A Fast, Scalable Meta-Heuristic for Network Slicing Under Traffic Uncertainty

Thomas Bauschert(✉) and Varun S. Reddy

Chair of Communication Networks, Technische Universität Chemnitz, 09126 Chemnitz, Germany
{thomas.bauschert,srva}@etit.tu-chemnitz.de

Abstract. Perceived to be one of the cornerstones of the emerging next generation (5G) networks, *Network slicing* enables the accommodation of multiple logical networks with diverse performance requirements on a common substrate platform. Of particular interest among different facets of network slicing is the problem of designing an individual network slice tailored specifically to match the requirements of the big-bandwidth next generation network services. In this work, we present an exact formulation for the network slice design problem under traffic uncertainty. As the considered mathematical formulation is known to pose a high degree of computational difficulty to state-of-the-art commercial mixed integer programming solvers owing to the inclusion of robust constraints, we propose a meta-heuristic based on ant colony optimisation algorithms for the robust network slice design problem. Experimental evaluation conducted on realistic network topologies from SNDlib reveals that the proposed meta-heuristic can indeed be an efficient alternative to the commercial mixed integer programming solvers.

Keywords: Ant colony optimisation · Meta-heuristics · Robust optimisation · Data uncertainty · Network slicing

1 Introduction

Powered by cloud computing and virtualisation techniques, Network softwarisation has paved the way for a holistic transformation of the underlying monolithic ICT infrastructure into a software-defined infrastructure where the legacy hardware-based networking components are replaced by software-based functions executed on general-purpose hardware [1,17]. While this increases the flexibility in deployment, operation and management of immersive next generation services, it is also expected to significantly improve the revenue of the network operators [22]. To leverage the benefits of network softwarisation, NGMN proposes the concept of *Network slicing* as a basis for enabling co-existence of a myriad of logical, self-sufficient, autonomous networks with distinct attributes on a shared substrate platform further opening up newer business prospects in the form of

P. A. Castillo et al. (Eds.): EvoApplications 2020, LNCS 12104, pp. 244–259, 2020.
https://doi.org/10.1007/978-3-030-43722-0_16

Over-The-Top (OTT) content/service providers such as Amazon, Hulu, Netflix [23,24]. Each of these logical networks, henceforth termed as network slices, represents an abstraction of a fraction of the shared physical substrate network resources tailored to meet customer/vertical-specific performance requirements.

Network slicing has gained traction in the recent years for its ability to provide "on-demand", end-to-end slices composed of an assortment of compute, storage, network, radio resources for a wide array of verticals/use-cases thereby empowering network operators to achieve the stringent yet diverse requirements of the emerging next generation (5G) services [19,24,26]. For example, deploying a network slice along a dynamic railway corridor will require very specific characteristics like high mobility, low latency and low throughput whereas a slice dedicated for collaborative automation within an industrial environment demands high reliability, guaranteed throughput with a relaxed mobility policy. Notwithstanding the imminent benefits, realising network slicing in practise entails several algorithmic challenges ranging from efficient resource provisioning mechanisms to online admission control policies [29].

Earlier proposals handled network slicing exclusively as a resource partitioning problem as in the virtual network embedding (VNE) problem addressed in [10,16,18]. Recent works have attempted to broaden the problem statement from merely assigning substrate network resources to pre-defined network slice requests to encompassing the design of the individual network slices (i.e., the network slice topology, the number of required virtual functions, their dimensioning, and the interconnections) as well as considering the impact of the stochastic traffic demands in the problem formulation [3,4]. In this work, we consider the problem of designing a large-scale logical network slice capable of handling the uncertain nature of traffic demands proposed in [4]. We first present an exact formulation for the network slice design problem (NSDP) under uncertainty where the uncertain traffic demands in the network slice request are characterised using the multi-band uncertainty model proposed by Büsing and D'Andreagiovanni [9]. The multi-band uncertainty model extends the Γ-robustness model [5] by partitioning the deviation interval into multiple sub-intervals thereby returning solutions with reduced conservatism without compromising on their robustness guarantees against traffic uncertainty.

The robust counterpart of the network slice design problem is known to pose computational challenges to state-of-the-art commercial mixed integer programming (MIP) solvers due to the inclusion of robust constraints. This observation is in line with prior works that tackle the presence of uncertainty in the problem formulation as in [10,11]. Inspired by the performance of the ant colony optimisation (ACO) algorithms on several real-world problems such as the travelling salesman problem (TSP), vehicle routing problem (VRP) and so on [8], we propose a meta-heuristic approach to solve the robust network slice design problem. Our approach employs the $\mathcal{MAX} - \mathcal{MIN}$ ant system [27] variant of the ACO algorithms within a hyper-cube framework [7] to guide the variable fixing procedure for the considered problem. Experiments conducted on a range of realistic network topologies from SNDlib [25] reveal that the proposed ACO-based meta-heuristic returns solutions of improved quality in comparison to the commercial MIP solvers.

The paper is structured as follows: We present a mathematical model for the robust NSDP by employing a multi-band uncertainty model in Sect. 2 followed by our ant colony optimisation approach to solve the robust NSDP in Sect. 3. Experimental results are presented in Sect. 4 with a conclusion in Sect. 5.

2 Robust Network Slice Design

In this section, we present the mathematical model for the robust network slice design problem. The physical substrate network infrastructure is modelled as an undirected graph G composed of a network of nodes V connected by edges E. The residual (i.e. unused) capacities of the nodes and edges are denoted by $c_v^0 \in \mathbb{R}_{\geq 0}$ and $c_e^0 \in \mathbb{R}_{\geq 0}$, respectively. The costs per allocated unit of the substrate network resources to the network slice is given by $\gamma_v^0 \in \mathbb{R}_+$ and $\gamma_e^0 \in \mathbb{R}_+$, respectively. Additionally, we allow the expansion of substrate network capacity in discrete steps of size $c_v \in \mathbb{Z}_+$ and $c_e \in \mathbb{Z}_+$ incurring a one-time installation costs denoted by $\gamma_v \in \mathbb{R}_+$ and $\gamma_e \in \mathbb{R}_+$, respectively.

We assume the network slice to be composed of a set of demands K, where each demand k is associated with an uncertain traffic volume $d^k \in \mathbb{R}$ that must be routed from source s^k to destination t^k through a sequence of network functions $F^k = \langle f_i \mid i \in \mathbb{N} \rangle$ usually derived from the corresponding service graph [2]. To comply with the restrictions arising due to technological, economic or geographical limitations, or security issues, we assume that only a subset of the substrate nodes $V(f_i) \subseteq V$ are capable of offering the functionality f_i in virtual modules of size $c_f \in \mathbb{Z}_+$.

In this work, we cast the NSDP under uncertainty within a layered graph framework [20]. This is achieved through the following steps: The given undirected graph G is converted into a directed graph $G' = (V, A)$, where for every edge $\{u, v\} \in E$ we add two arcs (u, v), (v, u) to A. The digraph G' is then transformed into a layered graph $G_L^k = (V_L^k, A_L^k)$, where the newly-constructed vertex set $V_L^k = \{v_i \mid v \in V, 1 \leq i \leq |F^k| + 1\}$ is obtained by creating $|F^k| + 1$ copies of the nodes in V. We retain the connection of the vertices according to the original digraph G' to obtain $A_L^{k,E} = \{(u_i, v_i) \mid (u, v) \in A, 1 \leq i \leq |F^k| + 1\}$. We then encode the potential routing of the demand k through the network functions F^k by creating a set of inter-layer arcs $A_L^{k,V} = \{(v_i, v_{i+1}) \mid v \in V(f_i), 1 \leq i \leq |F^k|\}$. Finally, we let $A_L^k = A_L^{k,V} \cup A_L^{k,E}$ to complete the transformation.

A formal problem statement for the robust network slice design problem can now be stated as:

Definition 1. The robust network slice design problem. *Let $G = (V, E)$ denote the physical substrate network infrastructure with residual node and edge capacities $c^0 \colon V \to \mathbb{R}_{\geq 0}$ and $c^0 \colon E \to \mathbb{R}_{\geq 0}$ expressed in bits per second, whose costs per occupied unit bandwidth is given as $\gamma^0 \colon V \to \mathbb{R}_{\geq 0}$ and $\gamma^0 \colon E \to \mathbb{R}_{\geq 0}$. Let the node and edge capacities be expanded, optionally, in discrete steps of size*

$c\colon V \to \mathbb{Z}_+$ and $c\colon V \to \mathbb{Z}_+$ costing $\gamma\colon V \to \mathbb{Z}_{\geq 0}$ and $\gamma\colon E \to \mathbb{Z}_{\geq 0}$ per unit module of installed capacity. Given a set of demands K, where each demand k is associated with an uncertain traffic flow of volume $d\colon K \to \mathbb{R}$ from s^k to t^k through a sequence of network functions F^k, the robust network slice design problem on a transformed layered graph instance $G_L^k = (V_L^k, A_L^k)$ concerns routing the uncertain traffic volume of each demand over the layered graph instance such that the cumulative costs of substrate resource utilisation and potential capacity expansions to host the network slice are minimised.

To this end, we employ the following family of decision variables to model the robust NSDP. Variables $x_a^k \in \{0,1\}$ indicate if demand k is routed over arc a of the layered graph. Variables $y_{vf} \in \mathbb{Z}_{\geq 0}$ specify the number of virtual modules allocated to the network function f hosted on node v. Variables $y_v \in \mathbb{Z}_{\geq 0}$ and $y_e \in \mathbb{Z}_{\geq 0}$ specify the number of capacity modules installed on the nodes and edges of the substrate network, respectively. The robust network slice design problem takes the form:

$$\min \max_{d \in \mathcal{D}} \quad \sum_{v \in V} \gamma_v y_v + \sum_{f \in F} \sum_{v \in V(f)} \gamma_v^0 c_f y_{vf} + \sum_{e \in E} \gamma_e y_e + \sum_{e \in E} \sum_{k \in K} \sum_{a \in A_L^{k,E}(e)} \gamma_e^0 d^k x_a^k \tag{1a}$$

$$\text{s.t.} \quad \sum_{a \in \delta_v^+} x_a^k - \sum_{a \in \delta_v^-} x_a^k = b^k \qquad \forall v \in V_L^k, k \in K \tag{1b}$$

$$\sum_{k \in K : f \in F^k} \sum_{a \in A_L^{k,V}(f,v)} d^k x_a^k \leq c_f y_{vf} \qquad \forall d \in \mathcal{D}, f \in F, v \in V(f) \tag{1c}$$

$$\sum_{f \in F : v \in V(f)} c_f y_{vf} \leq c_v^0 + c_v y_v \qquad \forall v \in V \tag{1d}$$

$$\sum_{k \in K} \sum_{a \in A_L^{k,E}(e)} d^k x_a^k \leq c_e^0 + c_e y_e \qquad \forall d \in \mathcal{D}, e \in E \tag{1e}$$

$$x_a^k \in \{0,1\} \tag{1f}$$

$$y_{vf}, y_v, y_e \in \mathbb{Z}_{\geq 0} \tag{1g}$$

Objective function (1a) minimises, for the worst-case realisation of the uncertain demands, the sum of capacity consumption and capacity installation costs for accommodating the network slice request on the substrate network infrastructure. Constraints (1b) are standard flow conservation constraints, where $b^k = 1$ if $v = s^k$, $b^k = -1$ if $v = t^k$, else 0. Constraints (1c) denote the capacity requirements of the network functions on the substrate nodes. Constraints (1d) and (1e) ensure that the consumption of the substrate network resources doesn't exceed the available capacity (residual and installed together) at the substrate nodes and edges, respectively. Note that constraints (1c) and (1e) must hold good for every realisation of the uncertain traffic flow d contained in the uncertainty set $\mathcal{D}$.

2.1 Multi-band Uncertainty Model

We consider the multi-band uncertainty model proposed in [9] to design the uncertainty set $\mathcal{D}$ for the robust NSDP. Under this model, we assume the uncertain coefficient $d^k, \forall k \in K$ to be an independent and bounded random variable represented by the nominal (or forecast) traffic volume $\bar{d}^k > 0$ and a deviation from the forecast traffic volume $\hat{d}^k$ belonging to the deviation range $[\hat{d}^k_{R^-}, \hat{d}^k_{R^+}]$, where $\hat{d}^k_{R^-} < 0$ and $\hat{d}^k_{R^+} > 0$ represent the maximum negative and positive deviation from the forecast traffic volume $\bar{d}^k$. The deviation range of each uncertain coefficient is partitioned into $R = R^- + 1 + R^+$ disjoint ranges on the basis of R deviation values:

$$-\infty < \hat{d}^k_{R^-} < \cdots < \hat{d}^k_{-1} < \hat{d}^k_0 = 0 < \hat{d}^k_1 < \cdots < \hat{d}^k_{R^+} < \infty$$

A band $r \in \{R^- + 1, \cdots, R^+\}$ now corresponds to the range $(\hat{d}^k_{r-1}, \hat{d}^k_r]$, and band $r = R^-$ corresponds to the single value $\hat{d}^k_{R^-}$. We impose a lower bound θ_r and an upper bound Θ_r on the number of realisations of the uncertain traffic coefficients in band r, where $0 \leq \theta_r \leq \Theta_r \leq |K|$ with $\Theta_0 = |K|$. To guarantee a feasible realisation of the uncertain traffic coefficients, we ensure $\sum_{r \in \{R^-, \cdots, R^+\}} \theta_r \leq |K|$.

The multi-band uncertainty set for the robust NSDP can now be defined as:

$$\mathcal{D} = \{d^k \in \mathbb{R} \mid d^k = \bar{d}^k + \sum_{r=R^-}^{r=R^+} \hat{d}^k_r z^k_r, \forall k \in K, z \in \mathcal{Z}\}$$

where

$$\begin{aligned} \mathcal{Z} = \Big\{ \theta_r \leq \sum_{k \in K} z^k_r \leq \Theta_r \qquad & \forall r \in \{R^-, \cdots, R^+\} \\ \sum_{r=R^-}^{r=R^+} \hat{d}^k_r z^k_r = 1 \qquad & \forall k \in K \\ z^k_r \in \{0, 1\} \Big\} & \end{aligned}$$

2.2 The Multi-band Robust NSDP

Since the goal of the decision maker is to be protected against the worst-case realisation of the uncertain traffic coefficients in multi-band uncertainty set $\mathcal{D}$, we introduce additional terms $\text{DEV}^{\Gamma}_{vf}(x, \mathcal{D})$ and $\text{DEV}^{\Gamma}_{e}(x, \mathcal{D})$ in every equation affected by the uncertain coefficients d^k in the robust NSDP indicating additional capacities required at the network functions and the substrate edges in order to cope with traffic uncertainty. These terms, however, render the robust NSDP formulation non-linear and can be linearised by transforming the inner maximisation problem into its dual equivalent as follows:

$$\text{DEV}_{vf}^{\Gamma}(x,\mathcal{D}) = \max \sum_{k\in K:f\in F^k} \sum_{a\in A_L^{k,V}(f,v)} \sum_{r\in R} \hat{d}_r^k x_a^k z_{rvf}^k \tag{2a}$$

$$\text{s.t.} \sum_{k\in K:f\in F^k} z_{rvf}^k \leq \Gamma_r \qquad \forall r \in R \tag{2b}$$

$$\sum_{r\in R} z_{rvf}^k \leq 1 \qquad \forall k \in K : f \in F^k \tag{2c}$$

$$z_{rvf}^k \in \{0,1\} \tag{2d}$$

where binary variables $z_{rvf}^k, \forall k \in K : f \in F^k, r \in R$ take the value 1 if the coefficient $\hat{d}^k$ falls in the r-th band. Objective function (2a) maximises the worst-case deviation for each constraint (1c). Constraints (2b) ensure that not more than Γ_r coefficients deviate in each band whereas constraints (2c) impose that each coefficient deviates in at most one band. As the constraint matrix of the problem is totally unimodular [9], variables z_{rvf}^k can be relaxed. By virtue of strong LP duality, the resulting LP can be replaced by its dual equivalent:

$$\text{DEV}_{vf}^{\Gamma}(x,\mathcal{D}) = \min \sum_{r\in R} \Gamma_r \pi_{vf}^r + \sum_{k\in K:f\in F^k} \rho_{vf}^k \tag{3a}$$

$$\text{s.t. } \pi_{vf}^r + \rho_{vf}^k \geq \sum_{a\in A_L^{k,V}(f,v)} \hat{d}_r^k x_a^k \qquad \forall r \in R, k \in K : f \in F^k \tag{3b}$$

$$\pi_{vf}^r, \rho_{vf}^k \in \mathbb{R}_{\geq 0} \tag{3c}$$

where $\pi_{vf}^r, \forall r \in R$ and $\rho_{vf}^k, \forall k \in K : f \in F^k$ are dual variables. The dual equivalent of $\text{DEV}_e^{\Gamma}(x,\mathcal{D})$ can be obtained in a similar fashion. Substituting the inner maximisation problems with their dual equivalents, we can obtain the compact reformulation of the robust NSDP for the multi-band uncertainty set.

3 An ACO-Based Meta-Heuristic for the Robust NSDP

In the previous section, we presented a compact reformulation of the robust NSDP for the multi-band uncertainty set. Through the inclusion of (hard) robust constraints in the formulation, we restrict the solution space to contain only those solutions that are robust against traffic uncertainty. Such inclusion is known to pose computational challenges to state-of-the-art commercial MIP solvers such as CPLEX as observed in [4,11,12]. To circumvent this computational difficulty, we present a resolution method that employs the $\mathcal{MAX}-\mathcal{MIN}$ ant system [27] variant of the ACO algorithms within a hyper-cube framework [7] to solve the robust NSDP.

Ant colony optimisation is a stochastic meta-heuristic that draws influence from the foraging behaviour of real ants [13]. A standard ACO meta-heuristic involves a family of computing agents iteratively constructing candidate solutions for the considered problem by means of probabilistic sampling from a set of solution components. Each solution component is typically associated with two values of attractiveness - an *a-priori* pheromone trail value and an *a-posteriori* desirability value - that influence the transition probability of the solution components. During the construction phase, a computing agent makes a probabilistic move from the current state (with a partial solution) to the next state by augmenting the incumbent solution with a new solution component until a complete solution is found. The probability of selecting the next move from a set of plausible moves is usually governed by a state transition rule. Upon the completion of the solution construction phase, the pheromone trail values of the solution components are reinforced essentially creating a positive feedback mechanism to aid the construction of improved solutions. The process is repeated until a termination condition is satisfied. For an exhaustive introduction to the theory and applications of ACO algorithms, we refer the reader to the works of [8, 14].

3.1 A Meta-Heuristic for the Multi-band Robust NSDP

In this section, we sketch our meta-heuristic that employs the $\mathcal{MAX} - \mathcal{MIN}$ ant system within a hyper-cube framework to guide the decision making strategy for our robust NSDP. A crucial decision within the problem formulation is to determine the routing paths for the network slice demands K in given substrate network infrastructure. Once the routing template for the network slice demands is identified, we can easily compute the capacities required to support the identified routing template to complete the robust network slice design solution. We accomplish the vital task of identifying the routing template through the ant-based solution construction module. Algorithm 1 presents a high-level description of the proposed ACO-based meta-heuristic. In the following, we explain, in further detail, the different phases involved in the meta-heuristic.

Initialisation. In the first step, we set the global-best (x^{gb}, y^{gb}) and restart-best (x^{rb}, y^{rb}) ant solutions to a null value, the convergence factor cf to 0, and the boolean variable *gb_update* to FALSE. Similar to [7], the pheromone trail values T are initialised to 0.5.

Ant-Based Solution Construction. The algorithm starts with a family of $\Psi > 0$ computing agents that set out to build feasible solutions to the multi-band robust NSDP at each iteration. Within the inner construction loop, agent $\psi \in \Psi$ incrementally constructs the solution to the problem as outlined in the CONSTRUCTNSDP(T) module. For every $k \in K$, agent ψ probabilistically selects a path p from the set of candidate paths P_k over which the traffic of the demand k can be possibly routed. The probabilities of the candidate paths to feature in the solution are determined by the following state transition rule proposed in [21]:

$$\mathfrak{pr}_{pk}^{\psi} = \frac{\alpha \cdot \tau_{pk} + (1-\alpha) \cdot \eta_{pk}}{\sum_{p \in P_k} \alpha \cdot \tau_{pk} + (1-\alpha) \cdot \eta_{pk}} \tag{4}$$

Algorithm 1. A meta-heuristic for the multi-band robust NSDP

Input: Problem instance, parameter file
$\left(x^{gb}, y^{gb}\right) := \emptyset; \left(x^{rb}, y^{rb}\right) := \emptyset; cf = 0; gb\_update :=$ FALSE
for all $\tau_{pk} \in \mathrm{T}$ **do**
    $\tau_{pk} = 0.5$
end for
while an arrest condition is not met **do**
    $\mathcal{S} := \emptyset$
    for $\psi = 1 : \Psi$ **do**
        $(\bar{x}, \bar{y})_\psi :=$ ConstructNSDP (T)
        $\mathcal{S} := \mathcal{S} \cup \{(\bar{x}, \bar{y})_\psi\}$
    end for
    if (iter % 50) = 0 **then**
        for $\psi = \Psi + 1 : \Psi + 3$ **do**
            $(\bar{x}, \bar{y})_\psi :=$ ConstructElitistNSDP (T)
            $\mathcal{S} := \mathcal{S} \cup \{(\bar{x}, \bar{y})_\psi\}$
        end for
    end if
    $\left(x^{ib}, y^{ib}\right) := \mathrm{argmin}\{f(\bar{x}, \bar{y}) \mid (\bar{x}, \bar{y}) \in \mathcal{S}\}$
    if $f\left(x^{ib}, y^{ib}\right) < f\left(x^{rb}, y^{rb}\right)$ **then**
        $\left(x^{rb}, y^{rb}\right) := \left(x^{ib}, y^{ib}\right)$
    end if
    if $f\left(x^{ib}, y^{ib}\right) < f\left(x^{gb}, y^{gb}\right)$ **then**
        $\left(x^{gb}, y^{gb}\right) := \left(x^{ib}, y^{ib}\right)$
    end if
    ReinforcePheromone $\left(cf, gb\_update, \mathrm{T}, \left(x^{ib}, y^{ib}\right), \left(x^{rb}, y^{rb}\right), \left(x^{gb}, y^{gb}\right)\right)$
    $cf :=$ ComputeConvergenceFactor (T)
    if $cf > 0.999$ **then**
        if $gb\_update =$ TRUE **then**
            for all $\tau_{pk} \in \mathrm{T}$ **do**
                $\tau_{pk} = 0.5$
            end for
            $\left(x^{rb}, y^{rb}\right) := \emptyset$
            $gb\_update :=$ FALSE
        else
            $gb\_update :=$ TRUE
        end if
    end if
end while
return $\left(x^{gb}, y^{gb}\right)$

where $\alpha \in [0, 1]$ controls the level of influence of the pheromone trail value τ and the desirability value η. We remark that this rule has the advantage of using simpler computational operations and lesser parameters over the classical transition rule. Upon termination of the inner construction cycle by agent ψ, we execute the variable fixing strategy for the robust NSDP as follows: For every demand k, we activate the arcs comprising the chosen routing path while deactivating the remaining arcs, consequently arriving at the fixing of the binary routing variables x_a^k. Having established a complete routing template for the network slice request, we determine the number capacity modules y_{vf} to be allocated network functions $f \in F$ using (1c). We then check if the substrate network has enough resources to fulfill the routing template under traffic uncertainty thereby deriving the fixings for variables y_v, and y_e, respectively. If no constraint of the problem is violated, we declare that the computing agent ψ has found a complete feasible solution to the multi-band robust NSDP, the costs for which can be assessed by using the equation (1a). A formal representation of the solution construction is depicted in Algorithm 2. At the end of the solution construction phase, we update variable $\left(x^{ib}, y^{ib}\right)$ to contain the best solution found by the ants in the current iteration.

Algorithm 2. Ant-based solution construction (ConstructNSDP)

```
Input: Problem instance, T
𝔓𝔯 := ∅; 𝒫 = ∅
for all k ∈ K do
    for all p ∈ P_k do
        𝔭𝔯_pk := StateTransitionRule (T)
    end for
    p := ProbabilisticSampling (𝔓𝔯, P_k)
    𝒫 := 𝒫 ∪ {p}
end for
(x̄, ȳ) := VariableFixing (𝒫)
return (x̄, ȳ)
```

Pheromone Reinforcement. Traditional pheromone reinforcement models consider only the iteration-best solution to update the pheromone trails. We employ a rather sophisticated model to update the pheromone trail values of the candidate paths wherein the pheromone deposit each candidate path receives is influenced by three different solutions: the iteration-best $\left(x^{ib}, y^{ib}\right)$, the restart-best $\left(x^{rb}, y^{rb}\right)$ and the global-best $\left(x^{gb}, y^{gb}\right)$ solutions. The level of influence of each of these solutions on the pheromone reinforcement depends on the state of convergence of the algorithm indicated by convergence factor cf. An update to the pheromone trail values is now performed by the following rule:

$$\tau_{pk} := \min\{\max\{\tau^-, \tau_{pk} + \varrho \cdot (\omega_{pk} - \tau_{pk})\}, \tau^+\} \tag{5}$$

where $\varrho \in (0, 1]$ is the pheromone evaporation rate, and τ^+, τ^- are the upper and lower bounds of the pheromone trail values. The update ensures that pheromone trail values of the solution components remain in the range $[\tau^-, \tau^+]$. Finally, parameter $\omega_{pk} \in [0, 1]$ is expressed as:

$$\begin{aligned} \omega_{pk} := \kappa_{ib} \cdot \delta\big(\left(x^{ib}, y^{ib}\right), (p, k)\big) + \kappa_{rb} \cdot \delta\big(\left(x^{rb}, y^{rb}\right), (p, k)\big) \\ + \kappa_{gb} \cdot \delta\big(\left(x^{gb}, y^{gb}\right), (p, k)\big) \end{aligned} \tag{6}$$

where κ_{ib}, κ_{rb}, and κ_{gb} are the weights of the solutions (x^{ib}, y^{ib}), (x^{rb}, y^{ib}), and (x^{gb}, y^{ib}), respectively such that the total sum of the weights doesn't exceed 1. These weights are chosen according to the schedule specified in Table 1. The term $\delta\big((x, y), (p, k)\big) \in \{0, 1\}$ takes the value 1 if the solution component (p, k) features in the solution (x, y), else 0.

Table 1. The schedule for weights κ_{ib}, κ_{rb}, and κ_{gb} depending on the convergence factor cf and the Boolean update variable *gb_update*.

	gb_update = FALSE				*gb_update* = TRUE
	$cf < 0.4$	$cf \in [0.4, 0.6)$	$cf \in [0.6, 0.8)$	$cf \geq 0.8$	
κ_{ib}	1	2/3	1/3	0	0
κ_{rb}	0	1/3	2/3	1	0
κ_{gb}	0	0	0	0	1

Convergence Factor. In the final phase of the solution construction, we compute the convergence factor cf which estimates the state of convergence of the algorithm using the formula (7). A convergence factor of $cf \geq 0.999$ indicates that the algorithm has converged and the probability of finding better solutions in the future iterations is extremely low. To overcome this, the boolean variable *gb_update* is set to TRUE and the pheromone trail values are reset to 0.5.

$$cf := 2 \times \left(\left(\frac{\sum_{\tau_{pk} \in \mathrm{T}} \max(\tau^{+} - \tau_{pk}, \tau_{pk} + \tau^{-})}{|\mathrm{T}| \times (\tau^{+} - \tau^{-})} \right) - 0.5 \right) \tag{7}$$

This concludes one complete iteration of the ACO-based meta-heuristic for the robust NSDP. To accelerate the search towards solutions of improved quality, at every 50^{th} iteration of the algorithm, we introduce a small number of elitist agents that conduct the probabilistic search on a smaller pool of the shortest candidate paths $P_k^e \subset P_k$ for each demand k. As a result of working on a smaller solution space, these elitist ants may not only find solutions of (possibly) improved quality but also influence the search towards such solutions in the subsequent iterations.

4 Performance Evaluation

In this section, we validate the performance of the proposed solution methodologies using realistic problem instances. We consider ten different network topologies from SNDlib [25] to model the underlying substrate network infrastructure. For each network topology, the residual capacities of the nodes are drawn at random from the tuple (2.0,3.0,4.0) Tbps weighted by (0.3,0.4,0.3), and the cost per occupied unit of the node resources is set to EUR 12.5/Gbps. Auxiliary capacity modules of size 40 Gbps can be installed on the nodes at a cost of EUR 50,000 per module. The residual capacities of the edges are sampled at random from (0.2,0.3,0.4) Tbps with probability (0.3,0.4.0.3). The cost per occupied unit of the edge resources is set to EUR 5/Gbps. The capacity of the edges can be optionally expanded in steps of size 10 Gbps with each module costing EUR 20,000.[1]

We consider the design of a network slice for the use-case of next generation emergency services for which the set of network functions comprising the service is assumed to be $F = \langle \text{VF1}, \text{VF2}, \text{VF3}, \text{VF4}, \text{VF5} \rangle$. These network functions can be instantiated on substrate nodes in virtual modules of size 1 Gbps. For every network function $f \in F$, we draw samples of size $\lceil |V|/2 \rceil$ uniformly at random from the physical substrate node set V to construct the candidate physical substrate nodes that support the functionality. Historical traffic traces for the demands of the network slice are generated using the following three-step procedure: First, for every demand k, we randomly draw a value from the tuple (10.0,20.0,50.0) Gbps with probability (0.3,0.4,0.3). Second, to enforce that the traffic coefficients d^k are normal distributed, for each demand k, we draw 1440 samples at random from a normal distribution of mean 0 and standard deviation of 50% of the respective value chosen in the first step. In the final step, the value determined in the first step is added to each of these 1440 samples to obtain the historical traffic traces. Each of the constructed problem instances is now solved using the proposed solution methodologies for bands $|R| \in \{2, 4, 6\}$, encompassing the 68-95-99.7 areas of the normal distribution of the uncertain traffic demands.

We employ a single-threaded Linux machine with Intel® Core™ i3-3120M CPU @ 2.5GHz and 8 GB RAM to conduct the performance evaluation. The compact reformulation (1) is implemented in JuMP v0.18 [15] —a modelling language for mathematical optimisation embedded in Julia v0.6 [6] and is solved using IBM® ILOG® CPLEX® Optimization Studio v12.7.1 [28] with a time limit of 3600 s. The ACO-based meta-heuristic is coded in Julia v0.6 and a truncated time limit of 2400 s is imposed on each problem instance. The parameters of the ACO-based meta-heuristic are hand-tuned to the following values: Candidate paths considered for the construction of the network slice design solution are computed using the $k-$shortest path algorithm [30], where k is set to 10. Six computing agents are considered for the ant-based solution construction. Additionally, three computing agents periodically conduct an elitist search to find improved solutions in quick intervals. The lower and upper bounds for the pheromone trail values are fixed at $\tau^- = 0.001$ and $\tau^+ = 0.999$, respectively and the pheromone evaporation rate ϱ is set to 0.1.

[1] As large coefficients are known to pose some problems at various stages of the solution process in CPLEX, the capacities and demands were scaled down by a factor of 1 Gbps, and the costs by EUR 1000.

We evaluate the performance of the proposed solution methods by comparing the absolute gap of the slice design solutions obtained from the respective methods. The absolute gap of a solution is defined as the difference between its best integer objective value and the best lower bound to the problem (returned by CPLEX). Table 2 reports the absolute gaps of the robust network slice design solutions obtained from the considered solutions methodologies, where $|R|$ indicates the number of bands employed in the multi-band uncertainty set to capture the uncertain traffic coefficients, "LB" indicates the lower bound to the considered problem instance which, in this case, is produced by CPLEX, and the values under columns labelled "Exact" and "ACO" indicate the absolute gaps of the robust network slice design solutions obtained from the commercial MIP solver CPLEX and the ACO-based meta-heuristic.

Firstly, the absolute gaps of the robust network slice design solutions from CPLEX are observed to be exceedingly high as compared to their ACO-based counterparts consequently rendering these solutions cost-ineffective to implement in practise. In addition, for 27% of the problem instances, CPLEX couldn't find a non-trivial solution. Despite solving these failed instances using a computationally powerful IBM Decision Optimisation on Cloud service, we observed a negligible improvement in the absolute gaps of the solutions (in the range of 0–3%). We remark that this behaviour is in line with many of the previous works where the robust counterparts were often hard to solve for the commercial MIP solvers thereby justifying the need for scalable heuristic methods [4,11]. The ACO-based meta-heuristic, on the other hand, performs significantly better by yielding solutions of reduced absolute gap for 97% of the considered instances. In most cases, we observe that these solutions are at least an order of magnitude better when compared to the state-of-the-art commercial MIP solvers. The improved performance of the ACO-based meta-heuristic is particularly evident for larger problem instances (i.e., GERMANY50, JANOS-US, PIORO40) thereby establishing the effectiveness of the proposed ACO-based meta-heuristic.

In order to further assess the performance of the meta-heuristic, we develop a simple greedy algorithm to solve the robust NSDP and compare the resulting solution costs with that of the proposed ACO-based meta-heuristic. The algorithm employs a greedy strategy of routing each demand k of the network slice request over the shortest path in the substrate network infrastructure. In the next step, the binary routing variables x_a^k are fixed to reflect the shortest path routings for each demand. After setting up the routing template for the network slice, the remaining variables and the costs of the constructed solution are derived similar to the procedure outlined in Algorithm 2. While the solutions returned by the simple greedy algorithm outperform those of CPLEX for 29 of the considered problem instances, these solutions are still found to be of inferior quality when compared to those of our ACO-based meta-heuristic.

Table 2. Numerical results for the robust network slice design problem.

				Absolute gap		
ID	Network	$\|R\|$	LB	Exact	Greedy	ACO
1	FRANCE	2	5.21E+4	-	1.48E+4	6.40E+3
		4	6.40E+4	7.89E+6	1.43E+4	7.59E+3
		6	6.76E+4	9.48E+6	1.41E+4	9.05E+3
2	GEANT	2	1.35E+5	-	1.66E+4	1.18E+4
		4	1.51E+5	9.26E+6	1.80E+4	1.27E+4
		6	1.55E+5	1.12E+7	1.87E+4	1.49E+4
3	GERMANY50	2	1.80E+5	-	3.30E+4	2.54E+4
		4	0.00E+0	3.32E+7	2.48E+5	2.43E+5
		6	0.00E+0	4.02E+7	2.60E+5	2.57E+5
4	INDIA35	2	1.26E+5	5.24E+4	3.26E+4	2.17E+4
		4	0.00E+0	2.80E+7	1.88E+5	1.80E+5
		6	0.00E+0	3.36E+7	1.96E+5	1.90E+5
5	JANOS-US	2	2.21E+5	1.14E+5	2.67E+4	1.67E+4
		4	2.49E+5	1.51E+7	2.57E+4	1.94E+4
		6	0.00E+0	1.86E+7	2.82E+5	2.78E+5
6	NEWYORK	2	3.10E+4	-	1.59E+4	8.79E+3
		4	3.82E+4	6.85E+6	1.49E+4	8.76E+3
		6	3.98E+4	8.17E+6	1.50E+4	9.51E+3
7	NOBEL-EU	2	9.36E+4	-	1.92E+4	9.14E+3
		4	1.09E+5	-	1.96E+4	1.11E+4
		6	0.00E+0	1.01E+7	1.34E+5	1.27E+5
8	NOBEL-US	2	1.54E+4	-	4.12E+3	5.28E+2
		4	1.77E+4	4.97E+5	3.95E+3	9.35E+2
		6	1.84E+4	3.75E+2	3.88E+3	7.53E+2
9	NORWAY	2	2.26E+5	2.03E+5	1.74E+4	1.09E+4
		4	0.00E+0	2.03E+7	2.76E+5	2.71E+5
		6	0.00E+0	2.45E+7	2.84E+5	2.82E+5
10	PIORO40	2	2.02E+5	-	4.63E+4	3.08E+3
		4	0.00E+0	3.93E+7	2.85E+5	2.77E+5
		6	0.00E+0	4.71E+7	2.95E+5	2.90E+5

We now focus on the performance of the ACO-based meta-heuristic over the course of its execution. Figure 1 traces the evolution of the costs of the robust network slice design solutions during the ACO-based meta-heuristic execution for two exemplary networks: JANOS-US and PIORO40. We observe from Fig. 1 that the final best solutions yielded by the ACO-based meta-heuristic show, on

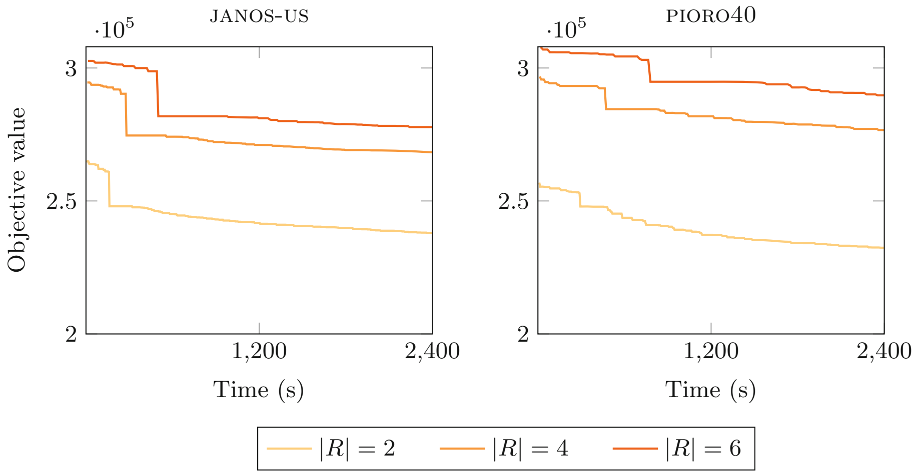

Fig. 1. Evolution of the objective value of the robust network slice design solutions yielded by the ACO-based meta-heuristic for bands $|R| \in \{2, 4, 6\}$.

average, a cost improvement of 8.27% against the initial best solutions over the course of 2400 s. This improvement is noticeably higher (9.91%) for instances with $|R| = 2$, and reduces (to 7.85% and 7.05%) with the increase in the number of bands. This can be explained as follows: The computational difficulty of the considered instances increases with the number of bands employed to capture the uncertain traffic coefficients. As a result, each computing agent ψ consumes more time at every iteration to construct a feasible solution to the robust NSDP, limiting the number of ant-based solution construction iterations. We remark that the computational burden of repetitive evaluation of the objective function after every iteration of the ant-based solution construction can be lessened by means of fitness approximation techniques. Re-defining the cost function (1a) using an approximate cost function may not only reduce the time consumed to execute the CONSTRUCTNSDP module in Algorithm 1 but can also accelerate the search process of the meta-heuristic through quick identification of good solutions in the modified solution landscape.

5 Conclusion

In this work, we propose a meta-heuristic based on the ACO algorithms to solve the robust network slice design problem. Experimentations conducted using realistic problem instances reveal that the proposed heuristic is capable of yielding solutions with improved absolute gaps in comparison to the commercial MIP solver CPLEX. As a further step, the ACO-based meta-heuristic can be integrated with a perturbative method relying on an exact solver to further improve the quality of the obtained network slice design solutions. We plan to evaluate the performance of the integrated ACO-based meta-heuristic with the existing solution methods [4] for the robust network slice design problem.

References

1. Afolabi, I., Taleb, T., Samdanis, K., Ksentini, A., Flinck, H.: Network slicing and softwarization: a survey on principles, enabling technologies, and solutions. IEEE Commun. Surv. Tutor. **20**(3), 2429–2453 (2018). Thirdquarter
2. Barcelo, M., Llorca, J., Tulino, A.M., Raman, N.: The cloud service distribution problem in distributed cloud networks. In: 2015 IEEE International Conference on Communications (ICC), pp. 344–350, June 2015
3. Baumgartner, A., Bauschert, T., D'Andreagiovanni, F., Reddy, V.S.: Towards robust network slice design under correlated demand uncertainties. In: ICC 2018–2018 IEEE International Conference on Communications, pp. 1–7, May 2018
4. Bauschert, T., Reddy, V.S.: Genetic algorithms for the network slice design problem under uncertainty. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO 2019, pp. 360–361. ACM, New York (2019)
5. Bertsimas, D., Sim, M.: The price of robustness. Oper. Res. **52**(1), 35–53 (2004)
6. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: a fresh approach to numerical computing. SIAM Rev. **59**(1), 65–98 (2017)
7. Blum, C., Dorigo, M.: The hyper-cube framework for ant colony optimization. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **34**(2), 1161–1172 (2004)
8. Blum, C.: Ant colony optimization: introduction and recent trends. Phys. Life Rev. **2**(4), 353–373 (2005)
9. Bsing, C., D'Andreagiovanni, F.: New results about multi-band uncertainty in robust optimization. In: Klasing, R. (ed.) SEA 2012. LNCS, vol. 7276, pp. 63–74. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30850-5_7
10. Coniglio, S., Koster, A., Tieves, M.: Data uncertainty in virtual network embedding: robust optimization and protection levels. J. Netw. Syst. Manage. **24**(3), 681–710 (2016). https://doi.org/10.1007/s10922-016-9376-x
11. D'Andreagiovanni, F., Krolikowski, J., Pulaj, J.: A fast hybrid primal heuristic for multiband robust capacitated network design with multiple time periods. Appl. Soft Comput. **26**, 497–507 (2015)
12. D'Andreagiovanni, F., Mett, F., Nardin, A., Pulaj, J.: Integrating LP-guided variable fixing with MIP heuristics in the robust design of hybrid wired-wireless FTTx access networks. Appl. Soft Comput. **61**, 1074–1087 (2017)
13. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **26**(1), 29–41 (1996)
14. Dorigo, M., Blum, C.: Ant colony optimization theory: a survey. Theoret. Comput. Sci. **344**(2), 243–278 (2005)
15. Dunning, I., Huchette, J., Lubin, M.: JuMP: a modeling language for mathematical optimization. SIAM Rev. **59**(2), 295–320 (2017)
16. Esposito, F., Matta, I., Ishakian, V.: Slice embedding solutions for distributed service architectures. ACM Comput. Surv. **46**(1), 6:1–6:29 (2013)
17. ETSI White Paper: Network functions virtualisation: an introduction, benefits, enablers, challenges and call for action. Technical report, October 2012
18. Fischer, A., Botero, J.F., Beck, M.T., de Meer, H., Hesselbach, X.: Virtual network embedding: a survey. IEEE Commun. Surv. Tutor. **15**(4), 1888–1906 (2013, Fourth)

19. Foukas, X., Patounas, G., Elmokashfi, A., Marina, M.K.: Network slicing in 5G: survey and challenges. IEEE Commun. Mag. **55**(5), 94–100 (2017)
20. Gouveia, L., Leitner, M., Ruthmair, M.: Layered graph approaches for combinatorial optimization problems. Comput. Oper. Res. **102**, 22–38 (2019)
21. Maniezzo, V.: Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem. INFORMS J. Comput. **11**(4), 358–369 (1999)
22. Mijumbi, R., Serrat, J., Gorricho, J., Bouten, N., De Turck, F., Boutaba, R.: Network function virtualization: state-of-the-art and research challenges. IEEE Commun. Surv. Tutor. **18**(1), 236–262 (2016). Firstquarter
23. NGMN Alliance: 5G White Paper. Technical report, February 2015
24. Ordonez-Lucena, J., Ameigeiras, P., Lopez, D., Ramos-Munoz, J.J., Lorca, J., Folgueira, J.: Network slicing for 5G with SDN/NFV: concepts, architectures, and challenges. IEEE Commun. Mag. **55**(5), 80–87 (2017)
25. Orlowski, S., Wessäly, R., Pióro, M., Tomaszewski, A.: SNDlib 1.0—survivable network design library. Networks **55**(3), 276–286 (2010)
26. Soenen, T., Banerjee, R., Tavernier, W., Colle, D., Pickavet, M.: Demystifying network slicing: from theory to practice. In: 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), pp. 1115–1120, May 2017
27. Stützle, T., Hoos, H.H.: $\mathcal{MAX}-\mathcal{MIN}$ ant system. Fut. Gener. Comput. Syst. **16**(8), 889–914 (2000)
28. CPLEX User's Manual: IBM ILOG CPLEX Optimization Studio (2017)
29. Vassilaras, S., et al.: The algorithmic aspects of network slicing. IEEE Commun. Mag. **55**(8), 112–119 (2017)
30. Yen, J.Y.: An algorithm for finding shortest routes from all source nodes to a given destination in general networks. Q. Appl. Math. **27**(4), 526–530 (1970)