Network Slice Embedding under Traffic Uncertainties - A Light Robust Approach

Andreas Baumgartner, Thomas Bauschert, Abdul A. Blzarour, Varun S. Reddy
Technische Universität Chemnitz, Chair of Communication Networks, 09126 Chemnitz, Germany
Email: andreas.baumgartner | thomas.bauschert | abdulaziz.alghosh-blzarour | srva@etit.tu-chemnitz.de

***Abstract*—5G networks are conceived to be highly flexible and programmable end-to-end connect-and-compute infrastructures. In that context, the concept of network slicing is of particular importance [1]. From a network infrastructure point of view, network slicing refers to the provisioning and assignment of physical substrate network resources to tenants. To enable an efficient resource allocation, suitable optimization models are required. In our previous contributions [2], we presented a model that takes into account traffic uncertainty by using the well known concept of Γ-robustness. We further extended this model to cope with general network slices and to consider also single substrate link and node failures [3]. In this paper, we present a novel model applying the concept of light robustness [4] [5] to address scalability issues of our previous models and to get a deeper insight into the tradeoff between the price of robustness and the realized robustness. We compare our model with a nominal and Γ-robust approach for different scenarios using network topology examples from SNDlib [6].**

I. Introduction

Network slicing is defined as a concept for the concurrent deployment of multiple logical, self-contained and independent networks onto a common infrastructure plattform [1]. Based on the ideas of Software Defined Networking (SDN) and Network Function Virtualization (NFV), network slicing describes the vision of an "on-demand", fully-automized provisioning of end-to-end service specific network slices according to technical requirements or customer preferences/business models. The key advantage of network slicing is to provide a enhanced degree of flexibility in terms of capability, elasticity and adaptability in the operation and maintenance of future communication networks.

In the context of 5G, network slicing is seen as a key opportunity to make the network capable of handling highly diverse types of communication between humans, machines, sensors and actors with different performance and QoS attributes: Each individual 5G network slice has the capability for best delivering a particular service type [7]. In [8], the network slicing concept is conceived as a 3-layer architecture: On top of the physical resource layer (i.e. the substrate infrastructure), the network slice instance layer provides the networking capabilities which in turn are required by the specific service instances. The service instance layer describes the end-user or business services.

The planning and provisioning of network slice instances comprises the placement of virtual network functions and their interconnections, i.e. the embedding of network service chains on the physical substrate network infrastructure. In the last years, several publications addressed the problem of virtual network embedding (see e.g. [9] and the references therein). The objective is to provide an (cost-)efficient embedding solution considering robustness and reliability (wrt. traffic fluctuations and substrate network failures) as well as quality of service (QoS) constraints. In our previous contributions [10], [11], we presented mixed integer linear programming (MILP) models for the VNF/service chain embedding problem in the special case of an virtual mobile core network scenario. We further extended our model to cope with traffic uncertainties by applying the well-known concept of Γ-Robustness [12]. In [13], we outlined a first optimization model for the robust and reliable combined network slice design and embedding problem and provided a comparative analysis for different (robust and non-robust) modelling approaches. In contrast to common VNE problem statements, which map a service (i.e. a given virtual network graph) directly onto a network infrastructure, the network slice design problem includes the determination of the network slice topology. In this work, we adapt the concept of light robustness [4] [5] to the network slice design and embedding problem and investigate the tradeoff between the price of robustness and the realized robustness for different robust optimization models and compare their solution times.

The paper is structured as follows: In Chapter II, the nominal and Γ-robust problem formulation are stated and the novel light robust modelling approach is presented. In Chapter IV we investigate the performance of the light robustness model compared to the nominal and Γ-robust model by means of example network scenarios taken from SNDlib and discuss the results. Finally, Chapter V provides a short summary and an outline of our future work.

II. The Network Slice Design Problem

In this section, we present a MILP model for the general network slice design problem. We consider a given substrate network, represented as an undirected graph (N_s, L_s), with given node capacities $c_{n_s} > 0$ and link capacities $c_{l_s} > 0$. The cost for fully utilising the resources of a substrate node n_s and a substrate link l_s is denoted by p_{n_s} and p_{l_s}, respectively. For each substrate network link l_s we assume a propagation delay Δ_{ls} proportional to its length. In the first step we consider the design of a single network slice onto a given substrate network targeting to minimise the overall costs.

978-3-901882-98-2 © 2017 IFIP

For a network slice the end-to-end traffic demands $d_t \in \mathbb{R}_+$, $t \in T$ must be routed through a set of virtual network functions N_v. These virtual network functions are provisioned on the substrate network nodes. Restrictions regarding the placement of these virtual functions may apply due to geographical, economical or security considerations. We thus introduce a parameter $\delta_{n_s}^{n_v} \in \{0,1\}$ indicating the ability of the substrate node n_s to host the respective virtual function n_v. Similarly, virtual links $l_v \in L_v$ need to be established between the virtual functions.

The decision variables $x_{n_s}^{t,n_v} \in \{0,1\}$ indicate if the traffic demand d_t is processed/routed through a virtual function n_v residing on a substrate node n_s. The variables $f_{l_s}^{t,l_v} \in \{0,1\}$ indicate if the traffic demand d_t on the virtual link l_v between the virtual functions $(n_{v_1}, n_{v_2}) \in L_v$ is routed over the physical link l_s. By this definition, we imply that the traffic is routed over a single path between source and destination. The variables $y_{n_s}^{n_v} \in \mathbb{Z}_{\geq 0}$ specify the number of capacity modules of size κ allocated to a virtual function $n_v \in N_v$ residing on a substrate node $n_s \in N_s$ and the variables $u_{n_s}, u_{l_s} \in \mathbb{R}_{[0,1]}$ denote the utilisation of the physical substrate node resources and link resources, respectively.

The mathematical formulation of the general network slice design problem is as follows:

$$\textbf{min} \sum_{n_s \in N_s} p_{n_s} u_{n_s} + \sum_{l_s \in L_s} p_{l_s} u_{l_s} \tag{1a}$$

$$\text{s.t.} \sum_{n_s \in N_s} \delta_{n_s}^{n_v} x_{n_s}^{t,n_v} = 1 \quad \forall t \in T, n_v \in N_v \tag{1b}$$

$$\sum_{t \in T} d^t x_{n_s}^{t,n_v} \leq \kappa \, y_{n_s}^{n_v} \quad \forall n_v \in N_v, n_s \in N_s \tag{1c}$$

$$\sum_{n_v \in N_v} \kappa \, y_{n_s}^{n_v} \leq c_{n_s} u_{n_s} \tag{1d}$$

$$\sum_{(n_s,w) \in L_s} f_{(n_s,w)}^{t,l_v} - f_{(w,n_s)}^{t,l_v} = x_{n_s}^{t,n_{v1}} - x_{n_s}^{t,n_{v2}} \tag{1e}$$

$$\sum_{t \in T} \sum_{l_v \in L_v} d^t f_{l_s}^{t,l_v} \leq c_{l_s} u_{l_s} \quad \forall l_s \in L_s \tag{1f}$$

$$\sum_{l_v \in L_v} \sum_{l_s \in L_s} \Delta_{ls} f_{ls}^{t,lv} \leq b_t \quad \forall t \in T, l_v \in L_v, ls_s \in L_s \tag{1g}$$

$$x_{n_s}^{t,n_v}, f_{n_s}^{t,n_v} \in \{0,1\}, y_{n_s}^{n_v} \in \mathbb{Z}_+, u_{n_s}, u_{l_s} \in \mathbb{R}_{[0,1]} \tag{1h}$$

The objective function (1a) minimises the cost for setting up the network slice on the substrate network.

Constraint (1e) enforces flow conservation at each substrate node $n_s \in N_s$ while constraint (1g) assures that the traffic flows will not exceed the end-to-end propagation time (b_t).

III. Robust Network Slice Design Models

In this section we outline robust optimization models for the network slice design problem with uncertain traffic demands. First we present a model based on the well-known Γ-robust approach and afterwards we derive a model based on the light robustness approach.

A. *The Network Slice Design Problem: Γ-robust approach*

The previous mathematical formulation represents the nominal model which does not take into account any abnormal deviation of the traffic demands which would lead to resource congestion in the substrate network. In the Γ-robust approach the traffic demand uncertainty is represented by a subset of size Γ of deviated demands against which the optimization solution should be robust. Γ represents the protection level where $\Gamma = 0$ means no protection at all (i.e. nominal traffic demand) while the highest Γ value means that all traffic demands are at their peak values. Here we assume that the traffic demands can be modelled as random variables d^t which take value in the range $[\bar{d}^t - \hat{d}^t, \bar{d}^t + \hat{d}^t]$ where $\bar{d}^t$ refers to the nominal values and $\hat{d}^t$ is the maximum traffic demand deviation.

The Γ-robust counterpart of the capacity constraints (1c) reads as follows:

$$\sum_{t \in T} \bar{d}^t x_{n_s}^{t,n_v} + \max_{\substack{D \subseteq T \\ |D| \leq \Gamma}} \sum_{t \in D} \hat{d}^t x_{n_s}^{t,n_v} \leq \kappa \, y_{n_s}^{n_v}$$

The Γ-robust model can be written as follows:

$$\textbf{min} \sum_{n_s \in N_s} p_{n_s} u_{n_s} + \sum_{l_s \in L_s} p_{l_s} u_{l_s} \tag{2a}$$

$$\text{s.t.} \sum_{n_s \in N_s} \delta_{n_s}^{n_v} x_{n_s}^{t,n_v} = 1 \quad \forall t \in T, n_v \in N_v \tag{2b}$$

$$\sum_{t \in T} \bar{d}^t x_{n_s}^{t,n_v} + \sum_{t \in T} \rho_{n_s}^{t,n_v} + \Gamma \pi_{n_s}^{n_v} \leq \kappa \, y_{n_s}^{n_v} \quad \forall t \in T, n_v \in N_v, n_s \in N_s \tag{2c'}$$

$$\rho_{n_s}^{t,n_v} + \pi_{n_s}^{n_v} \geq \hat{d}^t x_{n_s}^{t,n_v} \quad \forall t \in T, n_v \in N_v, n_s \in N_s \tag{2c''}$$

$$\sum_{n_v \in N_v} \kappa \, y_{n_s}^{n_v} \leq c_{n_s} u_{n_s} \tag{2d}$$

$$\sum_{(n_s,w) \in L_s} f_{(n_s,w)}^{t,l_v} - f_{(w,n_s)}^{t,l_v} = x_{n_s}^{t,n_{v1}} - x_{n_s}^{t,n_{v2}} \tag{2e}$$

$$\sum_{t \in T} \sum_{l_v \in L_v} \bar{d}^t f_{l_s}^{t,l_v} + \sum_{t \in T} \rho_{l_s}^t + \Gamma \pi_{l_s} \leq c_{l_s} u_{l_s} \quad \forall l_s \in L_s \tag{2f'}$$

$$\rho_{l_s}^t + \pi_{l_s} \geq \sum_{l_v \in L_v} \bar{d}^t f_{l_s}^{t,l_v} \quad \forall t \in T, l_s \in L_s \tag{2f''}$$

$$\sum_{l_v \in L_v} \sum_{l_s \in L_s} \Delta_{ls} f_{ls}^{t,lv} \leq b_t \quad \forall t \in T, l_v \in L_v, l_s \in L_s \tag{2g}$$

$$x_{n_s}^{t,n_v}, f_{n_s}^{t,n_v} \in \{0,1\}, y_{n_s}^{n_v} \in \mathbb{Z}_+, u_{n_s}, u_{l_s} \in \mathbb{R}_{[0,1]} \tag{2h}$$

By the constraints (2c'), (2c''), and (2f'), (2f''), the nominal model is converted into a Γ-robust one. They are the dual version of the constraints (1c) and (1f) of the nominal model which are affected by the traffic demand uncertainty. The Γ-robust model achieves a high realized robustness but usually results in a quite conservative and thus costly solution. Hence, we propose the light robust approach as an alternative to the Γ-robust model.

B. The Network Slice Design Problem: Light robust approach

The concept of light robust is to ensure a high level of robustness, similar to the Γ-robust model, but with a capped price for robustness. When the number of demands, which allowed to deviate simultaneously is increased, the resource constraints are about to be violated. In the light robust model, slack variables $\gamma_{n_s}^{n_v}, \gamma_{l_s}$ are used as subtracted terms of the demand side to assure the balance of the involved constraint. Obviously, the held value by the slack variables expresses the lack of resources at specific network nodes and links. The objective function of the problem formulation is to minimize these capacity constraint violations.

In contrast to the Γ-robust formulation, where the price of robustness was subject to optimisation, the light robust approach considers an upper bound on the price of robustness within the formulation thereby enabling the cost-aware design of network slices. The mathematical formulation of the light robust model is as follows:

$$\textbf{min} \sum_{n_v \in N_v} \sum_{n_s \in N_s} \gamma_{n_s}^{n_v} + \sum_{l_s \in L_s} \gamma_{l_s} \tag{3a}$$

$$\text{s.t.} \sum_{n_s \in N_s} \delta_{n_s}^{n_v} x_{n_s}^{t,n_v} = 1 \qquad \forall t \in T, n_v \in N_v \tag{3b}$$

$$\sum_{t \in T} \bar{d}^t x_{n_s}^{t,n_v} \sum_{t \in T} \rho_{n_s}^{t,n_v} + \Gamma \pi_{n_s}^{n_v} - \gamma_{n_s}^{n_v} \leq \kappa\, y_{n_s}^{n_v} \qquad \forall n_v \in N_v, n_s \in N_s \tag{3c'}$$

$$\rho_{n_s}^{t,n_v} + \pi_{n_s}^{n_v} \geq \hat{d}^t x_{n_s}^{t,n_v} \qquad \forall t \in T, n_v \in N_v, n_s \in N_s \tag{3c''}$$

$$\sum_{n_v \in N_v} \kappa\, y_{n_s}^{n_v} \leq c_{n_s} u_{n_s} \tag{3d}$$

$$\sum_{(n_s,w) \in L_s} f_{(n_s,w)}^{t,l_v} - f_{(w,n_s)}^{t,l_v} = x_{n_s}^{t,n_{v1}} - x_{n_s}^{t,n_{v2}} \tag{3e}$$

$$\sum_{t \in T} \sum_{l_v \in L_v} \bar{d}^t f_{l_s}^{t,l_v} + \sum_{t \in T} \rho_{l_s}^t + \Gamma \pi_{l_s} - \gamma_{l_s} \leq c_{l_s} u_{l_s} \qquad \forall l_s \in L_s \tag{3f'}$$

$$\rho_{l_s}^t + \pi_{l_s} \geq \sum_{l_v \in L_v} \hat{d}^t f_{l_s}^{t,l_v} \qquad \forall t \in T, l_s \in L_s \tag{3f''}$$

$$\sum_{l_v \in L_v} \sum_{l_s \in L_s} \Delta_{ls} f_{ls}^{t,lv} \leq b_t \qquad \forall t \in T, l_v \in L_v, l_s \in L_s \tag{3g}$$

$$\sum_{n_s \in N_s} p_{n_s} u_{n_s} + \sum_{l_s \in L_s} p_{l_s} u_{l_s} \leq (1+\sigma) \cdot z^* \tag{3h}$$

In the light robust approach slack variables γ_{n_s} and γ_{l_s} (in constraints (3c') and (3f") are introduced to allow violations of the capacity constraints. Additionally, a new constraint (3h) is provided, which yields an upper bound for the deterioration of the objective function. In (3h), the parameter z^* refers to an optimum solution of the nominal problem (Constraints (1a) till (1h)) and σ refers to the additional price we are willing to pay for robustness against traffic uncertainties. The objective function (3a) minimizes the violation of the node capacity and link capacity constraints ((3c'),(3c") and (3f"), (3g)).

IV. Performance Evaluation

In this chapter, we evaluate the two robust optimization approaches wrt. realised robustness, price of robustness, computation time and optimality gaps.

A. Scenario Setup

The models are evaluated applying three different network topology instances taken from SNDlib [6]: polska (12 nodes, 18 links), abilene (12 nodes, 15 links) and nobel-germany (17 nodes, 26 links). For all network instances, we assume a node capacity of 500 units ($c_{n_s} = 500$), a link capacity of 400 units ($c_{l_s} = 400$) and a virtual module size (κ) of 10 units. For the perfomance evaluation, the maximum value of Γ is set to 10 while the number of traffic demands of a virtual network slice is 15. We generate traffic matrices for 15-minute intervals over a time period of 30 days and calculate the arithmetic mean of each traffic demand (as the nominal demand value) and its standard deviation (considered as its max-deviated value) throughout the 2880 intervals. This values are then used as an input for both the Γ-robust as well as the light robust models.

To show the quality of the obtained solutions, we calculate the realised robustness (rr) considering all traffic demand instances. If a violation of any constraint (physical link or node capacity) is observed, the virtual network design and embedding solution is considered to be non-robust for the traffic matrix in this time interval. The realized robustness is calculated as follows:

$$rr = \frac{2880 - \text{number of violated intervals}}{2880} * 100$$

where the number 2880 refers to the total number of time intervals (see above).

Another evaluation measure is the price of robustness (por). It represents the added expenses due to robustification as a percentage of the cost of the nominal solution (i.e. the value of the objective function (**2a**) where no protection level is considered) and is calculated as follows:

$$\text{por} = \frac{z - z^*}{z^*} * 100$$

where z^* is the value of the objective function for $\Gamma = 0$ (nominal solution) and z is the counterpart objective function value for protection level $\Gamma > 0$. Usually, the higher the level of robustness, the higher is also the price of robustness.

For the light robust model, we add the objective function of the Γ-robust model (**2a**) as a constraint (**3h**). We assure

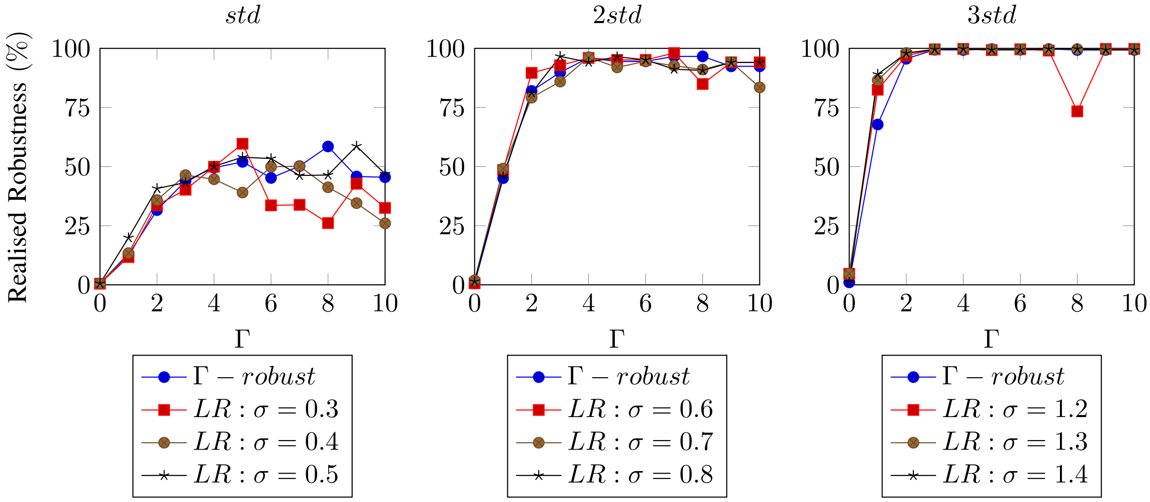

Fig. 1. Realised Robustness vs. Protection Level (Γ-value) - abilene network.

that the cost of the light robust solution is less than or equal to the sum of the cost of the nominal solution z^* (Γ = 0) and a surplus value σ (expressed as percentage of z^*). The parameter σ is a key parameter in the light robust model as it is used to set an upper bound for the additional cost for providing a certain level of robustness. In this way we get a robust solution with controllable expenses. Contrary, in the Γ-robust model we can only control the level of robustness - the cost of the solution is just a result of the optimization. To show the dependency of the Max-Dev demand values, we determine solutions for both models and all network topologies with two additional max-deviated values Max-Dev = 2·std and Max-Dev = 3·std. In case of the Γ-robust model we determine 99 solution instances (for 11 Γ values ranging from 0 to 10, 3 networks and 3 different maximum deviated values of the demands). For the light robust model, we obtain 1254 solution instances with respect to different σ-values, Γ values, max-deviated values and network topologies.

The optimization models are solved on a PC with an Intel i7-3930 CPU and 64 GB RAM, applying the AMPL modeling language [14] and IBM ILOG CPLEX Version 12.6.3.0.

B. Price of Robustness vs. Realised Robustness

For a comparison of both models, we compare the rr of the Γ-robust instance and its corresponding por. After that, this por value is used as reference to identify solutions of the light robust model which have a comparable por-value and a rr-value close to the respective Γ-robust model. Our simulations show, that the light robust model yields cheaper solutions with only a small decrease in the realised robustness compared to the Γ-robust approach: E.g. in the case of the abilene network, the Γ-robust model with $max - dev = 2 * std$ and $\Gamma = 4$, yields a realised robustness of 96% ($rr = 96\%$), and a price of robustness of 66.13% ($por = 66.13\%$). For the same network, the solution of the light robust model for $\Gamma = 7$ yields a realised robustness of 98.02% with a price of robustness of 60%.

In Figure 1, the realised robustness of both models is shown compared to their protection level (number of demands deviating simultaneously) for the case of the abilene network and for three max-dev values ($1std$, $2std$ and $3std$). The result of the Γ-robust model is compared to three instances of the light robust model (with three different σ values). The blue curves represents the eleven Γ-robust model solution instances. The results show, that the light robust approach performs similar to the Γ-robust model in terms of realised robustness.

In terms of optimality gaps and computation time, we compare both models with respect to different Γ values (protection levels). The results show, that for both models an increase of the Γ value leads also to an increase in the computation time. This is because we are providing more protection against traffic uncertainty, which makes the problem harder to solve. The average computation time of the LR model is shorter than of the Γ-robust model. This is due to the fact that the LR model is much easier to solve due to the slack variables, which absorb any infeasible solution that may occur due to budget restrictions. We conclude that the LR model is capable of finding Γ-robust-alike designs, but for less costs compared to the Γ-robust model.

V. Conclusion and Future Work

In this paper, we propose a light-robustness optimization model for the network slice design and embedding problem. The model is evaluated against the nominal and Γ-robust problem formulations for several network topologies and traffic scenarios. The results show that the light robustness approach yields a better performance in terms of computation times and optimality gaps. The LR model provides robust solutions with a similar level of realized robustness compared to the Γ-robust model, but with a lower price of robustness.

VI. Acknowledgement

The research leading to these results has received funding from the Celtic-Plus cluster Project SENDATE, Secure DCI (ID C2015/3-1).

REFERENCES

[1] A. Galis and C.-L. I, "Towards 5G Network Slicing - Motivations and Challenges," *IEEE 5G Tech Focus*, vol. 1, 2017.

[2] V. Reddy, A. Baumgartner, and T. Bauschert, "Robust Embedding of VNF/Service Chains with Delay Bounds," *IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, 2016.

[3] ——, "Robust and Reliable Virtualized Network Infrastructures," *21th Conference of the International Federation of Operational Research Societies,IFORS*, 2017.

[4] M. Fischetti and M. Monaci, "Light Robustness," *Robust and Online Large-Scale Optimization - Lecture Notes in Computer Science*, vol. 5868, pp. 61–84, 2008.

[5] A. Schöbel, "Generalized light robustness and the trade-off between robustness and nominal quality," *Mathematical Methods of Operations Research*, vol. 80, no. 2, pp. 161–191, 2014. [Online]. Available: http://dx.doi.org/10.1007/s00186-014-0474-9

[6] S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly, "SNDlib 1.0–Survivable Network Design Library," in *Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium*, 2007.

[7] L. Yim-Kwong, "One slice at a time: Sdn/nfv to 5g network slicing," *Huawei Communicate Issues 81*, 2017.

[8] N. Alliance, "Description of Network Slicing Concept," *NGMN 5G Project Requirements & Architecture Work Stream E2E Architecture*, 2016.

[9] A. Fischer, J. Botero, M. Beck, H. de Meer, and X. Hesselbach, "Virtual network embedding: A survey," *IEEE Communications Surveys & Tutorials*, vol. 15, pp. 1888–1906, 2013.

[10] A. Baumgartner, V. Reddy, and T. Bauschert, "Mobile Core Virtualization: A Model for combined Virtual Core Network Function Placement and Topology Optimization," *IEEE Netsoft*, 2015.

[11] ——, "Combined Virtual Mobile Core Network Function Placement and Topology Optimization with Latency Bounds," *Fourth European Workshop on Software Defined Networks (EWSDN)*, 2015.

[12] D. Bertsimas and M. Sim, "The price of robustness," *Operations Research*, vol. 52, pp. 35–53, 2004.

[13] A. Baumgartner, T. Bauschert, A. Koster, and V. Reddy, "Optimisation Models for Robust and Reliable Network Slice Design: A Comparative Analysis," *IEEE Global Communications Conference: Communication QoS, Reliability and Modeling*, 2017.

[14] R. Fourer, D. M. Gay, and B. W. Kernighan, *AMPL: A Modelling Language for Mathematical Programming*, 2nd ed. Brooks/Cole Publishing Company, 2003.