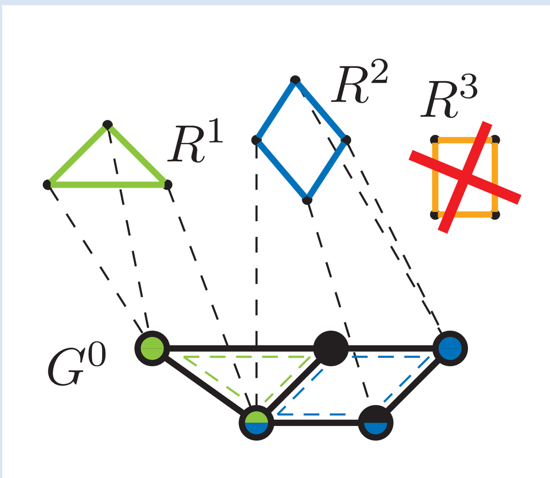
Virtual Network Embedding Under Uncertainty: Exact and Heuristic Approaches

Stefano Coniglio, Arie Koster, Martin Tieves
$\{coniglio, koster, tieves\}$@math2.rwth-aachen.de

Lehrstuhl II für Mathematik
RWTH AACHEN UNIVERSITY

Introduction

Given a substrate network $G^0 = (V^0, A^0)$ and a set R of virtual network requests (VNs $r \in R$), the **Virtual Network Embedding** problem (VNE) calls for an embedding of a selection of virtual networks onto the substrate, maximizing the profit (p_r) of the thus accepted VNs. An embedding is given by a *virtual-to-physical* mapping of nodes and links, subject to *capacity constraints* (c^0_i, k^0_{ij}).



Since, in practical scenarios, node (w^r_v) and link demands (d^r_{vw}) are typically *not known exactly*, we propose and investigate robust optimization approaches for VNE.

Formulating robust VNE

It is very unlikely for all requested resources to *simultaneously be at their peak*. Assuming that only a fixed number of these may reach such value, a trade-of between QoS and profit can be achieved by aiming at solutions which guarantee feasibility in *almost all* the cases.

Denoting with y^r whether a VN r is accepted and with x^r_{iv} whether r's virtual node v is mapped on $i \in V^0$, this corresponds to solving the **chance-constrained** problem:

$$\max \sum_{r \in R} p^r y^r \qquad (1)$$

$$\text{s.t.} \sum_{i \in V^0(r,v)} x^r_{vi} = y^r \qquad (2)$$

$$\Pr\Big(\sum_{r \in R} \sum_{\substack{v \in V^R:\\ i \in V^0(r,v)}} w^r_v x^r_{vi} \le c^0_i\Big) \ge \epsilon \qquad (3)$$

$$\Pr\Big(\sum_{r \in R} \sum_{v,w \in V^r} d^r_{vw} f^{vw,r}_{ij} \le k^0_{ij}\Big) \ge \epsilon \qquad (4)$$

$$\sum_{(i,j) \in \delta^+(i)} f^{vw,r}_{ij} - \sum_{(j,i) \in \delta^-(i)} f^{vw,r}_{ji} = x^r_{vi} - x^r_{wi} \qquad (5)$$

$$y^r, x^r_{vi}, f^{vw,r}_{ij} \in \{0,1\}. \qquad (6)$$

A computationally tractable way to approximate it is the so-called Γ-**robustness** model. Hereby, each constraint is protected against at most Γ many parameter deviations, i.e.,

$$\sum_i a_i x_i \le b \longrightarrow \sum_i \bar{a}_i x_i + \max_{|T| \le \Gamma} \sum_{i \in T} \hat{a}_i x_i \le b.$$

These protected versions of (3), (4) can be reformulated compactly. We call the resulting MILP the Γ-**robust VNE problem**.

Contribution

We propose an exact and a heuristic approach to solve the robust VNE problem. The Γ-**robust Mixed-Integer Linear Programming formulation** allows us to find solutions offering *large profits* and are guaranteed to be *feasible with a high probability*. We also introduce a MILP-based Γ-**robust heuristic** carrying out the node and link mappings sequentially.

Computational experiments indicate that:

1. The Γ robust formulation is suitable to solve small size instances,
2. the heuristic scales well, providing high quality solutions even for larger problems.

An exact Γ-robust approach

To better motivate the *potential of robust optimization* for VNE, we first evaluate VNE with worst case data (i.e., $\Gamma = \infty$ and $\epsilon = 1$) and with average data (i.e., $\Gamma = 0$ and $\epsilon = 0$), both with a time limit of one hour. Further, we present solution values, obtained by solving the Γ-**robust VNE problem** with the same time limit. As we can see, the Γ-robust solutions yield a substantial improvement (concerning the profit) over the solutions corresponding to worst-case data.

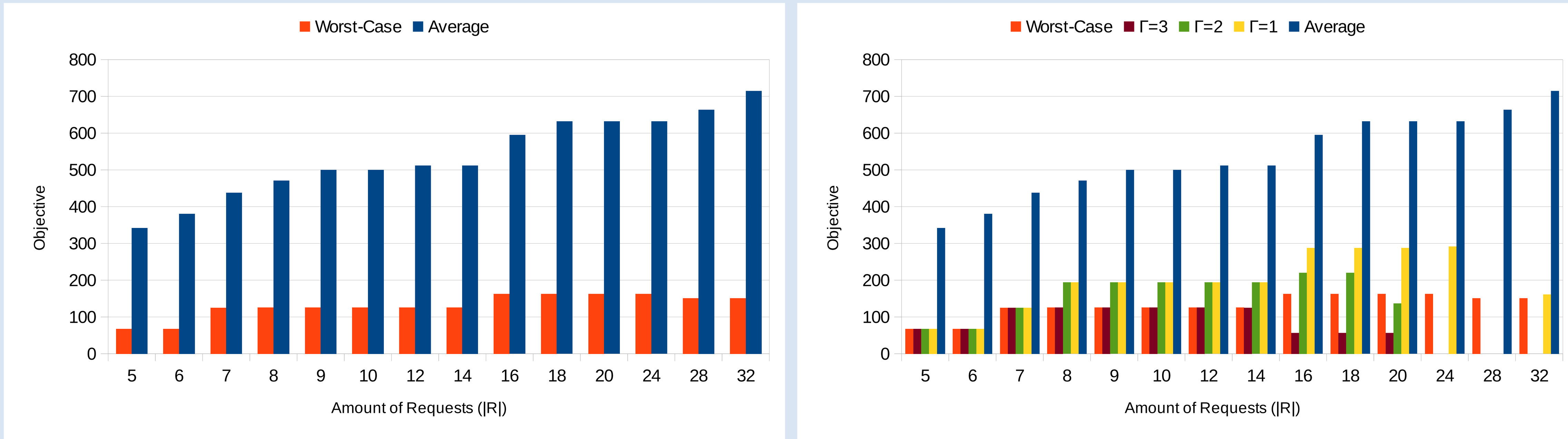


Figure 1: Worst/average case obj. function value (left), obj. function values for different Γ values (right).

Note that the value of an optimal Γ-robust solution should be between the average and worst case ones and that such value should be larger for smaller values of Γ. This is not always the case, indicating that the exact approach does not scale well for bigger sized instances.

A heuristic Γ-robust approach

Split the VNE problem in **two subproblems**, solved sequentially (within a time limit):

- **Phase I**: *admission control* (2) + *node embedding* (3), neglect link mapping (4)/capacities (5).
- **Phase II**: complete the solution by a *link mapping* (4), (5), still *allowing for rejection* of VNs (2).

Since Phase I is completely oblivious to the routing aspect, depending on a parameter z, on the distance $\sigma(i,j)$ between two nodes $i,j \in V^0$, and on a demand threshold T, we add constraints of the form

$$x^r_{vi} + x^r_{wi} \le 1 \qquad \begin{array}{l} \forall\, r \in R, vw \in V^r, \\ \forall\, i,j \in V^0 : \sigma(i,j) > z, d^r_{vw} > T \end{array}$$

to the first phase to better account for the routing aspect. The heuristic achieves the following results:

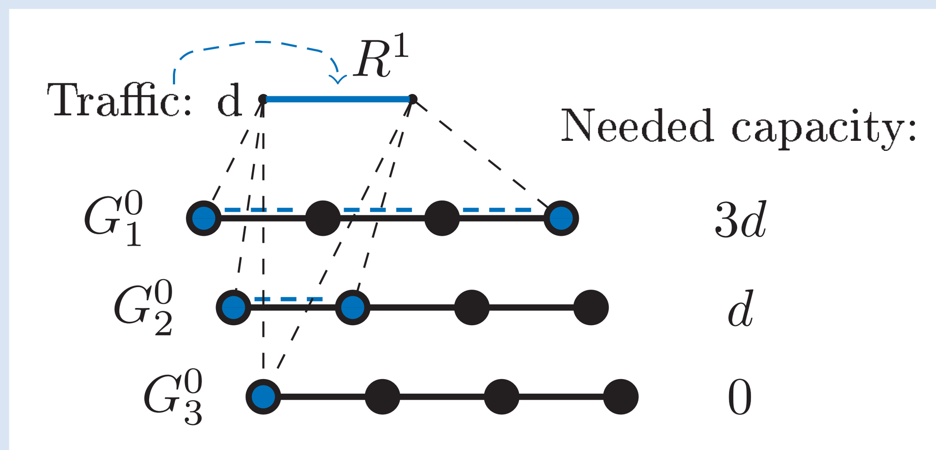


Figure 2: Distance vs. capacity req.

	Substrate $\lvert R\rvert$	5	6	7	8	9	10	12	14	16	18	20	24	28	32	Avg.
EXACT $\Gamma = 1$	ABILENE	68	68	125	194	194	194	194	194	288	288	288	292	-	161	**196**
	ATLANTA	332	374	453	489	553	-	-	-	-	-	-	-	-	-	**440**
	NOBEL-US	294	341	389	480	518	518	-	-	-	-	-	-	-	-	**423**
	POLSKA	265	312	398	452	550	550	-	-	-	-	-	-	-	-	**421**
	Avg.	**240**	**274**	**341**	**404**	**454**	**421**	**194**	**194**	**288**	**288**	**288**	**292**	**-**	**161**	**295**
HEURISTIC $\Gamma = 1$	ABILENE	68	68	125	194	194	194	194	194	288	288	288	324	381	381	**227**
	ATLANTA	332	374	453	489	553	589	783	723	823	860	815	872	825	124	**615**
	NOBEL-US	294	341	389	480	518	518	555	631	636	608	617	724	725	696	**552**
	POLSKA	265	312	398	422	520	520	562	606	597	580	580	548	584	710	**515**
	Avg.	**240**	**274**	**341**	**396**	**446**	**455**	**524**	**539**	**586**	**584**	**575**	**617**	**629**	**478**	**477**
EXACT $\Gamma = 2$	ABILENE	68	68	125	194	194	194	194	194	220	220	137	-	-	-	**164**
	ATLANTA	332	374	-	-	-	-	-	-	-	-	-	-	-	-	**353**
	NOBEL-US	294	294	294	98	-	-	-	-	-	-	-	-	-	-	**245**
	POLSKA	265	312	321	-	104	76	-	-	-	-	-	-	-	-	**215**
	Avg.	**240**	**262**	**247**	**146**	**149**	**135**	**194**	**194**	**220**	**220**	**137**	**-**	**-**	**-**	**190**
HEURISTIC $\Gamma = 2$	ABILENE	68	68	125	194	194	194	194	194	231	231	231	256	277	277	**195**
	ATLANTA	332	374	453	489	511	519	605	722	609	673	673	713	600	345	**544**
	NOBEL-US	294	294	342	366	402	402	366	402	405	401	402	469	335	496	**384**
	POLSKA	265	312	360	394	454	454	478	491	481	526	526	472	537	583	**452**
	Avg.	**240**	**262**	**320**	**361**	**390**	**392**	**411**	**452**	**432**	**458**	**458**	**478**	**437**	**425**	**394**

The cases were a better objective value is achieved by the heuristic are marked in green while cases in which the exact method produces a superior result are marked in red. Notably, for many of the instances where the exact approach does not find a solution, the heuristic can provide one. We also remark that for the heuristic, in both phases, a significant shorter runtime limit of 300s was employed.

A note on protection

We measure the **protection level** of a solution by testing it against 100 different realizations of uncertain data, counting how often a solution is infeasible. The solutions for average case (worst-case) data are always (never) violated. By increasing Γ, the protection level of both the exact approach (left) and the heuristic one (right) increase. Note that both approaches yield similar protection levels with respect to the parameter Γ, with slight advantages for the exact method (left).

	Substrate $\lvert R\rvert$	5	6	7	8	9	10	12	14	16	18	20	24	28	32	Avg.
$\Gamma = 1$	ABILENE	4	1	6	17	11	12	17	3	19	40	20	15	-	12	**14**
	ATLANTA	2	7	14	37	18	-	-	-	-	-	-	-	-	-	**16**
	NOBEL-US	8	14	6	28	36	22	-	-	-	-	-	-	-	-	**19**
	POLSKA	5	15	34	24	27	58	-	-	-	-	-	-	-	-	**27**
	Avg.	**5**	**9**	**15**	**27**	**23**	**31**	**17**	**3**	**19**	**40**	**20**	**15**	**-**	**12**	**18**
$\Gamma = 2$	ABILENE	0	0	1	0	2	1	0	1	0	0	0	-	-	-	**0**
	ATLANTA	0	0	-	-	-	-	-	-	-	-	-	-	-	-	**0**
	NOBEL-US	0	1	0	0	-	-	-	-	-	-	-	-	-	-	**0**
	POLSKA	0	4	0	0	0	0	-	-	-	-	-	-	-	-	**1**
	Avg.	**0**	**1**	**0**	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**0**	**-**	**-**	**-**	**0**
$\Gamma = 3$	ABILENE	0	0	0	0	0	0	0	0	0	0	0	-	-	-	**0**
	ATLANTA	0	-	-	-	-	-	-	-	-	-	-	-	-	-	**0**
	NOBEL-US	0	0	0	-	-	-	-	-	-	-	-	-	-	-	**0**
	POLSKA	0	-	-	-	-	-	-	-	-	-	-	-	-	-	**0**
	Avg.	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**-**	**-**	**-**	**0**

5	6	7	8	9	10	12	14	16	18	20	24	28	32	Avg.
0	0	6	11	11	11	11	11	11	11	11	37	40	40	**15**
10	6	10	14	21	22	49	57	59	59	65	51	56	0	**34**
7	14	23	27	17	21	32	49	53	55	34	39	49	49	**34**
5	7	16	21	26	26	47	45	47	50	50	48	41	40	**34**
6	**7**	**14**	**18**	**19**	**20**	**35**	**41**	**43**	**44**	**40**	**44**	**47**	**32**	**29**
0	0	0	0	0	0	0	0	0	0	0	3	0	0	**0**
0	1	1	3	2	1	1	0	4	3	2	1	2	0	**2**
0	0	0	1	1	1	1	0	2	0	1	0	0	4	**1**
0	1	0	2	1	1	1	3	1	2	2	1	1	2	**1**
0	**1**	**0**	**2**	**1**	**1**	**1**	**1**	**2**	**1**	**1**	**1**	**1**	**2**	**1**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**