Improving Restorability in Radio Access Network

Wee-Seng Soh
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
Email: wsoh@ece.cmu.edu

Zoe Antoniou
Nokia Research Centre
5 Wayside Road
Burlington, MA 01803
Email: zoe.antoniou@nokia.com

Hyong S. Kim
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
Email: kim@ece.cmu.edu

***Abstract*— In the past, wireless network reliability issues have received limited attention. Previously proposed solutions have focused on protection against base station failures, which may require significantly more base stations than necessary. Actual data from wireless carriers suggest that problems in the backhaul are the uttermost reliability problems faced, and is therefore the main focus of this work. Current radio access networks are based on tree and star-like topologies, which have no inherent restorability properties. We propose a heuristic topology enhancement method that adds redundant spans and upgrades existing infrastructure cost-effectively, in order to create partially meshed architectures that could provide the desired level of restorability against single span failure scenarios. Both span and path restoration techniques are explored. The algorithm was tested using several different variants of restoration mechanisms. Results show that the proposed heuristic algorithm is able to achieve reasonably good solutions in a time scale that is several orders of magnitude faster than an optimization approach based on binary integer programming formulation.**

I. Introduction

In recent years, there has been a rapid increase in wireless network deployment and mobile device market penetration. With increasing dependence on mobile devices, there are likely to be demands for wireless networks to offer the same reliability as wireline telecommunications and data networks. While there has been considerable research in wireline network reliability problems, wireless network reliability issues have received limited attention [1]–[5]. This is partly because wireless service providers are still struggling to recover their infrastructure and license costs [6]. Yet, the need to provide reliable wireless access is becoming a critical issue. A recent study by the state of New Jersey indicated that wireless E-911 calls account for 43 percent of all E-911 calls [7].

In the following, we provide a brief overview of some existing work on wireless network survivability. In [1], PCS network survivability issues are discussed, and a multi-layer framework for conducting survivability analysis is introduced. In [2], the use of multi-mode terminals that could connect to multiple heterogeneous overlay networks is proposed. In [3], Dahlberg *et al.* proposes overlapping base transceiver station (BTS) coverage areas that provide multiple wireless access link choices for mobile terminals (MTs). In [4], a two-level hierarchical cell-site architecture is proposed; a macrocell encompasses multiple smaller cells, and acts as a backup system to pick up traffic from any failed cell within its coverage. In [5], the use of radio repeaters at cell edges to redirect a failed cell's traffic to its neighboring BTSs is proposed. One common characteristic among the solutions proposed in [2]–[5] is that they are capable of surviving from a BTS failure. These solutions might be costly due to the high BTS/repeater density requirement.

In every cellular network, there exists a radio access network (RAN), which mainly consists of geographically dispersed BTSs, and network controllers, such as the base station controllers (BSCs) in GSM networks, or the radio network controllers (RNCs) in WCMDA networks. The transmission facilities (also known as backhaul) in existing RANs and upcoming 3G RANs are typically based on star, tree and chain topologies (see Fig. 1). These structures are highly susceptible to any form of failure; when a failure occurs, wireless access in one or more cells may be lost. According to [6], problems in the backhaul are the uttermost reliability problems faced by wireless carriers. Therefore, unlike previously proposed solutions that focus on protection against BTS failures, we shall focus on protection against backhaul failures.

In this paper, a heuristic algorithm is introduced to upgrade the existing RAN topologies cost-effectively, so as to improve their restorability against single span failure scenarios. The proposed approach suggests the addition of redundant spans to current RAN topologies, as well as, the upgrade of existing spans, in order to create partially meshed and restorable networks. Both span and path restoration techniques have been considered as candidate mechanisms to bypass the failed element. Several tests were also performed on an arbitrary RAN topology, and results are presented. Finally, a solution generated by the algorithm from one of the tests is also compared with those discovered by two commercial solvers.

II. Improving Restorability

Tree and star-like topologies, most commonly found in today's RANs, are very sensitive to any kind of failures. Since mobile operators have estimated that 30–50 percent of their operating costs are associated with the backhaul [8], any strategy that is developed to enhance restorability must be cost-effective. Given the high cost of access capacity and the large number of BTSs in a typical RAN, it might be very costly to use self-healing rings, because they require high-capacity spans in every hop. On the other hand, the use of

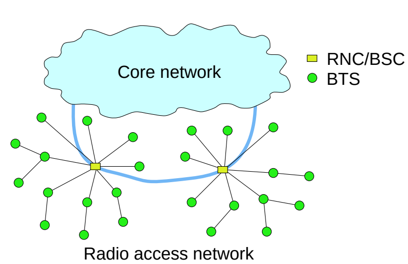


Fig. 1. The traditional architecture of the radio access network.

partially meshed architectures to achieve restorability has several advantages. It enables the reuse of existing infrastructures, and requires lower capacity spans. This makes possible the use of affordable microwave links and low capacity leased lines, as opposed to the more costly option of installing high capacity fibers. Another distinct advantage is that incremental protection can be provided as new spans are added gradually.

In practice, a mesh restorable network can be fully protected against any single element failure when sufficient spares and spans are added to the original network. The challenge is to realize the desired level of protection while minimizing the upgrading costs. This problem can usually be formulated as an integer programming (IP) problem, which can be solved using general-purpose solvers. However, the computational requirement can easily exceed the capability of current computers, because the number of constraints and variables increases very rapidly with the size of the network. Due to the disadvantage of an optimization approach, heuristics are often used, although the solutions obtained may be sub-optimal. In [9], a heuristic algorithm for adding new spans to tree-like RAN topologies is proposed. Among a set of BTSs that violate the desired loss constraint, a BTS is randomly chosen. A new span is then inserted to connect it to one of its neighboring BTSs using a decision rule based on the cost and availability changes that will result from the insertion. The backup paths of the original BTS and all its child BTSs are assumed to be routed through the default working path of the newly connected neighboring BTS. Span capacities along the backup paths are increased accordingly. The procedure is repeated until all BTSs meet the loss constraint. Although the above approach is able to improve restorability, the cost of the resulting design may be far from optimum. As is shown in [9], the subsequent addition of new spans may render some of the previously allocated capacity unnecessary. However, there is no mechanism in [9] that identifies and removes such redundancy.

A two-phase heuristic algorithm proposed in [10] for wireline networks appears to be more attractive. The first phase, known as *forward synthesis*, is a greedy approach that repeatedly adds spare capacity to the network where the greatest increase in restorability will result. The second phase, known as *design tightening*, removes any unnecessary redundancy from the design generated by the first phase. It is shown in [10] that it could generate near-optimal designs. The algorithm proposed in this paper uses [10] as a starting point, and has numerous enhancements with many practical considerations, which are described in the following subsections.

A. Potential Spans and Candidate Transmission Capacities

The connectivity of an existing RAN can be enhanced by adding extra spans between some of the nodes that do not have direct connection. These feasible candidate locations are called *potential spans*. Many existing spare capacity assignment (SCA) algorithms are designed for wireline networks and only consider upgrading existing spans to improve restorability. Our proposed algorithm simultaneously considers the addition of new spans, as well as the upgrading of existing spans.

In a real network, there are many constraints that dictate whether a potential span may be installed between any given pair of nodes, as well as, the set of candidate transmission capacities that may be used in the span. Examples of such constraints include leased line availability, terrain characteristics (e.g., line-of-sight for microwave links), licensing restrictions, distance, antenna space, technological constraints (e.g., interference between microwave channels), and so on. As a result, the set of candidate capacities at each potential span may be different. Also, they usually do not come in contiguous multiples of a common denomination (T1/E1). For example, it is common for microwave capacities in Europe to support 2×E1, 4×E1, 8×E1, 16×E1, E3, STM-0, STM-1, etc. [12]. In contrast, previous SCA algorithms usually assume that all integral multiples of a basic modular unit are valid options.

Besides the potential spans, those existing working spans with upgradable transmission capacities must also be identified. The algorithm needs to select a subset of the potential spans and upgradable working spans, and specify the corresponding transmission capacities that should be used in each span. The selections must be performed in a cost-effective manner, while satisfying the desired restorability target.

B. General Cost Model

The proposed algorithm is able to accept a general cost model as its objective function. In contrast, [10] associates cost with the total spare capacity in the network. In our algorithm, a unique cost can be associated with each span's candidate transmission capacity. It can be chosen to reflect the true monetary cost to use the transmission facility over a certain period of time. It is important to realize that the cost of the same capacity in different spans may differ.

Another advantage of accepting a general cost model is that the algorithm can exploit the presence of any economy-of-scale effects in the transmission capacity costs. This is especially important for the case of microwave links, as the capacity costs are expected to be nonlinear, such that a doubling in capacity may only result in a fractional increase in cost. With economy-of-scale effects, it may appear more attractive to aggregate restoration paths through some common spans to form high-capacity spans, rather than spreading the restoration paths that would otherwise require a large number of low-capacity spans. This may sometimes lead to longer restoration paths than those achieved using a linear cost model. The sparsening effect becomes more prominent as the economy-of-scale strengthens. By taking such effects into account, a solution that has a larger total spare capacity but a lower cost may still be recognized as a better solution over one that merely minimizes the total spare capacity with no regard for the actual cost.

C. Restoration Mechanisms

The algorithm in [10] only considers *span restoration*, in which rerouting occurs between the immediate end nodes of the span that fails. Span restoration is fast because the node responsible for triggering the recovery process is the one that detects the failure. However, it is widely known that span restoration is usually not as capacity-efficient as *path restoration* [11], in which the rerouting may occur anywhere between the source and destination nodes to bypass the span that fails. Our proposed algorithm is able to accommodate both span restoration and path restoration.

For span restoration, [10] only considers split protection paths, in which a severed span's traffic may be rerouted via multiple restoration paths using any granularity. The proposed algorithm, on the other hand, allows the following variants:

1) *No granularity constraint* (NG): every affected working path may be rerouted via multiple paths (same as [10]).
2) *Granularity of a working path* (WG): each affected working path is rerouted via a single path.
3) *Granularity of an entire span* (SG): all affected working paths are rerouted as a bundle via a single path.

For path restoration, there is usually a choice between *failure-independent* (FI) and *failure-dependent* (FD) schemes. In a FI scheme, each working path only has one backup path, which is span-disjoint from the working path. In a FD scheme, each working path may have more than one backup path, where different backup paths may be used to bypass different span failures. In [11], it is determined that the reduction in capacity requirement from using the FD scheme may be insignificant for sparse networks. Since the RAN is rather sparse, we only implement the FI scheme.

III. Description of Heuristic Algorithm

The proposed algorithm consists of two phases, namely, *forward synthesis* (FS) and *design tightening* (DT). Numerous enhancements have been made to accommodate the practical considerations described in Section II.

We shall first define the metric *restorability*. For span restoration, it is calculated as:

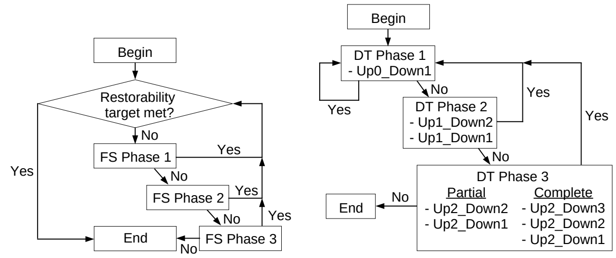
$$\text{Restorability(span restoration)} = \frac{\sum_{i \in \mathcal{S}_{\text{work}}} \rho_i}{\sum_{i \in \mathcal{S}_{\text{work}}} w_i}, \qquad (1)$$

where ρ_i is the restorable traffic when span i is cut, w_i is the working traffic requirement of working span i, and $\mathcal{S}_{\text{work}}$ is the set of all working spans in the RAN. For path restoration, it is calculated as:

$$\text{Restorability(path restoration)} = \frac{\sum_{p \in \mathcal{P}_{\text{work}}} \rho_p}{\sum_{p \in \mathcal{P}_{\text{work}}} w_p}, \qquad (2)$$

where ρ_p is the restorable traffic when path p is disrupted, w_p is the working traffic requirement of working path p, and $\mathcal{P}_{\text{work}}$ is the set of all working paths in the RAN.

An important utility that both FS and DT utilize repeatedly is a function known as *Restorability*(). It is used to compute the RAN's current restorability, using either (1) or (2) (depending on whether span or path restoration is selected). Every time it is called, it needs to identify all the feasible restoration paths corresponding to each span failure, before it can compute the overall network restorability. In order to speed up the process, the function utilizes pre-computed *path tables*. These path tables are different for span and path restoration. For span restoration, each span will have a corresponding path table that includes all the topologically possible restoration routes for that span. For FI path restoration, a path table is associated with each source-destination pair, and it includes all the topologically possible restoration routes that are span-disjoint from the original working path. Note that these candidate routes consist of both working and potential spans. In addition, only routes that fall within a specified hop limit H are recorded. The candidate restoration paths within each path table are sorted so that they will be searched in increasing hop length. Whether a candidate path is feasible when *Restorability*() is called shall depend on the current spare allocation in the RAN.



(a) (b)

Fig. 2. Flowchart for (a)forward synthesis phase, (b)design tightening phase.

Having described the utility that computes the RAN's restorability, the FS and DT phases are described next.

A. Forward Synthesis Phase (FS)

Fig. 2(a) shows the flowchart for the FS phase. The *restorability target* is a user input which must be specified. The FS first checks if the desired restorability target has been met. If not, it enters phase 1, in which a greedy search is performed for a single span within the RAN that can be upgraded to yield the steepest ascend in the restorability *vs*. cost curve. For each candidate span, the algorithm examines all its feasible capacity choices that are larger than its current capacity. In contrast, the algorithm in [10] only considers adding a fixed capacity size to a candidate span, and uses the best increase in restorability as its selection criterion, with no regard for the actual cost of the upgrade. Also, our algorithm examines both potential spans and working spans for possible upgrading, whereas [10] does not have the notion of potential spans.

If a span is found in phase 1 using the selection criterion, it is permanently upgraded. If the restorability target is not yet met, the algorithm repeats phase 1. If no single span can be upgraded to increase restorability within phase 1, the algorithm enters phase 2, and performs a greedy search for a two-span combination that leads to the steepest ascend in the curve. If an improvement is achieved in phase 2, the two spans are permanently upgraded. The algorithm will return to phase 1 if the restorability target has not been met. On the other hand, if phase 2 does not yield any improvement in restorability, the algorithm will enter phase 3.

In phase 3, the algorithm randomly picks a span that cannot be restored. Starting from its shortest candidate restoration path, the algorithm attempts to upgrade all bottleneck spans along this path. If all bottleneck spans along this path can be upgraded, the algorithm returns to check the restorability target. On the other hand, if any of the bottleneck spans is not upgradable, the algorithm checks the next candidate restoration path and so on until a feasible path is found for upgrading. If no such path is found, the algorithm repeats the procedure for a different span that cannot be restored. If no such path can be found for any of the remaining spans that still cannot be restored, the algorithm reaches a stalling point, and exits. When this happens, the RAN has reached its maximum restorability, although it does not meet the restorability target. In order to further improve its restorability, additional potential spans and larger capacity options must be included for consideration.

B. Design Tightening Phase (DT)

Fig. 2(b) shows the flowchart for the DT phase. The objectives of the DT phase are to remove any unnecessary spare capacity in the network, as well as, to swap a more expensive combination of span capacity assignment with a less costly one, while clamping the restorability at the final level achieved by the FS phase.

In phase 1 *Up0_Down1*, the algorithm looks at all spans in the network, one at a time, and determines if its capacity can be reduced to the next smaller size without reducing the restorability. Among those spans that satisfy this criterion, the span that yields the largest decrease in cost is reduced in size. If no such span can be found, the algorithm enters phase 2.

In phase 2, the algorithm first attempts *Up1_Down2*. It searches for a 3-span combination in the RAN such that if one span's capacity is upgraded to the next larger size, while the other two spans are downgraded to their next lower size, the cost of the network will decrease by the largest amount without reducing the restorability. Note that the search space of *Up1_Down2* also includes that of *Up1_Down1*. Therefore, if *Up1_Down2* does not yield any decrease in cost, but *Up1_Down1* does, then the latter's solution will be accepted. *Up1_Down2* has higher priority over *Up1_Down1* whenever a solution exists. If either test generates a solution, the algorithm will return to phase 1 again. Otherwise, it enters phase 3.

In phase 3, there are two options. One option is to run a complete phase 3, which requires a search for a 5-span combination of *Up2_Down3*. This can be time consuming especially for a very large network. Therefore, the algorithm allows the option of partial phase 3, in which only a 4-span combination of *Up2_Down2* is searched. In either case, the larger search space will always contain the smaller search space as before. If no solution is found in phase 3, the algorithm exits and the final capacity assignments have been reached. Otherwise, the algorithm returns to phase 1.

In the original DT phase proposed in [10], only the equivalents of *Up2_Down3*, *Up1_Down2*, and *Up0_Down1* are implemented. Since it regards total spare capacity as objective, and all capacity options are equally spaced, these are the only combinations that may reduce the objective. However, in this paper, we are looking at a more complex problem:

- different spans can have different sets of capacity choices,
- capacity options may be unequally spaced,
- capacity costs may exhibit economy-of-scale effects, and
- capacity costs are location-dependent.

Since the true monetary cost of the network is used as the objective, it is now possible for *Up2_Down1*, *Up2_Down2*, and *Up1_Down1* to reduce the objective.

IV. Tests and Results

A. Test Model

An arbitrary RAN topology with two star-like structures is used to test the designed algorithm (see Fig. 3). Each star consists of 20 BTSs, with capacity requirements chosen randomly from the set $\{2, 4, 6, 8, 10, 12\}$ Mbps. Here, we assume that all traffic requirements are symmetrical. The number above each BTS indicates its capacity requirement. The center of each star, also known as *first-level aggregation node*, is assumed to be connected to high capacity spans (shown as dotted lines)

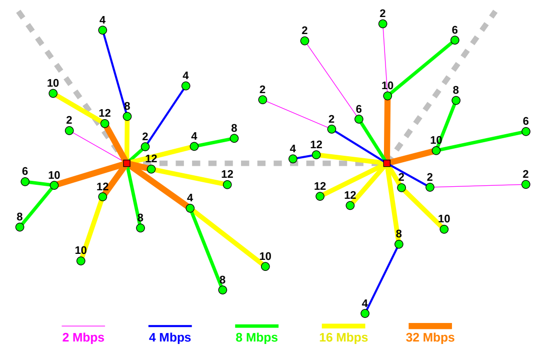


Fig. 3. Initial test topology before any restorability enhancement.

that are already protected. The remaining 40 working spans are initially unprotected. It is, also, assumed that there are 82 locations where potential new spans may be added (not shown), and all working spans are upgradable so long as they have not reached their maximum possible capacity.

The following tests focus on restoring the working traffic between the first-level aggregation nodes and their BTSs when any single working span fails. The objective is to reach 100% restorability while minimizing the cost. For simplicity, no direct inter-BTS traffic is assumed. Note that there is only one possible working path between a first-level aggregation node and each BTS within the star for the initial topology shown, and we assume that it is fixed throughout the tests. The initial capacity assigned to a working span is the minimum capacity option that can support its working traffic.

Although the algorithm allows every span to have a different set of capacity choices, we assume here for simplicity that the same set of capacity values, namely $\{2, 4, 8, 16, 32\}$ Mbps, applies to all spans. Our cost model uses the length of the span, L, in addition to a *base cost*, to determine the final span cost. Given a base cost of $C_{\text{base},x}$ for a span with capacity x Mbps, the cost of the span is assumed to be:

$$Cost(L, x) = C_{\text{base},x} \times \left(1 + \frac{L}{L_{\text{ref}}}\right), \tag{3}$$

where L_{ref} is a reference distance. The value of L_{ref} is used to adjust the sensitivity of the cost to changes in the span's length. When L_{ref} is small, an increase in length has a large effect on the span's cost. This could be the case for leased lines, where the cost is normally associated with its length. When L_{ref} is set to a large number, the cost of the span is less dependent on its length. This resembles the use of microwave links, where the hop distance has very little effect on the span's cost. The base cost of the smallest span capacity (i.e., 2 Mbps) is assumed to be 1 unit for both linear and nonlinear cost models. For span capacity $x > 2$ Mbps, the base cost is defined as follows:

$$C_{\text{base},x} = \begin{cases} x/2, & \text{for linear cost model} \quad (4) \\ 1.5^{\log_2 x - 1}, & \text{for nonlinear cost model} \quad (5) \end{cases}$$

Note that the nonlinear cost model is "4×3×", meaning that as the capacity quadruples, the cost triples. The nonlinear cost model favors the use of existing spans (those that already have capacity assignments), as opposed to the addition of new spans. For example, the base cost of upgrading a span from 2 Mbps to 4 Mbps is 0.5 units, whereas the base cost of installing a new 2 Mbps span (previously non-existent) is

TABLE I

SUMMARY OF TESTS PERFORMED.

Test	Span/path restoration	Property	Cost model
NG-L	Span	No granularity constraint	Linear
NG-NL	Span	No granularity constraint	Nonlinear
WG	Span	Granularity of a working path	Nonlinear
SG	Span	Granularity of an entire span	Nonlinear
FI	Path	Failure-independent restoration	Nonlinear

TABLE II

SUMMARY OF RESULTS OBTAINED.

Test	Extra cost (%)	No. of extra spans	Redundancy (%)
NG-L	41.91	48	87.06
NG-NL	62.26	33	96.52
WG	88.65	34	146.27
SG	97.06	35	177.61
FI	80.08	30	145.27

1 unit. Therefore, it is cheaper to use a restoration path that requires a span to be upgraded from 2 Mbps to 4 Mbps, rather than one that requires the installation of a new 2 Mbps span.

An important point to note is that the above cost model is solely used for testing only. The actual capacity choices and their costs at each span in a real RAN must be obtained individually after considering all practical constraints.

B. Tests Performed

Table I summarizes the different tests that were performed. The restorability target for each test was set to 1.0. In addition to using our heuristic algorithm for these tests, we have also formulated the test SG as a pure IP problem with binary variables. The SG is selected among all the tests because its problem size is the smallest compared to the rest. The formulation was encoded in MPS format, and passed to two general purpose solvers that utilize branch-and-bound techniques, namely, IBM's OSLMSLV, and Mosek. They were allowed to run on a 1.8 GHz machine for more than 48 hours each, and the best binary solutions discovered up till the time of manual termination were recorded.

C. Test Results

Table II summarizes the results obtained. From Tests NG-L and NG-NL, which differ only in their cost model, it is observed that linear cost model results in significantly larger number of extra spans being added. A closer look at the network designs (not shown here) reveals that 50% of the extra spans in Test NG-L have very small capacity (2 Mbps), as opposed to only 24% in Test NG-NL. These observations arise from the economy-of-scale effects introduced by the nonlinear cost model, which prefers to consolidate restoration paths to use higher capacity spans. Note that it is meaningless to compare the extra costs from the two tests, because they are obtained using different cost models.

The results from Tests NG-NL, WG, and SG are compared next. When the granularity of the restoration paths become coarser, the resulting extra cost, number of extra spans, and redundancy, all increase. This is because each restoration path will have to carry more traffic as the granularity becomes coarser, thus requiring more high capacity spans.

Finally, Test FI shows the result for failure-independent path restoration. Since each affected working path is assumed to be rerouted only via a single protection path, its granularity is equivalent to that of Test WG. By comparing their results, we see that path restoration results in lower cost as well as smaller number of extra spans being added.

For Test SG that was also formulated as a binary IP problem, both OSLMSLV and Mosek cannot obtain a global optimum solution at the end of 48 hours, due to the large problem size. However, each solver returned the best feasible solution found during the allocated time. The OSLMSLV returned an extra cost of 115.89%, while Mosek returned an extra cost of 102.84%. In contrast, the proposed heuristic algorithm obtained a lower extra cost of 97.06% in merely 10 seconds.

V. CONCLUSION

Traditional RAN topologies are mainly based on tree, chain and star structures, which have no inherent restorability properties. This work introduces an approach to upgrade existing RAN topologies cost-effectively so as to improve their restorability against single span failure scenarios. The proposed approach incrementally adds redundant spans, and upgrades existing spans, in order to create partially meshed and restorable architectures. It has numerous enhancements over existing heuristic approaches. A number of different practical issues are considered, such as potential spans and candidate transmission capacities, the use of a general cost model, and the ability to assume different variants of restoration mechanisms. From the tests on an arbitrary RAN topology, observations are summarized as follows. The presence of economy-of-scale effects in capacity costs results in a smaller number of extra spans. Cost and redundancy requirements increase when granularity of restoration paths becomes coarser. Also, path restoration results in lower cost and smaller number of extra spans being used. Finally, by formulating one of the tests as a binary IP problem, it is shown that the proposed heuristic approach can obtain a better solution than those discovered by commercial solvers after considerably long computation time.

REFERENCES

[1] D. Tipper, S. Ramaswamy and T. Dahlberg, "PCS network survivability," in Proc. *IEEE WCNC*, Sep. 1999.

[2] U. Varshney and A. D. Malloy, "An integrated approach for improving the dependability of the emerging wireless networks," in Proc. *IEEE GLOBECOM*, vol. 6, 2001, pp. 3693–3697.

[3] T. A. Dahlberg and J. Jung, "Survivable load sharing protocols: a simulation study," *ACM/Baltzer Science Wireless Networks Journal (WINET)*, vol. 7, issue 3, May 2001, pp. 283-296.

[4] T. Dahlberg, S. Ramaswamy, D. Tipper, "Issues in the survivability of wireless networks," in Proc. *IEEE MWCN*, May 1997.

[5] Y.-H. Wang, W.-S. Soh, M.-Y. Tsai and H. S. Kim, "Survivable wireless ATM network architecture," in Proc. *IEEE IC3N*, Oct. 2000, pp. 368–373.

[6] C. Harler, "The reliability of wireless networks – how much safety can you afford?" *America's Network*, Jan. 1998.

[7] State of New Jersey, "Report on the New Jersey wireless enhanced 911 system trial: the first 100 days," Jun. 1997.

[8] A. Bolle, "Capacity without ties – microwave transmission for cellular networks," *EE Times UK*, Nov. 13, 2002; http://www.electronicengineering.com/ features/OEG20021106S0001

[9] A. Szlovencsák, I. Gódor, J. Harmatos, and T. Cinkler, "Planning reliable UMTS terrestrial access networks," *IEEE Commun. Mag.*, Jan. 2002, pp. 66–72.

[10] W. D. Grover, T. D. Bilodeau, and B. D. Venables, "Near optimal spare capacity planning in a mesh restorable network," in Proc. *IEEE GLOBECOM*, 1991, pp. 2007–2012.

[11] Y. Xiong, L. Mason, "On state-independent and state-dependent path restoration in self-healing networks," in Proc. *IEEE ICC'98*, Jun. 1998, pp. 1114–1118.

[12] ECC of CEPT, "Guidance material for assessing the spectrum requirements of the fixed service to provide infrastructure to support the UMTS/IMT-2000 networks," ECC report 19, Oct. 2002.