Contradiction Management in Intent-driven Cognitive Autonomous RAN

Anubhab Banerjee*+, Stephen S. Mwanje*, and Georg Carle+

*Nokia Bell Labs, Munich, Germany
+Dept. of Informatics, Technical University of Munich, Germany
Email:{*anubhab.1.banerjee, stephen.mwanje*}@*nokia.com, carle@net.in.tum.de*

***Abstract*—Intent Based Networks (IBN) is a prominent feature in the design of the AI-enabled B5G networks. Intents are primarily used to transform the intention of a human operator into network configuration, operation, and maintenance strategies. Although IBN provides future network automation technologies, it also raises the risk of contradiction(s) in an intent which arises during the runtime and cannot be predicted or resolved beforehand. In this paper we propose a new design which helps to detect and remove contradiction(s) in the optimal way during the runtime and evaluate it. We evaluate our proposed solution in a simulation environment and also provide a brief overview of standardization impact of our work to show that it conforms with the worldwide mobile network standardization efforts.**

***Index Terms*—Network management, IBN, SON**

I. Introduction

Intent based networking (IBN) is a key tool for network and service management for next generation networks. From a human mobile network operator's (MNO) perspective, an intent expresses the operator's expectation from the network and service management. The goals and expected behaviors of an autonomous network system can be clearly defined with the help of one or multiple intents. Ideally an intent is described declaratively, i.e., as one or multiple utility level goals that describe the properties of an outcome rather than defining ways to achieve the goals [1]. It is up to the IBN system to explore various methods so that the goals can be achieved efficiently.

Intents are gaining popularity in the field of network and service management because it makes the job of MNO easier. In radio access network (RAN), intents have already been proposed to be used for slicing and resource management [1]. However, no focus was given on managing radio parameters like cell transmit power (TXP), remote electrical tilt (RET), etc. Currently these parameters are managed through a rule-based system called self-organizing networks (SON). Introduction of artificial intelligence has replaced rule-based SON with cognitive autonomous networks (CAN) [2]. Although CAN provides significant improvement over SON in terms of operational complexity and maintenance, it works in a closed-loop way without providing any interface for interaction with MNO.

Earlier in [3] we provided an intent based interface for MNO to use CAN for managing RAN control parameters. Although the design serves its purpose, the major drawback with the design is that it cannot detect contradictions within an intent. Contradictions in an intent arise when, each goal specified within an intent, requires changes in the same parameter but by different amounts. To overcome this problem, in this paper we propose a design to detect and remove contradictions from an intent to ensure uninterrupted executions of intents. Along with the end-to-end system design and evaluation of our proposed solution, we also discuss about the impacts of our work on standardization efforts in this paper. Apart from that, abbreviations used in this paper are listed in Table I.

978-3-903176-48-5 © 2022 IFIP

II. Background: intent-driven CAN orchestration

RAN contains several adjustable network control parameters (NCPs) (e.g., TXP) and key performance indicators (KPIs) (e.g., radio link failures (RLF)). A KPI value can be affected when a certain group of NCPs are changed, e.g., RLF can be affected by changing TXP. Also, changing a single NCP might affect multiple KPIs, e.g., changing TXP can affect both RLF and load. In [4] authors depicted the interconnection among RAN control parameters and KPIs as shown in Fig. 1. We see from Fig. 1 that a single NCP has influence over multiple KPIs and simultaneously a KPI is influenced by multiple NCPs as well. This type of inter-connected relationships among NCPs and KPIs make the RAN management a challenging task. Since all KPIs are crucial in determining the quality-of-service (QoS) of the network, at any time MNO might need to fulfill several KPI related targets simultaneously. To achieve so, MNO needs to have a complete knowledge of the inter-dependencies among the NCPs and KPIs, including the degree of change required in a certain NCP to obtain the required target KPI value.

The job of MNO becomes easier if there is a layer of abstraction between MNO and RAN where MNO can specify KPI related targets and the layer of abstraction generates the appropriate actions to achieve those targets. In [3] we proposed such a design which consists of three components: (i) CAN, (ii) intent specification platform (ISP) and (iii) intent-driven network automation function orchestrator (IDNAFO).

A. CAN

Each KPI is managed by a closed-loop control-based function, called cognitive function (CF). A CF has the responsibility to learn the dependency and variation of the managed KPI on the NCPs [2]. The NCPs, required to adjust the managed

TABLE I: List of Abbreviations used in this paper

Abbreviation	Full name	Abbreviation	Full name
CAN	Cognitive Autonomous Networks	CCO	Coverage and Capacity Optimization
CF	Cognitive Function	CIO	Cell Individual Offset
ETSI	European Telecommunications Standard Institute	IBN	Inten Based Networking
ICDR	Intent Contradiction Detector and Remover	ICP	Input Control Parameters
IDMS	Intent-Driven Management Services	IDN	Intent-Driven Networking
IDNAFO	Intent-Driven Network Automation Function Orchestrator	ISP	Intent Specification Platform
ITU	International Telecommunication Union	KPI	Key Performance Indicator
MLB	Mobility Load Balancing	MNO	Mobile Network Operator
MRO	Mobility Robustness Optimization	NCP	Network Control Parameter
NFV	Network Function Virtualization	ONF	Open Networking Foundation
RAN	Radio Access Network	RET	Remote Electrical Tilt
RLF	Radio Link Failures	SON	Self-Organizing Networks
TTT	Time-To-Trigger	TXP	cell Transmit Power

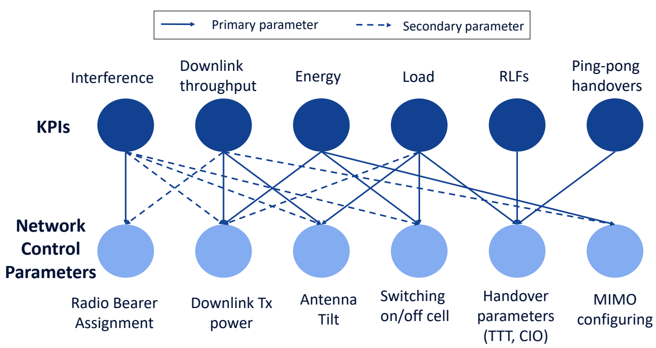

Fig. 1: Interconnection among RAN NCPs and KPIs

KPI, are called the input control parameters (ICPs) and the managed KPI is denoted as output of the CF. A CF always observes and learns how its output changes when ICPs are changed and based on the learning, a CF determines the values of the ICPs which optimize the CF's output in a certain network state [5]. CAN also has a Controller whose main responsibility is to resolve conflicts among the CFs [2], [5]. The idea of a Controller already exists in SON [6] and in [2], [5] we explained and justified its necessity. When a CF determines new optimal values of ICPs, it informs the Controller and the Controller takes the interests of all CFs into account while recalculating an ICP value [7]. We can abstract a CAN as shown in Fig. 2: F_1 and F_2 are two CFs with a shared ICP p. OCRS (contains the set of preferred values) and UF (a function that conveys the goodness of a certain value) are two metrics using which a CF can convey its preferences over an ICP to the Controller [5], [7].

From this overview, we see that CAN operates in a closed-loop manner and does not provide any interface for interaction with MNO for KPI or NCP customization. To overcome this problem, we propose ISP and IDNAFO which work as an interface between CAN and MNO.

B. ISP

As per definition, an intent can be any valid sentence in English language or a set of targets expressed in a structured manner. If the intent is a sentence in English language, it is necessary to parse and convert the intent to a predefined structure which is understandable by different network components. This intent, machine legible by different network components, is called a formal intent [3]. Main functionality of ISP is to convert an intent from MNO into a formal intent. Presence of ISP is optional and entirely depends on MNO, e.g., if MNO inputs formal intent, ISP is no longer required.

C. IDNAFO

IDNAFO takes a formal intent and generates appropriate actions to be executed by CAN. Workflow of IDNAFO consists of three sequential steps, and each step is executed by a separate block as described below:

- After receiving a formal intent, IDNAFO checks if the intent can be executed by CAN. This task is accomplished by Intent Identifier or II. For example, an intent like "switch off cell X" is beyond the operational capability of CAN. If II finds an intent to be out of scope, depending on the system design it may forward the intent to some other entities or back to MNO [3]. Otherwise, it proceeds to the next step.
- In the second step, the intent is classified based on its content (NCPs and/or KPIs). This is done by Intent Classifier or IC.
- At the last step, based on the classification, IDNAFO generates action commands for the Controller and/or the CFs and instructs them to execute the tasks. This is done by Intent Decision Maker or IDM.

Each IDNAFO functionality is shown in Fig. 3, and implementation details can be found in [3].

III. Problem Statement

Although we see that IDNAFO with CAN provide an end-to-end design to execute MNO intents, it does not detect the contradictions arising from the same intent. To elaborate the concept of contradiction, let us consider the following intent: "Increase handover success to x_1% and reduce load by y_1%". Mobility robustness optimization (MRO) and mobility load balancing (MLB) are two CFs responsible for managing the

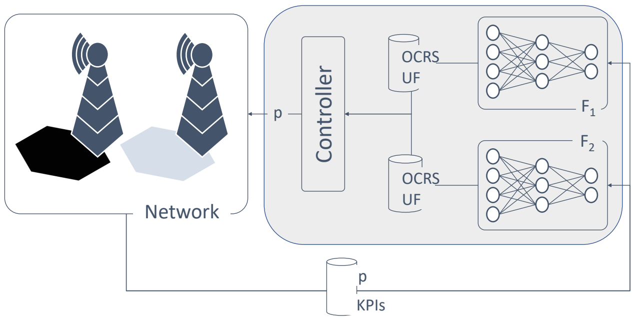

Fig. 2: CAN abstraction with CFs and the Controller

KPIs successful handovers and load. Both MLB and MRO have two shared ICPs: time-to-trigger (TTT) and cell individual offset (CIO). To achieve $x_1\%$ handover success and $y_1\%$ load reduction, let us assume that MRO and MLB propose t_1 and t_2 respectively as required TTT values. If $t_1 \neq t_2$, it gives rise to a contradiction in TTT value. Similarly, if MLB and MRO propose different CIO values, that gives rise to a contradiction in CIO value as well.

Now, it is not possible for MNO or IDNAFO to know beforehand what TTT or CIO values MLB and MRO will suggest and if those values will be equal. As already mentioned earlier, depending on the network state, these TTT or CIO values are generated by the CFs based on their learning and unknown to any other entity. Since IDNAFO is not capable of detecting or removing this kind of contradiction within an intent, any intent with multiple KPIs can potentially give rise to contradictions which are not visible until execution. To circumvent the contradictions, it is recommended to have a system or solution that enables MNO to visualize potential contradictions before the intent is executed.

IV. Proposed solution and evaluation

To overcome the aforementioned problem, we propose a new functionality called Intent Contradiction Detector and Remover (ICDR) which can be implemented separately or within the Controller. Functionality of ICDR is two-fold:

- Contradiction detection, i.e., it notifies MNO if an intent contains any type of potential contradiction.
- Contradiction removal, i.e., ICDR also notifies MNO about possible alternatives to circumvent the contradiction.

Our proposed design consists of two parts: (i) a graphical user interface (GUI) or similar interface that interacts with MNO, and, (ii) ICDR that interacts with CAN. The interface provides two options (Fig. 3):

- Option 1 (”GET OVERVIEW”): if MNO is unsure if the intent is contradiction free, it is always recommended to use this option. If the intent is contradiction free, via the GUI display ICDR informs MNO that the intent can be executed as it is. If ICDR detects contradiction(s) in the intent, it proposes a few possible alternatives to MNO to circumvent the contradiction.
- Option 2 (”EXECUTE”): if MNO is sure that the intent is contradiction free, it is recommended to select this option. Usually intent that deals with a single NCP or KPI can be safely selected for this.

A. Contradiction detection

As soon as ICDR is triggered, it separates the NCPs from the KPIs and requests the CFs, whose outputs are those KPIs, to propose configurations so that each individual KPI target can be achieved. Following the previous example, ”increase handover success to $x_1\%$ and reduce load by $y_1\%$”, ICDR identifies the KPIs (handover success, load) and requests the CFs managing those KPIs (MRO, MLB) to propose configurations (values of TTT and CIO) so that the targets can be achieved. Now it might be the case that both targets cannot be achieved simultaneously, for example, MLB finds that in the current network state, load can be reduced by $y_1'\%$ at maximum, $y_1' < y_1$. In that case, MLB always sends the TTT and CIO values for which load reduction is $y_1'\%$.

Now, for the sake of explanation, let us assume that MLB proposes (t_l, c_l) and MRO proposes (t_r, c_r) as required (TTT, CIO) values to achieve the targets specified in the intent. There are four possibilities:

- Possibility 1 (P1): $t_l = t_r$ and $c_l = c_r$.
- Possibility 2 (P2): $t_l = t_r$ and $c_l \neq c_r$.
- Possibility 3 (P3): $t_l \neq t_r$ and $c_l = c_r$.
- Possibility 4 (P4): $t_l \neq t_r$ and $c_l \neq c_r$.

Out of these four possibilities, only in P1 there is no contradiction, in P2 and P3 there is one contradiction each and in P4 there are two contradictions.

B. Contradiction removal

For contradiction removal, in ICDR we use a Nash Social Welfare Function (NSWF) [8] based approach. In [9], it has been mathematically proven that in a traditional bargaining scenario, NSWF provides the ‘optimal’ solution; here ‘optimal’ signifies that the solution is optimal for all the parties involved in the bargain. Since in CAN all the CFs are given equal priorities, NSWF is therefore used to remove contradictions and find values which are optimal for the combined interests of all the CFs.

In case of a contradiction, ICDR requests all the CFs involved (following the same example, MLB and MRO) to send their UFs over the conflicting NCP(s) (in P2, conflicting NCP is CIO, in P3 it is TTT and in P4, both TTT and CIO). After obtaining the UFs, ICDR finds the conflicting NCP value for which the product of the UFs is maximum, because, that is how NSWF optimal is calculated [8]. To find the NSWF solution, for each configuration value, the Controller calculates the product of the UFs and selects the one for which the product is maximum. To put it mathematically, if p is a shared configuration among CFs: $F_1, F_2, \ldots, F_n$, UFs of the CFs are: $f_1(p)$, $f_2(p)$, . . . , $f_n(p)$ and $\{p_1, p_2, ..., p_m\}$ are acceptable values of p, then p_i is an NSWF solution provided

$$\prod_{r=1}^{n} f_r(p_i) > \prod_{r=1}^{n} f_r(p_j) \tag{1}$$

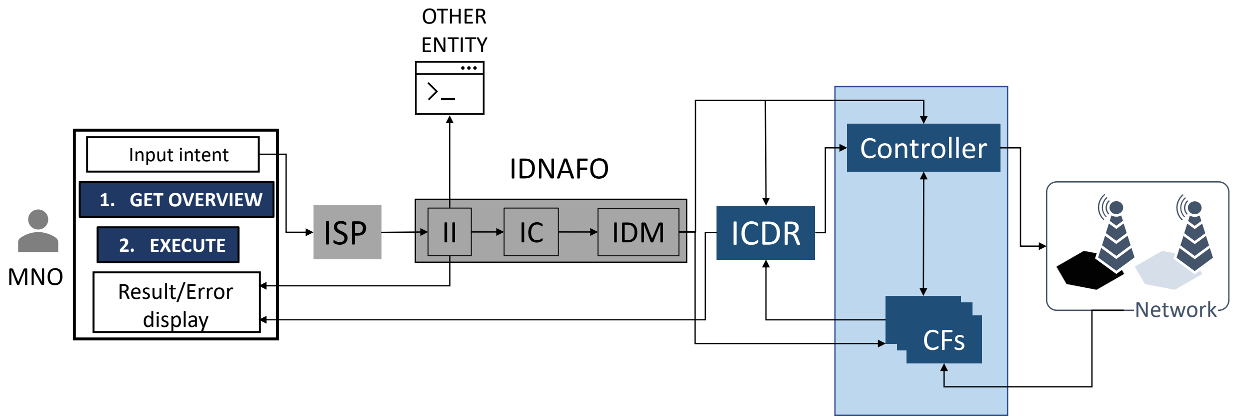

Fig. 3: End-to-end architectural design of our proposed solution

Option	CIO	Load reduction	Handover success increase
1	c_l	y_1	x_2
2	c_r	y_2	x_1
3	c_n	y_n	x_n

Fig. 4: Suggestions to MNO from ICDR as alternatives

$\forall j \in [1, m]$, $j \neq i$.

Let us elaborate further with an example. Let us assume that ICDR finds a contradiction in the intent and it is the case P2, i.e., there is a contradiction over CIO. ICDR asks both MLB and MRO to send their UFs, calculates NSWF and finds c_n to be the NSWF value of CIO. Then ICDR asks MLB to send the load values corresponding to c_n, c_r and asks MRO to send the handover success values corresponding to c_n, c_l. Then ICDR puts everything in a table as shown in Fig. 4 and send them to MNO to choose from, highlighting the recommended option. Targets specified in the intent are highlighted in the first two rows, and ICDR recommended option is listed in the last row.

C. *End-to-end workflow*

After MNO inputs an intent, the intent is sent to ISP and is converted into a formal intent. The formal intent is then forwarded to II by ISP to check if the intent can be executed by CAN. If II finds the intent to be invalid, it sends the intent back to MNO or to other networking entities, based on the implementation. Otherwise, II forwards the intent to IC, which classifies the intent based on its content and separates the control parameters and KPIs. These are then forwarded to IDM, which identifies the CFs managing the KPIs and forwards the control parameter and KPI related instructions to the Controller and those CFs respectively.

If MNO chooses ”EXECUTE” while inputting the intent, IDM asks the Controller to execute the parameter related actions. IDM also asks the CFs to send the necessary configurations to the Controller which are required to execute the KPI related actions and IDM asks the Controller to change the configurations proposed by the CFs. In case of a conflict, the Controller resolves it using NSWF [3].

If MNO chooses ”GET OVERVIEW”, IDM sends the parameter related instructions to ICDR. IDM also asks the CFs to send the necessary configurations to the ICDR which are required to execute the KPI related actions. After ICDR receives the configurations from the CFs, it checks for potential contradictions (overlaps) in those suggested configurations. If no contradiction is found, ICDR informs MNO that the intent is contradiction-free and ready for execution. If ICDR finds contradiction, it calculates the possible alternatives as discussed in Section IV-B and sends these alternatives to MNO.

End-to-end workflow of our proposed solution (ICDR combined with IDNAFO) is shown in Fig. 5.

D. *Experimental Set-up*

To evaluate ICDR, we implement the end-to-end system in a simulation environment. We already discussed implementation of CAN and IDNAFO in our earlier works, [7] and [3] respectively, and they are not reproduced here. Implementation of ICDR is pretty straightforward and is done using Python 3.7. Since the performance of ICDR is independent of how IDNAFO takes the intent as its input, we did not implement ISP and used formal intent as input.

Most challenging part of the implementation is creating an intent database for RAN (IDB-R) since there is currently no such database is publicly available. We created an IDB-R consisting of 30 formal intents manually by ourselves, while making sure that each intent contains at least 1 contradiction. Theoretically, any finite number of contradictions can exist in an intent, however, in IDB-R, maximum number of contradictions in a single intent is currently 4. However, since from Eq. 1 we see that contradiction removal process of ICDR is

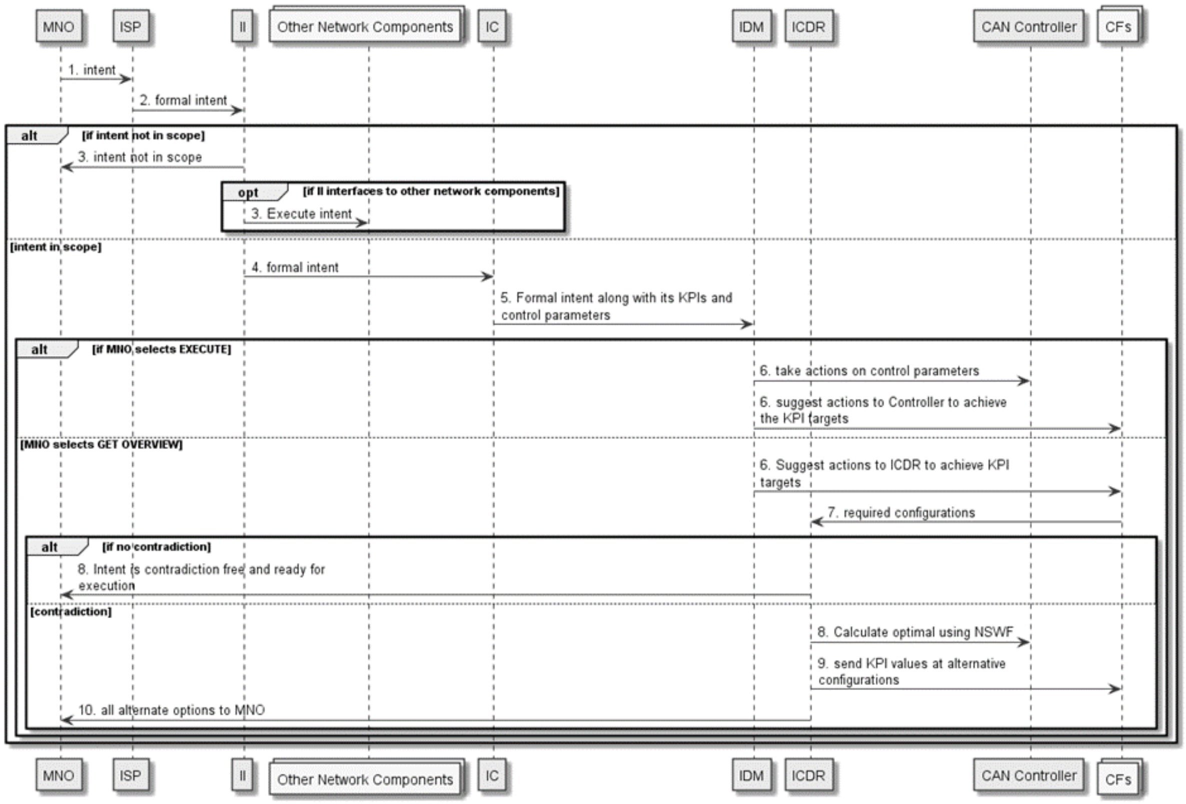

Fig. 5: Flowchart of our proposed solution

independent of the number of contradictions, this value does not affect our evaluation framework. On an average, an intent in IDB-R contains 3 contradictions.

E. Evaluation

In Section IV-B we already mathematically established that when NSWF is used, contradiction removal process is optimal. So, instead of evaluating the optimality of our proposed solution, in this Section use the following three metrics for performance evaluation of our proposed framework:

(i) *Success in contradiction detection*: as the name implies, this measures the success rate of ICDR in detecting a potential contradiction in an intent. For our proposed solution, the success rate 100%, i.e., ICDR is able to detect all the 90 contradictions.

(ii) *Success in contradiction removal*: this measures the success rate of ICDR in removing a potential contradiction using NSWF. For our proposed solution, this success rate is also 100%, i.e., ICDR is able to remove all the 90 contradictions using NSWF.

(iii) *Time*: this accounts for the total time taken to detect and resolve a contradiction and is measured in ms. For our proposed solution, we used a quadcore Intel Core i5-10310U@1.7 GHz CPU, where this value varies between 0.012 - 0.019 ms.

From these three metrics, we see that our proposed solution is efficient for use in real time network management.

V. Related works

A. Intent based resource allocation in RAN

There exist a lot of research works which study intent based slicing mechanisms in RAN [1], [10]. European Telecommunications Standards Institute (ETSI) proposed an open source orchestrator for slicing networks based on NFV specifications [11]. The Linux Foundation also provides ONAP - an open source E2E network orchestrator of virtual appliances [12]. Apart from the orchestrators, [13] described the network slicing framework and the basic design challenges faced for performing RAN slicing in 5G. In [14] the researchers proposed another such management and orchestration framework for slicing of next generation RAN.

Among the existing research works, FlexRAN [15] comes closest to the work done in this paper. FlexRAN runs in a gNB and allocates resources to different RAN slices in a dynamic environment. However, there is significant difference with our work: we propose an intent based management of RAN control parameters using CAN whereas FlexRAN is designed for RAN slicing. To the best of our knowledge, we are the first ones to address the problem of possible contradiction within a single intent and propose intent based orchestrator, enabled with contradiction detection and removal, for RAN control parameters.

B. Existing standards

The 3rd Generation Partnership Project (3GPP), ETSI, Open Networking Foundation (ONF) and International Telecommunication Union (ITU) have all developed their own study groups on intent based networking. TR:28.812-"Study on scenarios for Intent driven management services for mobile networks" in Release 16 in the scope of SA5 is 3GPP's effort on intent-based network management that started back in 2018 and is still an ongoing effort [16]. It describes the concept of intent-driven management services (IDMS) to consumers of 5G. Use of IDMS is planned to originate from network service providers and operators in the scenarios of network provisioning, network optimization, coverage and capacity management - precisely the scenarios mentioned in this paper.

ETSI also started the Zero-touch network and Service Management (ZSM) working group [17] for describing means for network automation in 2018. This document concentrates on policy-driven and intent-based automation along with intent-based service orchestration. ITU-T Study Group 13 defines intent as a declarative mechanism (written in ML meta-language) where tech-agnostic ML use case can be deployed by operators inside their focus group on ML for Future Network including 5G (FG-ML5G). In that sense, intents are used as high level ML pipeline components. However, building this meta-language is also foreseen as one of the main challenges in future implementation of intent-based networking.

VI. Impacts on standardization

After discussing the existing standards on intent-driven RAN management, in this Section we discuss the impact of our work on ongoing standardization efforts. It is advisable that any real-life deployment of ICDR supports multi-vendor integration. To achieve so, inputs and outputs of ICDR need to be specified in network management specifications, for example, in 3GPP SA5 or ETSI ZSM. Although 3GPP SA5 already provides descriptions on controlling the behavior of the CFs by configuring their goals, it does not provide any mean for configuring the Controller. So, in that case 3GPP resource model needs to be extended with models for CAN Controller functionality and the methods to configure such Controller.

On the other hand, the ICDR takes a formal intent based on which it determines if the intent can be fulfilled without any contradiction. Such a formal intent may be generated and compiled by an ISP coming from any non-telco vendor like a ML audio processing startup. To allow for integration between these non-telco-centric ISPs and the telco-centric ICDR, the intent specification interface and specifically the structure of formal intent need to be standardized. This is actually an in-extensive extension to existing standards since the formal intent's attributes and their values are already speechified in the existing specifications. For example, managed objects, control parameters and metrics are already specified in the 3GPP network resource models. Additionally, the intent specification interface needs to be extended with messages through which

- The ICDR indicates to the operator that a given intent contains contradictions.
- The ICDR informs the MNO of possible alternatives.
- The MNO specifies a given preference among the candidate options.

VII. Conclusion

IBN plays a crucial part in network and service management in the next generation networks. Although there exist quite a number of research papers on IBN based RAN management, majority of them only provide an abstract overview without any implementation design. In this paper we discussed an IBN based CAN orchestration for RAN management and implemented our proposed solution in a simulation environment for evaluation purposes. Along with that, we also discussed the impact of our work on standardization efforts. In future we plan to study and evaluate different weighted prioritization solutions, instead of NSWF, in this solution framework.

References

[1] L. Pang, C. Yang, D. Chen, Y. Song, and M. Guizani. A survey on intent-driven networks. *IEEE Access*, 8:22862–22873, 2020.

[2] A. Banerjee, S. S. Mwanje, and G. Carle. Game theoretic conflict resolution mechanism in cognitive autonomous networks. In *2020 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, pages 1–8. IEEE, 2020.

[3] A. Banerjee, S. S. Mwanje, and G. Carle. An intent-driven orchestration of cognitive autonomous networks for ran management. In *2021 17th International Conference on Network and Service Management (CNSM)*, pages 380–384. IEEE, 2021.

[4] S. Hämäläinen, H. Sanneck, and C. Sartori. *LTE self-organising networks (SON): network management automation for operational efficiency*. John Wiley & Sons, 2012.

[5] A. Banerjee, S. S. Mwanje, and G. Carle. Optimal configuration determination in cognitive autonomous networks. In *2021 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 494–500. IEEE, 2021.

[6] K. M. Addali, S. Y. B. Melhem, Y. Khamayseh, Z. Zhang, and M. Kadoch. Dynamic mobility load balancing for 5g small-cell networks based on utility functions. *IEEE Access*, 7:126998–127011, 2019.

[7] A. Banerjee, S. S. Mwanje, and G. Carle. On the implementation of cognitive autonomous networks. Internet Technology Letters. 2021; e317. https://doi.org/10.1002/itl2.317, 2021.

[8] Eric Van Damme. The nash bargaining solutions is optimal. *Journal of Economic Theory*, 38(78):100, 1986.

[9] Eric Van Damme. The nash bargaining solutions is optimal. *Journal of Economic Theory*, 38(78):100, 1986.

[10] Y. Wei, M. Peng, and Y. Liu. Intent-based networks for 6g: Insights and challenges. *Digital Communications and Networks*, 6(3):270–280, 2020.

[11] OSM. Open source mano. Available online: https://osm-download.etsi.org/ftp/Documentation/201902-osm-scope-white-paper/#!02-osm-scope-and-functionality.md.

[12] ONAP. Onap: Open networking automation platform. Available online: https://www.onap.org/.

[13] R. Ferrus, O. Sallent, J. Pérez-Romero, and R. Agusti. On 5g radio access network slicing: Radio interface protocol features and configuration. *IEEE Communications Magazine*, 56(5):184–192, 2018.

[14] R. Ferrús, O. Sallent, J. Pérez-Romero, and R. Agusti. On the automation of ran slicing provisioning and cell planning in ng-ran. In *2018 European Conference on Networks and Communications (EuCNC)*, pages 37–42. IEEE, 2018.

[15] Mosaic5G. Network sharing using flexran. Available online: https://mosaic5g.io/flexran/.

[16] 3GPP Technical Specification Group Services and System Aspects. Study on scenarios for intent driven management services for mobile networks, telecommunication management. https://bit.ly/2vJ2aOr, 2019.

[17] ETSI. Zero touch network and service management (zsm) means of automation. https://bit.ly/2THSfRl, 2018.