and restoration times for various classes of service. The scale-free network, shown in Fig. 3, used in research, was generated by Pajek software.

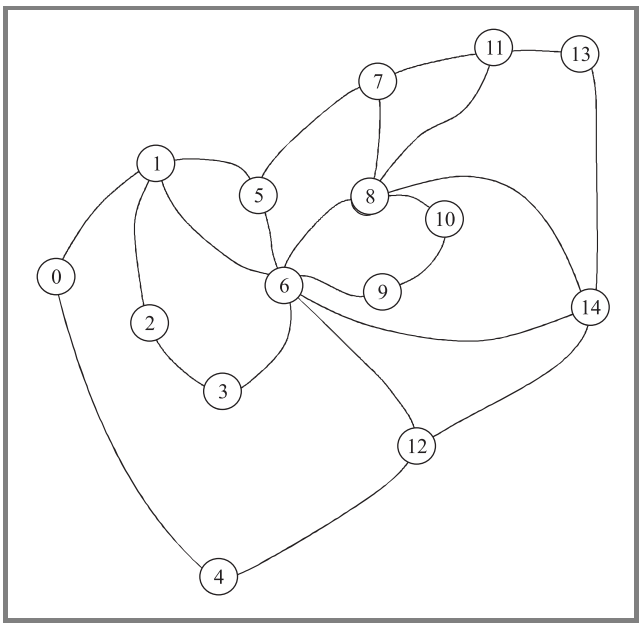


Fig. 3. The SFN artificial scale-free network.

According to the values of the normalized betweenness centrality parameter, given in Table 1, nodes: 1, 6, 8, 14 having high values of BC^* can be referred to as centers.

Table 1
Values of the normalized betweenness centrality for nodes of the SFN network

Node k	0	**1**	2	3	4	5	**6**
$BC^*(k)$	0.063	0.367	0.023	0.077	0.046	0.135	1.000

7	**8**	9	10	11	12	13	**14**
0.085	0.462	0.081	0.035	0.058	0.213	0.027	0.300

All the paths for survivable connections were obtained with help of AMPL/CPLEX environment as well as with dedicated network simulator, implemented in C++ environment.

Due to the complexity of the optimal algorithm (NP completeness of the investigated ILP formulation) calculation of paths was infeasible for real large networks. Simulation focused on measuring the number of broken connections and restoration times was then run with the use of our dedicated network simulator.

The following properties were assumed for all the directed links of the network:

– equal number of available channels (here 8);

– equal capacity of all channels;

– equal length (1890 km).

For each connection we assumed:

– class of service chosen randomly out of M available classes;

– metrics given in Eq. (2) in all active path computations;

– standard distance metrics, given in Eq. (5), in all backup path computations;

– a demand of resource allocation equal to the capacity of one channel of a link;

– protection against a single node failure.

Connections were broken due to attacks on nodes (one each time). The probability of each node to be attacked was assumed to be proportional to the values of $BC^*(k)$. During each experiment for the SFN network, 30 logical topologies were generated. Each logical topology is a graph of a fixed (here 10) number of randomly chosen pairs of source and destination nodes. For each logical topology, failures of 100 nodes were generated to measure the numbers of broken connection and values of restoration time.

4.1. The ILP results

Table 2 shows results of path computation obtained by solving the optimization problem stated in Subsection 3.1 with help of AMPL/CPLEX environment.

It was to find paths for connections of 3 classes of service available. The results prove that smaller the class number was (meaning higher requested level of service continuity), more active paths omitted nodes of high degree. For instance a connection between nodes 1 and 14 of the highest (0) class of service, was realized by establishing a long (but omitting centers; here 6, 8) active path: (1, 5, 7, 11, 13, 14) and a much shorter backup path (1, 6, 14). On the contrary, a connection of the lowest class (here 2), between nodes 0 and 4, was realized by a short active path (0, 4) and a much longer backup path (0, 1, 6, 12, 4).

Table 2

Example paths for 10 survivable connections of 3 classes of service (0, 1 and 2), established according to the proposed model

Connection:	(0,4)		priority:		2	
Active path:	0	4				
Backup path:	0	1	6	12	4	
Connection:	(0,11)		priority:		1	
Active path:	0	1	5	7	11	
Backup path:	0	4	12	6	8	11
Connection:	(1,12)		priority:		1	
Active path:	1	6	12			
Backup path:	1	0	4	12		
Connection:	(1,14)		priority:		0	
Active path:	1	5	7	11	13	14
Backup path:	1	6	14			
Connection:	(2,7)		priority:		1	
Active path:	2	1	5	7		
Backup path:	2	3	6	8	7	
Connection:	(2,11)		priority:		0	
Active path:	2	1	5	7	11	
Backup path:	2	3	6	8	11	
Connection:	(5,11)		priority:		1	
Active path:	5	7	11			
Backup path:	5	6	8	11		
Connection:	(5,13)		priority:		2	
Active path:	5	7	11	13		
Backup path:	5	6	14	13		
Connection:	(6,8)		priority:		2	
Active path:	6	8				
Backup path:	6	14	8			
Connection:	(0,14)		priority:		2	
Active path:	9	6	14			
Backup path:	9	10	8	14		

4.2. Probability of demand rejection

We considered the phenomenon of demand rejection due to all possible reasons. The main causes included:

– lack of available link channels;

– topology bottlenecks.

Figure 4 shows the probability of demand rejection for 3 classes of service, while Fig. 5 – for 6 classes of service, respectively.

Comparing results for M classes of service, we see that CPLEX served all the demands. This is due to the fact that only 10 demands were defined for each logical topology and there was enough resources to serve all of them. In contrast, our heuristic algorithm found local optima and,

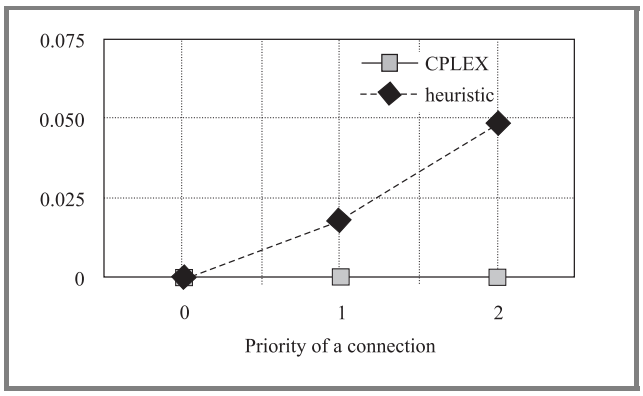


Fig. 4. Probability of demand rejection for 3 classes of service.

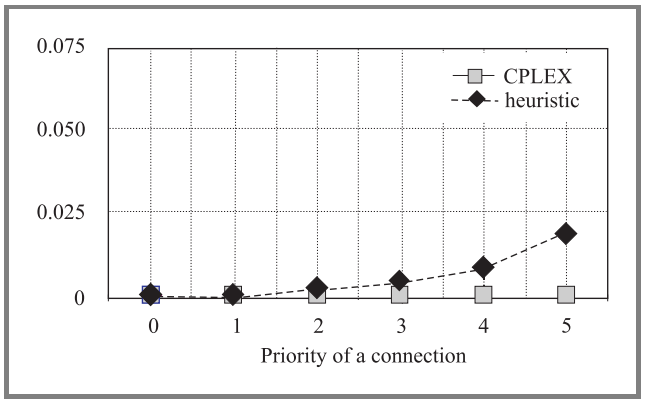


Fig. 5. Probability of demand rejection for 6 classes of service.

due to bottlenecks, some of the demands were rejected. There is no significant difference in the total number of rejected demands between 3 and 6 classes of service.

4.3. Number of active path links

Figures 6 and 7, and Table 3 show the numbers of active path links as the function of the connection class of service. When increasing the level of the requested continuity of service of a connection (decreasing the class number), its active path becomes longer. This is because it omits nodes of high degree. In the worst case for 3 classes of service, the average length of active path obtained for the class 0, was about 1.31 times worse than for the class 2.

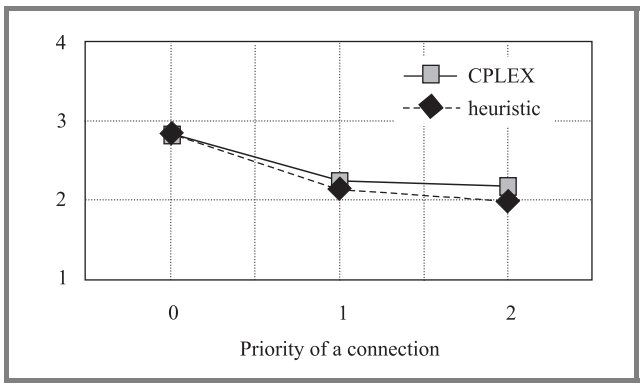


Fig. 6. Average number of active path links for 3 classes of service.

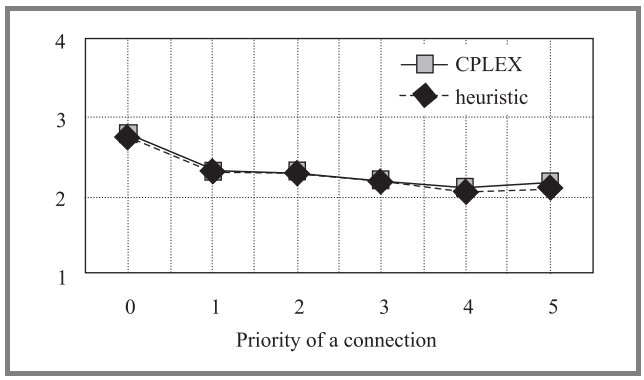


Fig. 7. Average number of active path links for 6 classes of service.

Table 3
Average number of active path links for 3 and 6 classes of service (CPLEX)

Number of service classes 3				
Class number		0	1	2
Active path length [links]	(CPLEX)	2.84	2.23	2.16
	(heuristics)	2.84	2.15	1.98
95% confidence interval [links]	(CPLEX)	0.48	0.27	0.27
	(heuristics)	0.49	0.26	0.26

Number of service classes 6							
Class number		0	1	2	3	4	5
Active path length [links]	(CPLEX)	2.79	2.32	2.33	2.22	2.12	2.19
	(heuristics)	2.77	2.32	2.30	2.20	2.06	2.09
95% confidence interval [links]	(CPLEX)	0.69	0.50	0.46	0.38	0.37	0.33
	(heuristics)	0.67	0.51	0.46	0.39	0.38	0.33

Typically, heuristic algorithms give worse results than the respective ILP ones. Generally, this remains true for our SACC algorithm. However, due to the non-zero probability of rejecting the demands for the SACC algorithm (Figs. 4 and 5), the obtained active paths turned out to be shorter than the respective ILP ones. This was due to the smaller network congestion, obtained when SACC algorithm was used, as it established less connections than the ILP algorithm. This feature caused similar effect regarding the aggregate number of broken connections and the total restoration time, described in the next subsections.

4.4. Number of broken connections

Figures 8 and 9 show the aggregate numbers of broken connections as the function of the connection class of service. They prove that the proposed model results in a significant decrease in the number of broken connections. The higher the requested level of service continuity is, the decrease

in the number of broken connections gets more visible. In the best case, observed for 6 classes of service, about 67% less connections were broken for the class 0, compared to the results for the class 5.

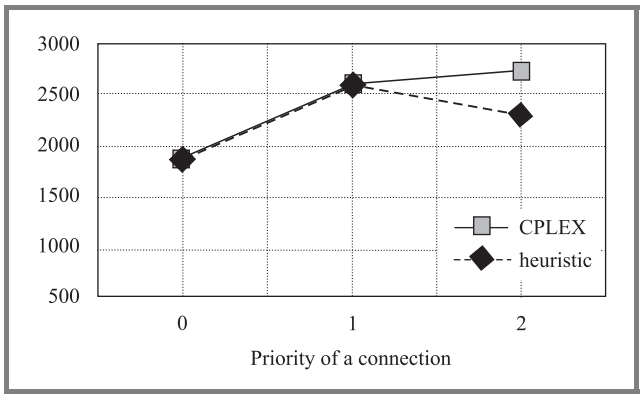


Fig. 8. Aggregate number of broken connections for 3 classes of service.

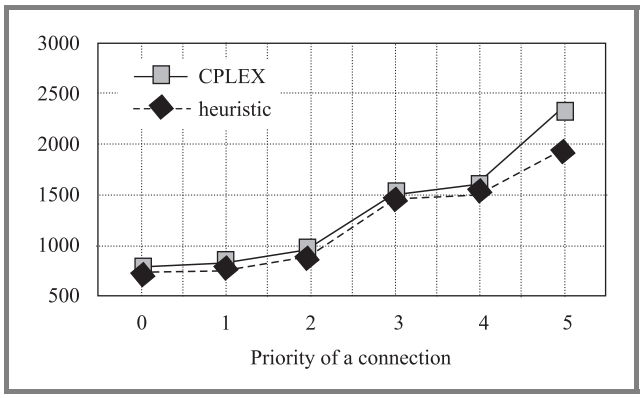


Fig. 9. Aggregate number of broken connections for 6 classes of service.

4.5. Restoration time

Figures 10 and 11 show the values of the average restoration time as the function of the connection class of service. They represent the time needed to restore a connection after

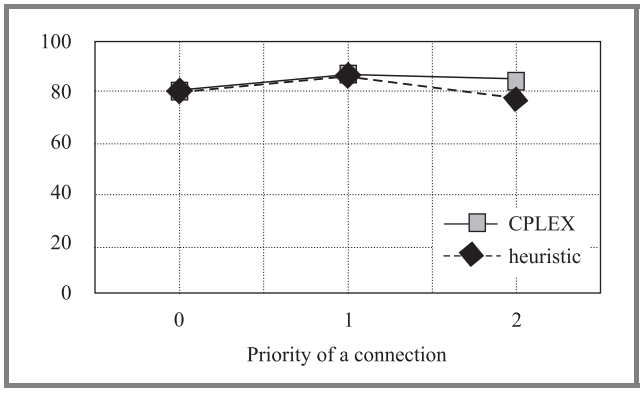


Fig. 10. Average restoration time [ms] for 3 classes of service.

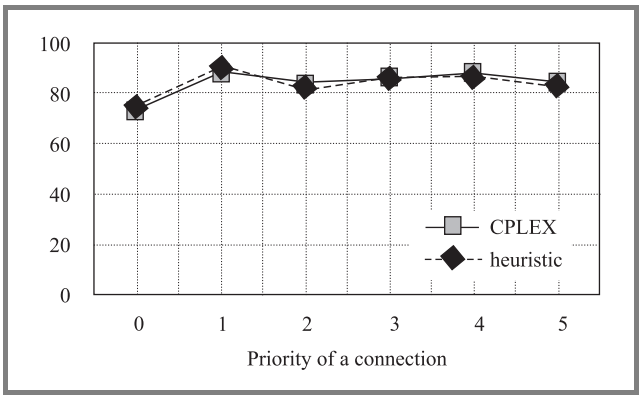


Fig. 11. Average restoration time [ms] for 6 classes of service.

a failure of a node, according to the protocol described in [11]. They prove that for the SFN network the value of restoration time does not depend much on the service class. This dependency is, however, very remarkable when analyzing the aggregate value of restoration time. Such an aggregate value for a given class k is calculated as the sum of all restoration times for connections of class k during one experiment.

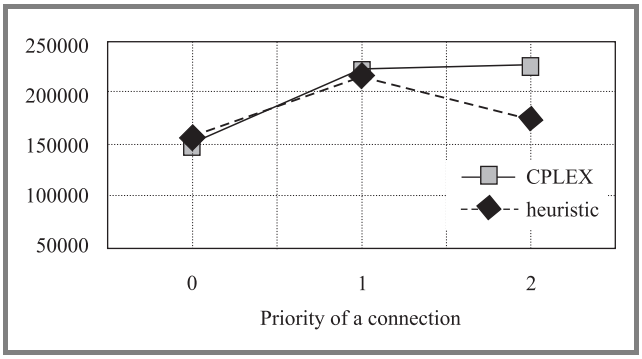


Fig. 12. Aggregate restoration time [ms] for 3 classes of service.

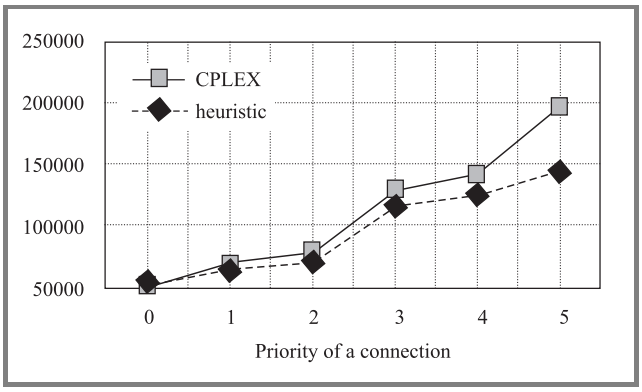


Fig. 13. Aggregate restoration time [ms] for 6 classes of service.

Figures 12 and 13 prove that the proposed model leads to a significant reduction in the value of the aggregate restoration time. The higher the requested level of service con-

tinuity is, the aggregate value of restoration time is more reduced (even about 4 times for the class 0, when 6 classes are used).

5. Conclusions

Results obtained by us confirm that the proposed model contributes to a significant reduction in the number of broken connections for classes of high priority. The active paths of such connections omit central nodes and are thus more resistant to attacks. On the other hand, low priority connections are often exposed to attacks. They are, however, realized by the shortest active paths (in the sense of distance), providing shorter values of data transmission delay.

Concluding the paper, we point out that in a scale-free network, one cannot have connections that are simultaneously resistant-to-attack and are realized by short active paths. A short active path of a connection means that with high probability it goes through central nodes and thus is at high risk of breaking.

However, we claim that establishing attack-resistant connections, realized by short active paths could be possible for networks of a regular topology. For the scale-free network it would be the best to make it more regular. Topology improvements are, however, beyond the scope of this paper and constitute the subject for future research.

References

[1] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks", *Rev. Mod. Phys.*, vol. 74, pp. 47–97, 2002.

[2] A. Barabási and R. Albert, "Emergence of scaling in random networks", *Science*, vol. 286, pp. 509–511, 1999.

[3] R. Bhandari, *Survivable Networks – Algorithms for Diverse Routing.* Boston [etc.]: Kluwer, 1999.

[4] Q. Chen and D. R. Shi, "The modeling of scale-free networks", *Phys. A*, vol. 335, *Elsev. Sci. B*, pp. 240–248, 2003.

[5] T. Gierszewski and J. Rak, "Scale-free networks", Scientific Report, Gdańsk University of Technology, 2004.

[6] K.-I. Goh, E. Oh, B. Khang, and D. Kim, "Classification of scale-free networks", *Proc. Natl. Acad. Sci.*, vol. 99, pp. 12 583–12 588, 2002.

[7] R. Kawamura, "Architectures for ATM network survivability", *IEEE Commun. Surv. Fourth Quart.*, vol. 1, no. 1, 1998.

[8] Z. Liu, Y-Ch. Lai, N. Ye, and P. Dasgupta, "Connectivity distribution and attack tolerance of general networks with both preferential and random attachments", *Phys. A*, vol. 303, *Elsev. Sci. B*., pp. 337–344, 2002.

[9] M. Pióro and D. Medhi, *Routing, Flow and Capacity Design in Communication and Computer Networks*. Amsterdam [etc.]: Morgan Kaufmann, 2004.

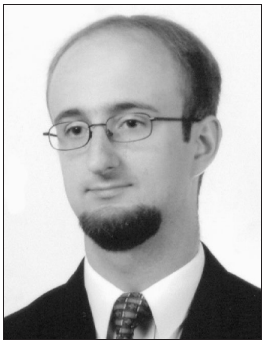
[10] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks", Part I – "Protection", in *Proc. IEEE INFOCOM*, New York, USA, 1999, pp. 744–751.

[11] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks", Part II – "Restoration", in *Proc. IEEE Integr. Circ. Conf. ICC*, Vancouver, Canada, 1999, pp. 2023–2030.

[12] J. W. Suurballe, "Disjoint paths in a network", *Networks*, vol. 4, pp. 125–145, 1974.

Wojciech Molisz joined the Gdańsk University of Technology, Poland, in 1968, where he is employed until now. In 1980, he was invited by the International Telecommunications Union (ITU) to the Institute of Telecommunications, Oran, Algeria, to give lectures on computer networks. From 1991 to 1993, he joined the Nuclear Research Centre, Karlsruhe, Germany, where he was involved in two international research projects in the frames of the ESPRIT programs. In the VOICE II Project (ESPRIT III), he served as the workpackage leader. Doctor of Science W. Molisz is the author and co-author of four monographs and an academic book "Solution Methods of Optimization Problems" and the author of more than 100 papers, reports and conference proceedings. His current interests are in the survivability of broadband networks, especially in the optical communications networks.

e-mail: womol@eti.pg.gda.pl
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Narutowicza st 11/12
80-952 Gdańsk, Poland



Jacek Rak received the M.Sc. degree in computer science from Gdańsk University of Technology (GUT), Poland, in 2003. Since 2003 he has been working as an Assistant at GUT. He is currently working toward the Ph.D. degree in computer science at GUT. His current research areas include: routing, design, dimensioning and analysis of high speed (particularly wavelength routed) backbone networks with focus on survivability.

e-mail: jrak@pg.gda.pl
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Narutowicza st 11/12
80-952 Gdańsk, Poland