A Lightweight Policy-aware Broker for Multi-domain Network Slice Composition

Xuan-Thuy Dang*, Fikret Sivrikaya†

*DAI-Labor, Technische Universitaet Berlin, Germany
{xuan-thuy.dang, manzoor-ahmed.khan}@dai-labor.de

†GT-ARC gGmbH, Berlin, Germany
fikret.sivrikaya@gt-arc.com

***Abstract*—Network virtualization technologies allow 5G and beyond mobile networks to support new vertical applications of various service types. Network slicing (NS) further provides a concept for multi-tenant access to common network infrastructure spreading from core and transport to radio access networks. Nevertheless, the management and orchestration of network slices across multiple infrastructure and administrative domains have only recently gained attention. In this work, we first present a reference architecture for NS management and its brokering layer that enables NS discovery and dynamic composition. We detail the designs of the main NS brokering components, i.e., NS information model and NS request handling, which address some unique challenges of multi-domain NS brokering. Our approach combines machine learning and logic reasoning for the selection of NSs based on their functional, non-functional attributes and other policy constraints. Finally, we validate the NS request matching approach and observe a high prediction accuracy. We conclude the work with a discussion on possible improvement to the NS matching approach and its extension for cross-domain NS management.**

***Index Terms*—Autonomous Network Management, 5G, Network Slicing**

I. INTRODUCTION

The 5th generation mobile network (5G), with its manyfold improvements in capacity and flexibility, is expected to enable novel and demanding applications such as autonomous driving, eHealth, and smart cities. In supporting such diverse vertical services, the 5G core and the radio network can be virtualized and configured for specific Quality of Service (QoS) requirements of different use cases. This concept of dynamically enabling flexible multi-tenant access to mobile network infrastructure is known as *Network Slicing* (NS). Given the necessary conditions that each network segment (domain) and function can be treated as programmable and composable services, which has become a reality thanks to the flexible network architecture with SDN, NFV technologies, service-oriented network management approaches become a viable option for cross-domain NS. Managing mobile networks as services requires non-functional and qualitative constraints to be considered, i.e., stakeholder's relationships and business objectives, affinity constraints, application awareness, user preferences, etc. These constraints are often overlooked in the current single provider and monolithic mobile network architectures, which results in the prevalence of numerical models and approaches to network management. We argue that numerical approaches are intractable or fail to capture the high-level stakeholder relationships and complex system behaviors. This is evidenced by the common assumptions about the discreteness and the dimensions of the system variables considered by numerical approaches. Arguably, handling the network as set of composed services is required for true autonomous network management. A true cross-domain mobile network orchestration platform is yet to be seen. The policy-based network management system created in 2001 [1] is the most flexible network management model in use, which suffers from the manual rules creation and scalability (number of policy rules in a complex system), not to mention policy conflicts and inconsistency. The last significant contribution towards the autonomous network management concept is the FOCALE control loop concept created in 2006 [2]. Only recently, the concept of intent-based networking started to readdress the service-oriented network management branch. There is a large gap where the achievement of service-oriented computing (SOA, federation, trust, etc) can be applied for multi-domain network management.

In this work, we first provide a background on network slicing and multi-domain slicing architectures in Section II. Section III discusses the architecture layers with relevant details on state of the advancement and related work. We then propose a multi-domain NS architecture with a brokering layer facilitating cross-domain NS composition in Section IV. The main components of the NS broker, i.e., NS information model and NS request handling, are detailed, showing our approach to address most important challenges of NS brokering layer. The approach detailed in Section IV-B, which combines machine learning and logic reasoning, is proposed for the selection of NS based on the semantic description of their attributes and policy constraints. Finally, in Section V we validate the NS request matching algorithm and suggest future improvements.

II. BACKGROUND ON CROSS-DOMAIN NETWORK SLICING

In this section, we first provide a background on network slicing enablers and concepts. There are advanced architectures proposed for flexible intra-domain network slice management and cross-domain network slice orchestration. In multi-domain slicing, we differentiate between federated and brokering models.

A. Network slicing perspectives

There exist several network slicing definitions from different SDOs, which reflect their respective network virtualization paradigms and communication systems. Despite the common objectives of network slicing, the diverse focuses on core enabling technologies, network domains, and architectures result in varying concepts.

In the 3rd Generation Partnership Project's (3GPP) perspective, a network slice is a logical end-to-end network that can be dynamically created. A given User Equipment (UE) may access to multiple slices over the same Access Network. Each slice may serve a particular service type with respective Service-level Agreement (SLA). Network Slice is defined within a Public Land Mobile Network (PLMN) and includes the Core Network Control Plane, User Plane Network Functions, and the 5G access network (AN) [3]. The main concepts of 3GPP NS are:

- Network Function: A processing function in a network, which has 3GPP defined functional behavior and interfaces. A network function can be a physical equipment or software implementation of the network element deployed on virtualization infrastructure.
- Network Slice: A logical network that provides specific network capabilities and network characteristics.
- Network Slice instance: A set of network function instances and their required resources (compute, storage and networking resources) which form a deployed NS.

In the Open Network Foundation's perspective, slicing requires the partitioning and assignment of a set of resources that can be used in an isolated, disjunctive or shared manner [4]. A set of such dedicated resources, e.g., bandwidth on a network link, forwarding tables in a network element (switch, router), processing capacity of servers and network elements, can be called a Slice Instance. ONF slicing is based on SDN concept, in which i) network services are built from network resources and ii) provisioning, management and control of the services and related resources are executed via the controller. The controller works with two major types of resource views: it interacts with its tenants via a tenant specific Client Context and with its resources via a resource specific Server Context. The term virtualisation is used to describe the function of a controller to aggregate and abstract the underlying resources it manages and controls. Views onto such virtualized resources, or resource groups dedicated to particular clients, are provided to tenants (clients, applications, users) via northbound interfaces. Client context is a similar concept to network service in NFV. Server context is a similar concept to NFV NFVI resources.

B. Multi-domain Network Slicing Architectures

We envision future mobile networks, that are composable from multi-provider networks and multi-administrative domain components and require an evolved management architecture. Figure 1 shows the two architectures for multi-administrative domain management, which extend the single network domain management architectures, e.g., the European Telecommunications Standards Institute (ETSI) NFV Management and Orchestration (MANO) [5] and SDN [6] architectures. The bottom *Intra-domain network slicing* layer consists of the current single operator virtualized and slice-able network infrastructure, e.g., C-RAN, virtual mobile network core, SD-WAN. SDN and NFV enable programmable network functions which can be composed and provided as services. The middle *Multi-domain slice management* layer is a facility layer, which allows cross-domain management and integration. Its components address heterogeneity issues of resource management, integration, policy conflicts, security and trust, etc. There are two models for the realization of this layer depending on the level of trust among stakeholders: federation and brokering. In contrast to current federated multi-domain architectures (Figure 1, left), we propose a broker based architecture with a brokering layer and some main functionalities (Figure 1, right). At the top layer *Multi-domain virtual networks*, the mobile virtual network operators (MVNO) make use of the resources as network slices exposed by infrastructure providers to create multi-administrative domain mobile networks.

III. RELATED WORK

In this section, we review the advancement of network slicing enablers and technologies in the intra-domain NS and multi-domain NS management layers of the architecture above. Some legacy service brokering models in service oriented computing domain are reviewed. Based on those models we identify the requirements for network slice brokering, some of which are addressed with the broker based architecture proposed.

A. Intra-domain network slicing Layer

Different management aspects for 3GPP network slices are defined in [7]. Managing a complete Network Slice Instance (NSI) is not only managing all the functionalities but also the resource necessary to support certain set of communications service. An NSI not only contains Network Functions (NFs), e.g, belonging to access network (AN) and core network (CN), but also the connectivity between the NFs. If the NFs are interconnected, the 3GPP management system contains the information relevant to connections between these NFs such as topology of connections, individual link requirements (e.g. QoS attributes), etc. For the part of the Transport Network (TN) supporting connectivity between the NFs, the 3GPP management system provides link requirements to the management system that handles the part of the TN supporting connectivity between the NFs. NSI can be composed of network slice subnets of PNFs or VNFs.

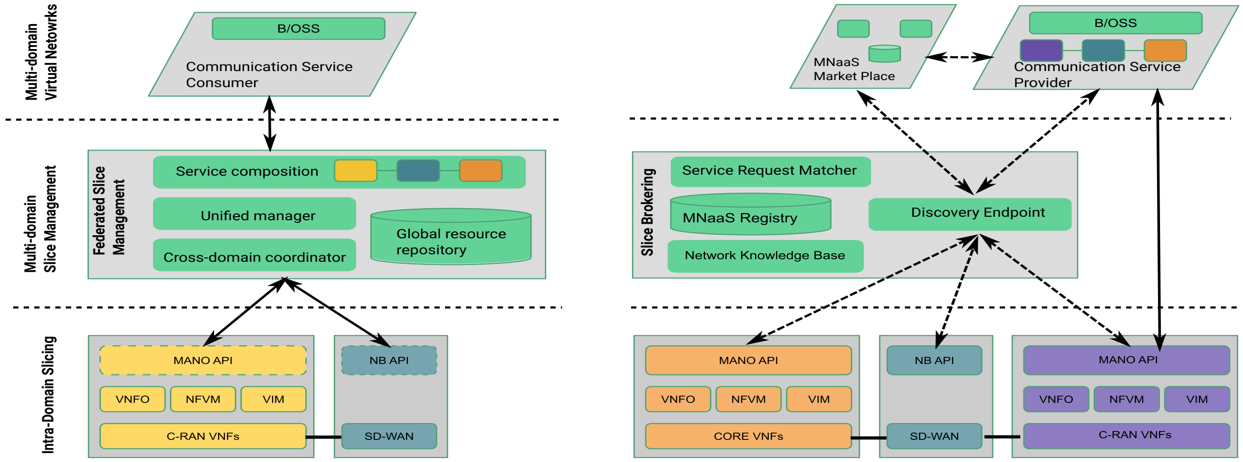


Fig. 1. Multi-domain network slicing management architectures: federated (left), brokering (right). The dotted arrows show stakeholder-neutral communication and the solid arrows show operator specific network protocols

Network slice instance lifecycle management considers the network service instance to be independent from the network slice instance it uses. Typically a network slice instance is designed in preparation phase, instantiated in Instantiation, Configuration and Activation phase, operated in Run Time phase and finally decommissioned Decommissioning phase. 3GPP also defines 3 management functions:

- Communication Service Management Function (CSMF): Responsible for translating the communication service related requirement to network slice related requirements.
- Network Slice Management Function (NSMF): Responsible for management and orchestration of NSI lifecycle. Derive network slice subnet related requirements from network slice related requirements.
- Network Slice Subnet Management Function (NSSMF): Responsible for management and orchestration of NSSI.

In our previous work [8], we proposed a network service overlay framework (NSOF) for an improved end-to-end autonomous intra-domain slice MANO and a realisation of NS as a service. The management components are based on the NFV MANO or SDN architectures, which expose a set of north-bound or management interfaces (MANO API showed in Figure 1) for the integration with B/OSS systems allowing the description of high-level business objectives. The autonomous translation from high-level objectives to lower-level network control can be implemented by the domain MANO with PBNM approaches [9] through management policies written in domain specific languages (DSL) and southbound network control protocols, e.g., SNMP, OpenFlow, etc.

B. Multi-domain Network Slice Management Layer

Multi-domain network slices stretch across administrative domains. Taking advantages of the decoupled and abstracted network infrastructure as services, MVNOs can provide communication services for different types of vertical applications sharing the same mobile network infrastructures. The virtual networks are composed from different RAN, core, and transport network slices. There are two cross-domain management models that facilitate cross-domain slice composition.

1) Cross-domain Federated Slice Management: For the integration of cross-domain infrastructures, a common management layer is required. Depending on the generality of the solutions, several management approaches are implemented in practice. For large MNOs, who own RAN, Core, and transport network infrastructures, a hierarchical management layer for single administrative domain is applied. In a multi-administrative domain network, a federated management system is applied, which consists of common interfaces agreed among the different infrastructure and service providers.

A multi-domain orchestrator is proposed in [10] for the cross-domain handling of slice requests. The orchestrator fulfills a slice request by negotiating resources provided by neighboring network domain. The established cross-domain slices are managed by the multi-domain orchestrator through delegation of management task to the specific domain orchestrators forming a NS management hierarchy. A hierarchical multi-domain orchestration architecture is proposed in [11], which include a cross-domain Resource Broker. The specific domain management layers interact with the broker. However, a unified orchestration and management layer is required for the control of the aggregated multi-domain NSI. The multi-domain NS architecture proposed in [12] also features a unified management layer. The ETSI NFV group specifies a multi-domain management architecture in [13] with cross NFV orchestrator interfaces. Such approach increases the complexity of MANO layer and results in a highly coupled systems.

2) Cross-domain slice Brokering: In contrast to federation models, the service brokering layers are stakeholder neutral and do not include service composition logics. Through NS brokering application interfaces, a MVNO's OSS system looks up appropriate network infrastructures that meet its business objectives and requirements, e.g., URLLC, eMBB, mMTC, cost. Similarly, provider of specialized infrastructure (e.g, Information Centric Edge Network) can combine their networks

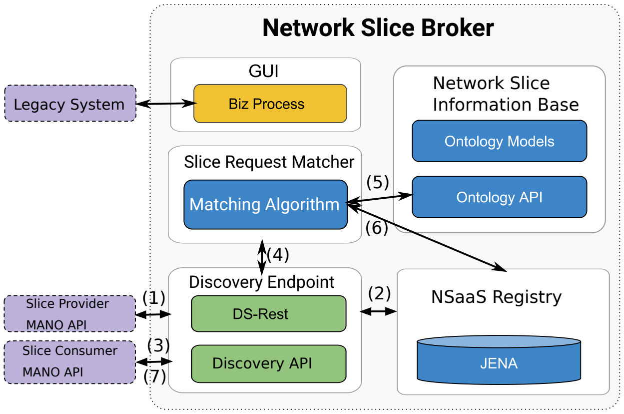


Fig. 2. A brokering architecture for multi-domain slice composition

with other provider's transport and core network to form new network slice through service contracts and autonomous slice configuration without the need for establishing ad-hoc federated systems. New type of players can specialize on providing network service composition and management functions to other communication service providers, e.g., market place infrastructure, network management as a service (NMaaS). Compared to other federated management systems [12], [11], the NMaaS functions is removed from the middle brokering layer and integrated with service discovery functions through generic interfaces. This allows more flexibility in NS composition and orchestration approaches selection, which can be provided by third-party stakeholders.

In [12], a Service Broker Stratum is proposed to handle incoming slice requests from MVNOs, and application service providers. Its functionalities include: NS admission control and service negotiation, multi-tenant interface with a federated multi-domain Service Conductor plane, NSI billing and accounting, NSI life-cycle scheduling. The Service Broker collects abstracted service capability and administrative domain information creating a global service support repository. It also interacts with the B/OSSs in order to collect business, policy and administrative information. Although being designed as a NS meta information store, the Service Broker still involves in the service composition process and is coupled with the federated service composition function.

In the service oriented computing domain, various service brokering architectures were proposed. In [14] a trust-aware brokering layer was proposed for cloud based services. The authors in [15] proposed a brokering layer with game theory based approach for SLA negotiation. A QoS based service brokering approach was proposed in [16] services oriented architecture. Most service brokering architecture presented focus on specific service attributes, which results in the formulation and solving of service selection problem as optimization problems. E.g., QoS-aware composition problem aiming at an optimal execution plan to maximize end-to-end QoS of the composed service can be modeled as a multidimensional, multi-objective, multi-choice knapsack problem (MMMKP) [17]. Most solution approaches model the problem as an ILP, whose computation cost is avoided with heuristic, reduced constraints and objectives. For example, the authors of [18] aim at efficiently placing virtual network functions and deploying service function chains. Their approach considers constraints such as the number of instantiated virtual network functions, physical and virtual resource consumption, and QoS metrics such as end-to-end latencies. In order to create a generic brokering layer for diverse resources in multi-domain networks, the assumptions about service selection criteria must be avoided.

We summarize some important requirements for the broker based multi-domain NS orchestration, which are the focus of this work, as follow:

- *Policy-awareness:* As a general NS broker, the matching process can not be based on specific numerical parameters, e.g., QoS measurement and capacity specification. Additional policy and business goal formulations are also considered, e.g., affinity constraints, high-level quality indication terms. This requirement eliminates common numerical approaches in many intra-domain and federated systems for service composition and embedding.
- *Network slicing abstractions:* Formal models of the infrastructure components and services, their behaviors and policies from the network domain, among others, are required for the specification of architectural differences and complex stakeholder interactions in multi-domain settings.
- *Portability:* The main difference from broker based multi-domain NS orchestration is the lack of ad-hoc protocols and cross management layers, which are agreed among the stakeholders. This requires intra-domain NS management layers to expose meta description of the NSs' functionalities, behaviors and policies while internally translating these policies to intra-domain operations.

IV. AN ARCHITECTURE FOR THE CROSS-DOMAIN NETWORK SLICE BROKER

The federated management models to date are evolution approaches that extends the current monolithic architecture with additional network management functions for multi-domain mobile network [11], [12]. In contrast, on the basis of virtualized mobile network infrastructure with service oriented network slices, the proposed NS brokering layer only provides an abstraction of the underlying infrastructure and generic service discovery functions to facilitate cross-domain network management and orchestration. The main functional blocks of the proposed NS broker depicted in Figure 2 are:

- **Network Slice Information Base** contains the meta information, i.e, the common description of the NS concepts, components, functions, and services, which are fundamental for cross domain inter-operations. Although the concepts are based on standard network information base, e.g., common information model [19], semantic service description, the management system at each network domain may follow different standardized or proprietary schemes. This component provides additional meta information framework and transformation schemes.

- **NSaaS registry** contains descriptions of each network domain's exposed network services. Such descriptions include the functional, service agreement, constraints and other non-functional information provided by the infrastructure providers. The NS descriptions are based on the information representation schemes in the network information base.
- **Discovery Service Endpoints** are application protocols and messages allowing infrastructure, CSPs and consumers to register, update, and lookup the exposed mobile network services. These protocols are stakeholder neutral and based on standard interfaces and protocols (e.g., REST, HTTP). Internally, service discovery functions rely on the other functional blocks.
- **Network Slice Request Matcher** implements general search and matching logics to fulfill the network slice requests. The requests contain SLA, functional, and non-functional parameters that should be matched with the potential network slice.

A typical multi-domain network slice composition facilitated by the NS broker is shown in Figure 2 and described next. A service (e.g., cloud) provider slices her virtual infrastructure, which can be used by various application services, with different QoS, non-functional and functional policies and other attributes listed in Table I. The slice descriptions including those attributes, provisioning methods, management APIs, among others, are registered with the NS broker using post method of the provided Discovery Endpoint (1). The descriptions are optionally transformed based on the NS description schemas defined in the NS Information Base, and stored in the NSaaS Registry (2). A slice consumer (e.g., VNF provider) use the Discovery Endpoint to search for cloud services with a set of requirements, e.g., geo-location, cost, etc, which are contained in slice request messages (3). The request messages are handled by the Slice Request Matcher, which match the required parameters with the available slices in the registry (4). We detail a graph embedding algorithm in the next section, which enables such service mapping based on the providers' objectives (attributes and policies). The matching process makes use of the meta information in the NS Information Base (5) and the NS description in the Registry (6). The resulting set of suitable NSs descriptions are returned to the Slice Consumer (7). The NS description should provide necessary information allowing NS composition to be initiated by the stakeholders.

Following the described NS brokering sequence, we next provide details of the main functional blocks of the proposed NS brokering architecture: the NS information base and NS request matcher.

A. Network Slice Information Base

The NS ontology in Figure 3 illustrates a common model of the NS Information Base (NSIB). The NSIB provides formal abstractions of the physical and virtual network and computing infrastructure in order to enable the mapping and composition of multi-domain NS. The abstraction includes their respective system entity models, policies and goals, attributes, constraints, among others. While various object models (e.g, DMTF's Common Information Model) are widely adopted as abstraction for reactive management in current static B/OSSs, the dynamic and multi-stakeholder interactions envisioned in the multi-domain slice management requires such an abstraction framework that can capture system and stakeholder behaviors. For this purpose, we employ Event Calculus (EC) [20] to create formal system models of each network layer, which support logical reasoning, i.e, deriving low-level policies required to fulfill a high-level goal, or identifying the causes to a conflicting state. The ontology is constructed using a descriptive logic Language, DL-Lite [21], to contain the formal descriptions of different types of network slices, which enables logical reasoning upon both functionalities of the NS and their behaviors specified with EC predicates. Reasoning and querying in DL-Lite is polynomial in data complexity.

QoS	**Functional**	**Non-Functional**	**Other**
Bandwidth	Input	Cost	Timeliness
Latency	Output	Availability	Efficiency
Throughput	Precondition	Responsiveness	Geo-location
Jitter	Effect	Reliability	
Capacity		Security	

TABLE I
NETWORK SLICE ATTRIBUTES

The NSIB serves as a semantic framework for the description of layer and stakeholder specific NSs or as a meta model for the NSaaS Registry as depicted in Figure 2. Terminological foundation (T-Box) of the NSIB ontology is defined with the DL ontology language and EC ontology. The former is a common language to define ontologies for different stakeholder domain models. Thus, it allows the application of reasoning approach on distributed, cross domain knowledge. We extend the Open-Multinet[1] ontology for virtual network infrastructure management with policy concepts to capture each layer's systems and stakeholders' objectives besides NFV concepts similar to [22]. The semantic framework is generic, i.e., existing network and service models of different stakeholders described with domain specific languages (e.g., the OMG's SBVR [23]), can be automatically transformed to NSIB facts. Therefore, existing policy editing tools and models can be reused. As an illustrative example, Table II summarizes the NSIB's atomic concepts and roles, which represents MVNO slice concepts and policies in the NSIB.

B. Network Slice Request Matching

In this section, we detail on an interest matching algorithm for the selection of NS (descriptions) based on the QoS and other policies contained in the NS request. Due to the aforementioned attribute generality requirements, logical

---

[1] http://w3c.github.io/omn/

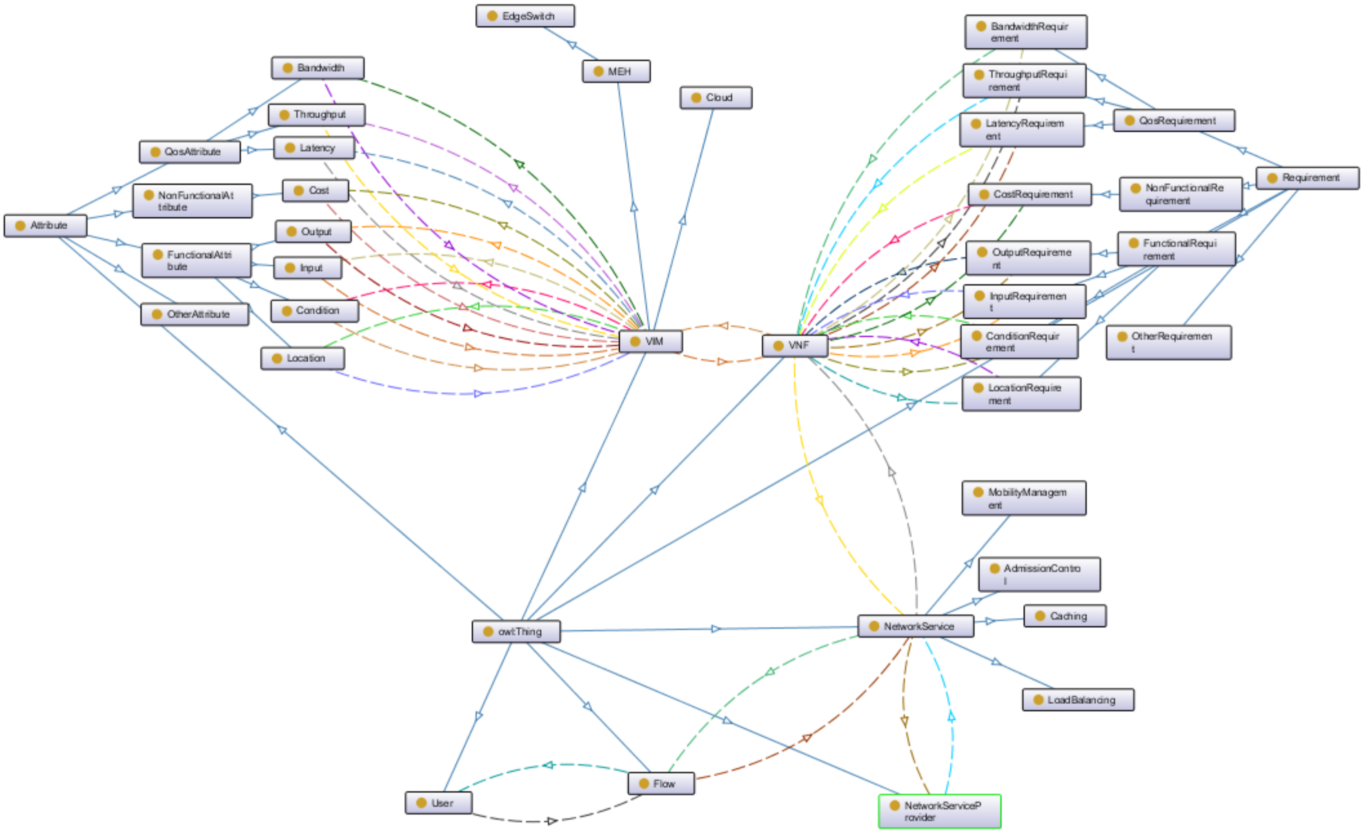


Fig. 3. Network knowledge base ontology for network services and policies description

TABLE II
DESCRIPTION OF A MVNO POLICY WITH DL-LITE ONTOLOGY: PROVISIONING FOR USER u'S FLOW f WITH QOS CLASS c AND NETWORK STATISTIC s.

MVNO Ontology	Predicate	Description
Concepts	MVNO, Flow, QoSClass, NetworkStatistic, NetworkParameter[Bandwidth, Delay, Jitter]	Constants representing object in the system or their inheritences, i.e, in square brackets
Role Concepts	Provisioning(User(u), QoSClass(c)), HasFlow[Create,Remove](User(u), Flow(f)), Metering(Flow(f), NetworkStatistic(s))	Basic role, or their inheritances, associated with a concept representing relationship between two objects
Roles	hasQoSClass(Flow(f), QoSClass(c)), hasNetworkParam(QoSClass(c), NetworkParameter(p))	Simple roles representing relationship between object instances
Attributes	measuredBandwith(Flow(f), 10mbps), delayParameter(QoSClass(c), 1ms), creationTime(Create(User(u), Flow(f)), Timestamp(ts))	Attributes of a concepts or role concepts, which describe the state and invariable of concepts
Operations	provisioned(MVNO(op), Provisioning(p)), createFlow(User(u), Create(c)), removeFlow(User(u), Create(c)), measured(Statistic(s), Metering(m))	Object methods resulting in the addition, modification and removal of system states

reasoning approach is used for categorizing NSs based on their attributes. Because first order logic ontology is applied for the NS description, logical reasoners and rules are used to infer the types and suitability of the NSs according to each attribute type. However, the reasoning process can be compute-intensive with large NS Registry and number of attribute types. This leads to high response time for NS request processing.

We apply a simple machine learning approach on the ontology graph showed in Figure 3, which represents the relationships among the NS instances, their attributes and types entities. The entities, their attributes, and their relationships are represented as *triples* of concepts and predicates in the NSIB, e.g., $< VM1, has\_cost, high >$. The learning model is trained to learn the associations of a NS instance (e.g., a virtual machine) with a group of attributes parameters. The model is then used to predict a NS using the attributes specified in the NS request (e.g., a VNF specification). We adopt an energy based model (EBM) [24] to embed the entities and relationships to a continuous vector space $\mathbb{R}^n$ (latent space with n dimension) so that for each triple $< E_i, R, E_j >$ in the NSIB exist three vectors (embedding points) e_i, r, e_j in $\mathbb{R}^n$ and e_i, e_j are close (i.e., in vector distance) to each other in relation with r. To learn the embedding, we use a margin loss function (hinge loss):

$$\begin{aligned} &L(< E_i, R, E_j >) = \\ &max(0, m + E(< E_i, R, E_j >) - E(< \overline{E_i}, R, \overline{E_j} >)) \end{aligned} \tag{1}$$

where E is the energy function of a triple, $< \overline{E_i}, R, \overline{E_j} >$ is a negative triple (false fact), which is generated from a triple in the NSIB by replacing one of the entities (head or tail) with a random entity, m is a positive margin between the two energy values. We use the energy function defined in [25] as either the L_1 or the L_2-norm of the triple's embedding as follows:

$$E(e_i, r, e_j) = \|e_i + r - e_j\|_{L_1/L_2} \tag{2}$$

The model is trained by minimizing the margin loss over all embedding of the positive and negative triples in the KB:

$$L = \sum_{<E_i,R,E_j>\in\mathcal{P}} \sum_{<\overline{E_i},R,\overline{E_j}>\in\mathcal{O}} [m + E(e_i, r, e_j) - E(\overline{e_i}, r, \overline{e_j})]_+ \tag{3}$$

where, $\overline{e_i}$ and $\overline{e_j}$ are the embedding of $\overline{E_i}$ and $\overline{E_j}$, $\mathcal{P}$ is the set of (positive) triples in KB, $\mathcal{O}$ is the set of negative triples generated, and $[x]_+ = max(0, x)$. Stochastic gradient descent in mini-batch is used to train the model which results in a loss function with low energy for positive embedding and high energy for negative ones. More details of the model construction (named TransE) is provided in [25].

V. EVALUATION OF THE SERVICE REQUEST BROKERING

We experiment with the autonomous placement of the network services provided by multiple providers on virtualization infrastructures. One example of the network service is the mobility management service (realized in [26]), which consists of a chain of flow handling functions to be deployed on forwarding switches. Various QoS constraints and other attributes (specified in the embedding policies) as in Table I are used as the requirements of the network service providers' (NSP) network services and as the capacity and feature of VIM resources.

The EBM is trained with the following parameters: the embedding vector dimension n is set to 100, the margin value m is variable, the gradient descent learning rate is 0.01, the distant function is L1, and the mini-batch size is 4800. The NSIB consists of 26 classes and its datasets are constructed as following:

- Ontology with no inverse relations: 21 relations, 434 entities, 3030 triples (e.g., $< VM1, has\_cost, high >$).
- Ontology with some inverse relations: 26 relations, 481 entities, 4165 triples (e.g., $< VM1, host\_vnf, VNF2 >$ and $< VNF2, is\_deployed, VM1 >$).
- Ontology with all inverse relations: 42 relations, 341 entities, 4021 triples.

Figure 4 shows the impacts of the inverse relationships proportion in the NSIB on the learning results. The false negative triples (the generation of head or tail entity of a triple in the NSIB with a random one results in a triple that already exists) are filtered out. We observe that the accuracy increases with the increasing proportion of inverse relationships for both hit10 rate (ratio of correctly predicted entities in top 10 predicted triple) and mean rank value (energy of correct triple). In contrast, no significant gain is achieved with the addition of inferred relationships in the NSIB (semantic smoothing). This can be explained as the inverse relationships help in balancing the heads and tails prediction. The NSIB without inverse relationships contains large number of triples with the same head, i.e., a VIM node has multiple attributes.

From the previous observation we focus on the learning results for NSIB with full inverse relationships. Figure 5 shows the impact of false negative triples on the learning accuracy. It can be clearly observed that the presence of false negative triples in the dataset reduces the accuracy. For all the NSIB models, we observe an optimal margin value $m = 7$.

The highest prediction accuracy (hit10) observed is 80% for this data set of around 4000 triples. The result depends greatly on the structure of NSIB graph, which can be enforced with reasoning approaches. By combining computing intensive reasoning process with machine learning, the impact on response time can be shifted forward to training phase and higher accuracy can be achieved in testing phase. More robust learning models will be the subject of our future works. Subsequently, the predicted results can be used in numerical optimization approaches, e.g., ILP, as a refinement step.

VI. CONCLUSION

We presented a reference architecture for NS management and its middle brokering layer that enables NS discovery and dynamic composition. We detailed the designs of the main NS brokering components, i.e., NS information model and NS request handling, which address some unique challenges of multi-domain NS brokering. Our approach combines an energy based machine learning model and logic reasoning for the selection of NSs based on their attributes and policy constraints. The approach shows high prediction accuracy. We concluded the work with a discussion on possible improvement to the NS matching approach and its extension with numerical optimization technique for cross-domain NSs management.

Acknowledgements: This work was supported in part by the German Federal Ministry of Transport and Digital Infrastructure under grant reference number 16AVF1021A and by the German Federal Ministry of Education and Research under grant reference number 01IS16045. We thank Onat Tanriover for his technical support.

REFERENCES

[1] R. Rajan, D. Verma, S. Kamat, E. Felstaine, and S. Herzog, "A policy framework for integrated and differentiated services in the internet," *IEEE network*, vol. 13, no. 5, pp. 36–41, 1999.

[2] J. Strassner, "Knowledge representation, processing, and governance in the focale autonomic architecture," *Autonomic Network Management Principles: From Concepts to Applications*, p. 253, 2010.

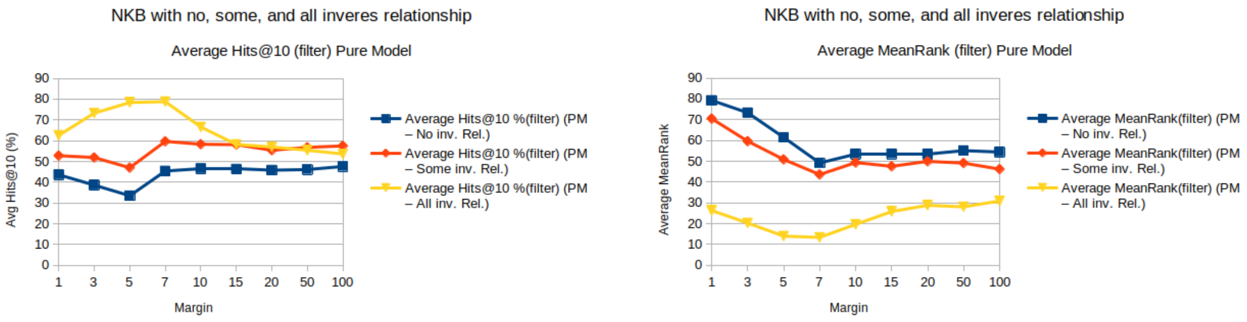
[3] 3GPP, "System architecture for the 5G System (5GS)," 3rd Generation Partnership Project (3GPP), http://www.3gpp.org/, Tech. Rep., 6 2019.

[4] ONF, "Applying SDN architecture to 5G slicing," Open Networking Foundation (ONF), https://www.opennetworking.org/wp-content/uploads/2014/10/Applying_SDN_Architecture_to_5G_Slicing_TR-526.pdf, Tech. Rep., 2016, accessed: 2019-08-28.

[5] ETSI GS NFV-MAN 001 V1.1.1 (2014-12), "Nfv, network functions virtualisation; management and orchestration," 2014.

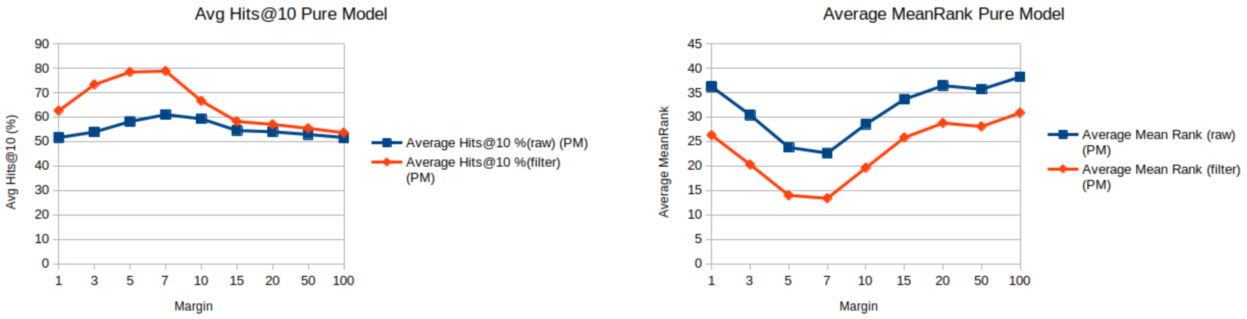
[6] "SDN architecture overview," , accessed: 2014-11-28.

[7] 3GPP, "Telecommunication management; Study on management and orchestration of network slicing for next generation network," 3rd Generation Partnership Project (3GPP), http://www.3gpp.org/, Tech. Rep., 2018.



(a) Hit@10 of pure NSIB (b) Mean rank of pure NSIB

Fig. 4. Impact of the inverse relationship in NSIB on learning results



(a) Hit10 of pure NSIB (b) Mean rank of pure NSIB

Fig. 5. Impact of false positive triples on learning results

[8] X.-T. Dang, M. A. Khan, and F. Sivrikaya, “An autonomous service-oriented orchestration framework for software defined mobile networks,” *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, pp. 277–284, 2019.

[9] A. Westerinen, J. Schnizlein, J. Strassner, M. Scherling, B. Quinn, S. Herzog, A. Huynh, M. Carlson, J. Perry, and S. Waldbusser, “Terminology for Policy-Based Management,” RFC 3198 (Informational), Internet Engineering Task Force, Nov. 2001, accessed: 2017-03-21. [Online]. Available: http://www.ietf.org/rfc/rfc3198.txt

[10] R. Guerzoni, I. Vaishnavi, D. Perez Caparros, A. Galis, F. Tusa, P. Monti, A. Sganbelluri, G. Biczók, B. Sonkoly, L. Toka *et al.*, “Analysis of end-to-end multi-domain management and orchestration frameworks for software defined infrastructures: an architectural survey,” *Transactions on Emerging Telecommunications Technologies*, vol. 28, no. 4, p. e3103, 2017.

[11] V. Sciancalepore, C. Mannweiler, F. Z. Yousaf, P. Serrano, M. Gramaglia, J. Bradford, and I. Labrador Pavón, “A future-proof architecture for management and orchestration of multi-domain nextgen networks,” *IEEE Access*, vol. 7, pp. 79 216–79 232, 2019.

[12] T. Taleb, I. Afolabi, K. Samdanis, and F. Z. Yousaf, “On multi-domain network slicing orchestration architecture and federated resource control,” *IEEE Network*, pp. 1–11, 2019.

[13] G. ETSI, “Network functions virtualisation (nfv) release 3; management and orchestration; report on architecture options to support multiple administrative domains,” Technical report, ETSI, Tech. Rep., 2018.

[14] X. Li, H. Ma, F. Zhou, and W. Yao, “T-broker: A trust-aware service brokering scheme for multiple cloud collaborative services,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1402–1415, 2015.

[15] B. K. Ray, S. Khatua, and S. Roy, “Negotiation based service brokering using game theory,” in *2014 Applications and Innovations in Mobile Computing (AIMoC)*. IEEE, 2014, pp. 1–8.

[16] D. A. Menascé and V. Dubey, “Utility-based qos brokering in service oriented architectures,” in *IEEE International Conference on Web Services (ICWS 2007)*. IEEE, 2007, pp. 422–430.

[17] A. Strunk, “Qos-aware service composition: A survey,” in *2010 Eighth IEEE European Conference on Web Services*, Dec 2010, pp. 67–74.

[18] M. C. Luizelli, L. R. Bays, L. S. Buriol, M. P. Barcellos, and L. P. Gaspary, “Piecing together the nfv provisioning puzzle: Efficient placement and chaining of virtual network functions,” in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, May 2015, pp. 98–106.

[19] J. Strassner, “Knowledge management issues for autonomic systems,” in *16th International Workshop on Database and Expert Systems Applications (DEXA’05)*, Aug 2005, pp. 398–402.

[20] A. K. Bandara, E. C. Lupu, and A. Russo, “Using Event Calculus to formalize policy specification and analysis,” *in Proc. of the 4th IEEE Workshop on Policies for Distributed Systems and Networks, POLICY*, pp. 1–14, 2003.

[21] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati, “Tractable reasoning and efficient query answering in description logics: The dl-lite family,” *Journal of Automated reasoning*, vol. 39, no. 3, pp. 385–429, 2007.

[22] I. Oliver, S. Panda, K. Wang, and A. Kalliola, “Modelling nfv concepts with ontologies,” in *2018 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Feb 2018, pp. 1–7.

[23] N. A. Manaf, A. Antoniades, and S. Moschoyiannis, “Sbvr2alloy: an sbvr to alloy compiler,” in *Service-Oriented Computing and Applications (SOCA), 2017 IEEE 10th International Conference on*. IEEE, 2017, pp. 73–80.

[24] Y. Lecun, S. Chopra, and R. Hadsell, *A tutorial on energy-based learning*, 01 2006.

[25] A. Bordes, N. Usunier, A. García-Durán, J. Weston, and O. Yakhnenko, “Translating embeddings for modeling multi-relational data,” in *NIPS*, 2013.

[26] X. T. Dang, M. A. Khan, S. Peters, and T. Dorsch, “Realization of handover management in sdnized 3gpp architecture with protocol independent forwarding,” in *2018 Wireless Days (WD)*, April 2018, pp. 60–67.