Incremental Capacity Planning in Optical Transport Networks Based on Periodic Performance Metrics

Fernando Morales*, Marc Ruiz, and Luis Velasco

Optical Communications Group (GCO), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

**e-mail: fmorales@ac.upc.edu*

ABSTRACT

Incremental planning is performed periodically to decide how a backbone optical network has to be updated to serve the forecast traffic during the next planning period. Based on reliable traffic prediction, new equipment is installed and its capacity is ready to be used. Nonetheless, due among others to the introduction of new services, exact prediction is not usually available, which leads to install more capacity than that required thus, increasing network expenditures. To reduce expenses, we propose to monitor some performance metrics and launch the incremental capacity planning problem (INCA) to meet the target performance when some performance metric drops under a threshold. A heuristic algorithm is proposed to solve INCA in practical times. We show that INCA produces savings in terms of used cards with respect to periodical planning thus, demonstrating the utility of our proposal.

Keywords: incremental capacity planning, on-demand network planning, optical networks.

1. INTRODUCTION

Dynamicity of the traffic supported by optical transport networks is rather limited as a result of large traffic aggregation. In consequence, they are statically configured and managed, which entails capacity over-provisioning. Similarly, long planning cycles are used to upgrade and prepare optical transport networks for the next planning period; to ensure that the forecast traffic and failure scenarios can be supported, spare capacity is usually installed thus, increasing network capital expenditures (CAPEX).

In this context, some authors have proposed interesting works to plan the backbone network for multiple periods [1]-[7]. The authors in [1] assume knowledge of the traffic for every period, so as to provide a globally optimal solution to the planning problem. That knowledge is, however, not easy to be gained and it can lead to large inefficiencies when real and forecast traffic diverge. In view of that, incremental planning was proposed. Authors in [2] tackled the problem of reducing power consumption targeting the power-aware logical topology design. Authors in [3]-[4] tackled the problem of designing translucent optical networks. Authors in [5] proposed techniques for multi-period planning based on routing optimisation. Authors in [6] studied a migration scenario aiming at increasing the capacity of optical networks. Finally, authors in [7] presented a planning tool to assist in the network design for the next planning period considering new services, population, and inventory.

Notwithstanding, planning a network requires reliable traffic forecasting so the resulting network design can cope with future traffic. Nonetheless, making prediction for such traffic increments is not an easy task especially if, in addition, it comes with spatial changes in the demand (e.g. during business and night hours). In fact, in most cases there are just forecasts for traffic volume growth over time, which results in installing spare capacity that is not even used during the planning cycle. Aiming at reducing network expenditures, in this paper we propose a pay-as-you-grow approach, where new capacity is installed in accordance with traffic growth.

2. INCREMENTAL CAPACITY PLANNING

Figure 1 presents the proposed augmented network life-cycle, where the network performance is monitored and incremental capacity planning is triggered when a threshold has been exceeded. The INcremental CApacity (INCA) planning problem consists in deciding which resources need to be added to the network to ensure some performance metrics. Two network performance metrics are usually of the interest for network operators: the grade of service (i.e., blocking probability) and restorability (the ratio between the number of LSPs that are successfully restored and the total number of LSPs to restore [8]) under single link failure scenarios.

For illustrative purposes, imagine that the network in Fig. 2a is currently in operation and a blocking probability threshold has been exceeded. In such case, in-operation network planning can be triggered to re-optimize network resource utilization (see in-operation network planning cycle in Fig. 1). In the case that no feasible solution is found for the in-operation planning problem, the only way to reduce blocking probability is by adding new resources to the network. To ensure performance metrics, an incremental capacity planning algorithm might decide adding a new link connecting nodes X4 and X6, as shown in Fig. 2b. To add the new fiber link, two spare line-cards must be installed in the end OXCs and connected to available optical fibers connecting end buildings.

978-1-5090-1466-8/16/$31.00 ©2016 IEEE

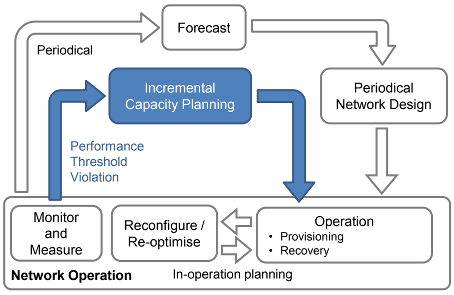

Figure 1. Augmented networks life-cycle.

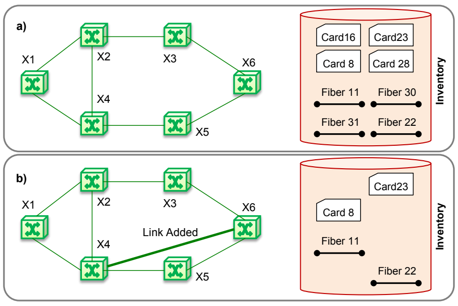

Figure 2. Example of incremental capacity planning.

To add a new link, the INCA algorithm needs to know the current state of the network including state of the resources and established LSPs. Furthermore, it needs information about physical resources such as optical fibers, line-cards and card slots, even those not yet installed. This information is usually stored in an inventory database together with geographical information about its location. In the example in Fig. 2a, the inventory reflects that four spare line-cards and four fibers are available. Consequently, planning algorithms must access such inventory database to decide which line-card must be selected, based e.g., in its geographical location, and which card slot a card must be plugged into. In Fig. 2b, a planning algorithm selected line-cards 16 and 28 and fibers 30 and 31 to create fiber link X4-X6.

3. SOLVING THE INCA PLANNING PROBLEM

The INCA problem can be formally stated as follows:

Given:

- The augmented network topology represented by graph $G_x(N, E_x)$, where N represents the set of optical nodes and E_x the set of links. The subset $L \subseteq E_x$ contains inactive links ready to be installed.
- The physical layout of each node, in terms of card slots.
- A set of available line-cards and the card-slot compatibility.
- The cost structure of adding new links.
- The blocking probability and restorability thresholds to be ensured.

Objective: Minimize the cost of extending the network topology, considering the cost of activating new links and installing new line-cards.

Output: The subset of links in L to be activated and line-cards to be installed in every node.

A general algorithm to prepare data to solve the INCA problem is presented in Table 1. It receives the current graph $G(N, E)$ representing the optical network, where E the set of fiber links connecting two optical nodes in N. The set of demands D currently being served, a connection to the inventory database and some other parameters are also received as input data.

Operation and inventory databases are first correlated (lines 1-2 in Table 1). Next, free line-cards and fiber links are retrieved from the inventory database (lines 3-4) and an augmented graph $G_x(N,E_x)$ is created by adding new links (set L) to the current graph (line 5). The INCA problem is solved using the augmented graph (line 6). Once a solution is found, the list of actions to be performed, e.g., installing line-cards, connecting fiber links to interfaces, activating fiber links, etc., is computed and returned (line 8). Note that those actions require manual intervention, so an operator should redirect them to the most appropriate engineering teams. Once the new resources are activated and become available to control and management planes, the new capacity can be used to serve traffic and for recovery purposes.

To specify the performance target to solve the INCA problem, we denote as provisioning-wise INCA (Pw-INCA) the scheme that focuses on blocking probability and recovery-wise INCA (Rw-INCA) the scheme that centers on restorability. A parameter ρ in the formulation allows choosing either approach.

Rw-INCA aims at guaranteeing that the restorability after the single failure of every active link is higher than the given threshold. In this approach, D specifies the set of currently established connections. We define a set of scenarios Q, where each scenario represents a failure in an active link (i.e., $|Q|=|E|$). The set $D(q)$ specifies the demands affected by a failure in the link of scenario q. Restoration is applied to each set of demands, where available resources include all the resources used by every demand in the set before the failure, except the failed link.

Pw-INCA targets at ensuring that the probability of accepting new incoming requests is higher than the threshold. Let us assume that we know the conditions under which the network is operating. This translates into a traffic matrix D, where every source-destination pair d specifies two components: *i*) volume b_d and *ii*) occurrence probability p_d. The way to compute the expected blocking probability is by routing each demand individually and accumulating occurrence probabilities. When solving Pw-INCA, a scenario is created for each individual demand in the traffic matrix (i.e., $|Q| = |D|$).

Table 1. Incremental capacity planning algorithm.

INPUT G(*N,E*), *D*, *Inventory*, *params*	
OUTPUT *ActionList*	
1:	*Eq* ← getEquipment (*Inventory*)
2:	Correlate (*N*, *Eq*)
3:	*LC* ← getFreeLineCards (*Inventory*)
4:	*F* ← getFreeFiberLinks (*Inventory*)
5:	$G_x(N, E_x = E \cup L)$ ←augmentGraph (*G*, *Eq*, *LC*, *F*)
6:	*S* ← solveINCA (G_x, *D*, *params*)
7:	**if** *S*=∅ **then return** *INFEASIBLE*
8:	**return** computeActionList (*S*)

Table 2. GRASP algorithm.

INPUT $G_x(N, E_x)$, *D*, *Q*, *T*, *R*, *α*, *ρ*	
OUTPUT *BestSol*, *BestFitness*	
1:	*BestSol*←Ø; *BestFitness*←∞
2:	**for each** $q \in Q$ **do** $c(q)$ ← computeGreedyCost(q)
3:	**for each** $e \in E_x$ **do** setMetric(e)
4:	**for** 1..*maxIter* **do**
5:	<*Sol*, *pfm*> ← doConstruct(G_x, *D*, *Q*, *T*, *R*, *α*, *ρ*)
6:	**if** *pfm* ≥ *thr* **then**
7:	<*Sol*, *fitness*> ← *doLocalSearch*(*Sol*)
8:	**if** *fitness* < *BestFitness* **then**
9:	*BestSol* ← *Sol*; *BestFitness* ← *fitness*
10:	**return** <*BestSol*, *BestFitness*>

The GRASP meta-heuristic has been successfully applied to solve networking problems (e.g., [9]). Table 2 presents the pseudo-code of the GRASP-based algorithm to solve the INCA problem. It receives the extended graph G_x, the set of demands *D* and scenarios *Q* as defined before, the inventory represented by sets *T* and *R*, the *α* parameter used in the constructive phase, and the INCA selector *ρ*. The algorithm assigns a greedy cost to every scenario proportional to the used capacity of the scenario link (line 2 in Table 2); costs will be used during the constructive phase. Next, the metric of every link is set to one if the link is already installed or a high value (e.g., the cardinality of set *E*) if the link is not yet installed in the network (line 3).

A number of iterations are performed, each generating a solution (lines 4-9). At each iteration, a copy of the extended graph is created and used to construct a solution (line 5). If the solution is feasible, a multi-start local search procedure is used to find a local optimal solution (line 7), which is stored provided that improves the best solution found so far (lines 8-9). The best solution and its fitness value are eventually returned (line 10). Once the GRASP heuristic returns a solution, the assignment problem needs to be solved to decide line-cards to nodes matching minimizing transportation costs. This problem can be solved by applying the *Hungarian method* [10].

4. SIMULATION RESULTS

Let us evaluate the performance of Pw-INCA and Rw-INCA algorithms through simulations. An ad-hoc event-driven simulator was developed in C++. Monitoring runs at fixed intervals (monthly) and evaluates the performance metrics. For Pw-INCA, the maximum blocking probability was set to 0.5% (i.e. thr = 99.5%), whilst for Rw-INCA the restorability threshold was set to 95%. Their performance is compared to solving the INCA problem using perfect forecast traffic for the next year.

Figures 3a, 3b present the obtained blocking probability along the considered period. Figure 3a plots results for the Pw-INCA algorithm run monthly after monitoring traffic, and at the beginning of each year based on perfect traffic estimations. As shown, the overall blocking is too high for the Pw-INCA algorithm, where blocking peaks of 1.5% and 0.5% can be observed in monthly and yearly runs, respectively; these peaks reflect the difficulty of translating current traffic distribution into a set *D*. Note that blocking probability increases rapidly during the last year as a result of inventory exhaustion, so we will ignore that year for the rest of the analysis. In contrast, the Rw-INCA algorithm (Fig. 3b) shows a constant close-to-zero blocking probability, both for the monitored-based monthly and yearly runs, where the highest blocking peak is under 0.15%. Blocking probability when no incremental capacity planning is done is also plotted in Fig. 3 for reference, despite its unacceptable values after the first year.

Plots in Figs. 4a, 4b present the results for restorability. Pw-INCA algorithm in Fig. 4a shows poor restorability even for the yearly based option since the capacity is added without considering restorability. Conversely, the results for the Rw-INCA algorithm in Fig. 4b show that restorability is virtually 100% for the yearly option and no worse than 98% for the monthly one.

Let us now compare the capacity added by INCA options; Fig. 5 plots the accumulated added capacity. It is clear that if traffic estimations are perfect, as it is assumed here, the yearly based INCA planning algorithms will install exactly the same capacity at the beginning of each year that their monthly counterparts. In addition, plots in Fig. 4c reveal that the Rw-INCA scheme installs more capacity than the Pw-INCA one; that is another reason behind their better performance. It is interesting to quantify the amount of extra capacity that is installed by the yearly options. Aiming at a fairly comparison, we define cards-year as the unit to measure the number of line-cards installed in the network during a time period. Note that increasing the amount of cards-year entails more CAPEX for the network operator. In addition, assuming that monthly and yearly options provide a comparable performance, the difference in cards-year represents the added capacity that it is not strictly needed. Table 3 presents such difference under the Pw-INCA and the Rw-INCA schemes. The amount of not needed capacity is remarkably high, which gives value to the on-line option. Note that the installed capacity in the yearly option can be even higher as a result of imperfect traffic estimations.

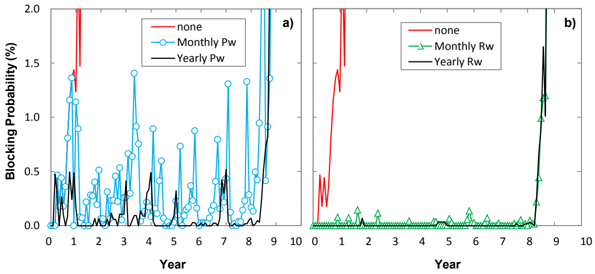

Figure 3. Blocking prob: Pw-INCA (a) and Rw-INCA (b).

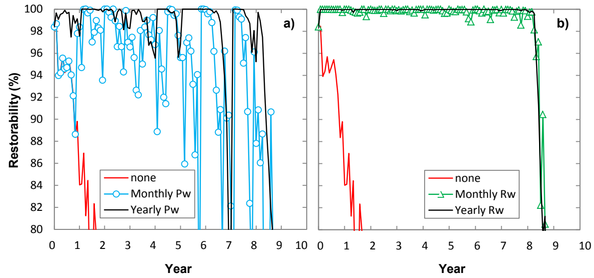

Figure 4. Restorability vs. time: Pw-INCA (a) and Rw-INCA (b).

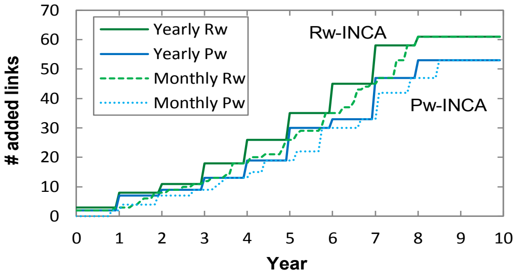

Figure 5. Added links vs. year.

Table 3. Cards-year: yearly-monthly difference.

	Year #									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Pw-INCA	3.3	5.8	3.0	3.0	3.7	13	4	9.8	6	0
Rw-INCA	1.8	6.2	2.8	6.7	10	11	11	14	0	0

5. CONCLUSIONS

Planning the network on-demand was proposed in this paper by solving the incremental capacity planning (INCA) problem. Results showed that the Pw-INCA provides peaks of high blocking and low restorability, whereas the Rw-INCA showed a close-to-zero blocking probability and virtually full restorability, installing more capacity instead. Up to 14 cards-year of capacity was required by periodical planning with respect to on-line INCA, thus demonstrating the potential savings of our proposal.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the Spanish MINECO SYNERGY project (TEC2014-59995-R) and from the Catalan Institution for Research and Advanced Studies (ICREA).

REFERENCES

[1] S. Strauss, A. Kirstadter, and D. Schupke, "Multi-period planning of WDM-networks: comparison of incremental and EoL approaches," in *Proc. International Conference in Central Asia on Internet (ICI)*, 2006.
[2] E. Bonetto, L. Chiaraviglio, F. Idzikowski, and E. Le Rouzic, "Algorithms for the multi-period power-aware logical topology design with reconfiguration costs," *IEEE/OSA Journal of Communications and Networking*, vol. 5, pp. 394-410, 2013.
[3] N. Shinomiya, T. Hoshida, Y. Akiyama, H. Nakashima, and T. Terahara, "Hybrid link/path-based design for translucent photonic network dimensioning," *J. Lightw. Technol.*, vol. 25, pp. 2931–2941, Oct. 2007.
[4] Z. Zhu, X. Chen, F. Ji, L. Zhang, F. Farahmand, and J. Jue, "Energy-efficient translucent optical transport networks with mixed regenerator placement," *J. Lightw. Technol.*, vol. 30, pp. 3147-3156, Oct. 2012.
[5] N. Geary, A. Antonopoulos, E. Drakopoulos, and J. O'Reilly, "Analysis of optimisation issues in multi-period DWDM network planning," in *Proc. IEEE INFOCOM*, vol. 1, pp. 152-158, 2001.
[6] C. Meusburger and D. Schupke, "Optimizing the migration of channels with higher bitrates," *J. Lightw. Technol.*, vol. 28, pp. 608-615, 2010.
[7] M. Ruiz, L. Velasco, A. Lord, D. Fonseca, M. Pióro, R. Wessäly, and J.P. Fernández-Palacios, "Planning fixed to flexgrid gradual migration: drivers and open issues," *IEEE Communications Magazine*, vol. 52, pp. 70-76, 2014.
[8] M. Clouqueur and W. Grover, "Availability analysis of span-restorable mesh networks," *IEEE J. Sel. Areas Commun.*, vol. 20, pp. 810-821, 2002.
[9] A. Castro, L. Velasco, M. Ruiz, M. Klinkowski, J. P. Fernández-Palacios, and D. Careglio, "Dynamic routing and spectrum (re)allocation in future flexgrid optical networks," *Elsevier Computers Networks*, vol. 56, pp. 2869-2883, 2012.
[10] A. Schrijver, "*Combinatorial Optimization: Polyhedra and Efficiency*," Springer, 2003.