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 (IBNs) are mainly used to transform a user’s intent into network configuration, operation, and maintenance strategies. IBN is a prominent feature for designing the AI-enabled next generation networks. In this paper we propose intent-driven orchestration of cognitive autonomous RAN for managing network control parameters. Our proposed design enables the Controller to detect and remove contradiction, which may arise from a single intent, in the runtime. We also provide a brief overview of existing standards on this aspect and standardization impact of our research to show that it conforms with the worldwide mobile network management 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 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 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 and automation 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 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 of our proposed solution, we also discuss about the impacts of our work on standardization efforts in this paper.

The rest of the paper is organized as follows. The following Section gives an overview of existing works on CAN and intent-driven orchestration of CAN. In Section III we identify the flaws of the design regarding contradiction issues and elaborate on the problem addressed in this paper. To overcome the flaws, we propose a new end-to-end architectural design for intent contradiction detection and removal in Section IV. Section V covers existing research works in intent based resource allocation in RAN and existing standards, following which we describe the standardization impact of our work and conclude the paper in Section VII. Also, abbreviations used in this paper are listed in Table I.

II. Overview of intent-driven orchestration of CAN for managing RAN control parameters

There are several adjustable network control parameters (NCPs) (e.g., TXP) and key performance indicators (KPIs) (e.g., radio link failures (RLF)) in a cell. A KPI value can be affected when a certain group of NCPs are changed. For example, cell downlink throughput can be affected by changing cell TXP and RET. Also, changing a single NCP might affect multiple KPIs simultaneously. For example, changing cell TXP can affect both downlink throughput and cell load. This type of inter-connected relationships among NCPs and KPIs make the management of RAN a challenging task. Since all KPIs are crucial in determining the quality-of-service (QoS) of the network, at a certain
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             |
| ICDDR        | 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          |
| NPV          | 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                |

point of 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 an 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 as intents and the layer of abstraction generates the appropriate actions to be executed in the network. In [3] we proposed the design of such a layer of abstraction 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), in CAN. A CF has the responsibility to learn the dependency of the managed KPI on the NCPs and how the managed KPI changes when NCPs change [2]. The NCPs, which are required to be changed to adjust the managed 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 one or multiple ICPs are changed in different network states. Based on the learning, a CF can determine the necessary configurations (values of the ICPs) for which its output is optimal in a certain network state [4].

To coordinate among the CFs, there is a Controller whose main responsibility is to resolve conflicts regarding shared resources [2], [4]. The idea of a Controller in CAN is not new and exists in SON as well [5]. In [2], [4] we explained and justified the necessity of the CAN Controller. After the network state changes, if a CF determines new values of ICPs which optimize its output, the CF conveys this information to the Controller. For each ICP, the Controller takes the interests of all CFs into account before determining a final value which is optimal for the combined interest of all CFs [6]. Without any intricate details we can abstract a CAN as shown in Fig. 1. $F_1$ and $F_2$ are two CFs with a shared ICP; $p$. OCRS and UF are two metrics using which a CF can convey its preferences over an ICP to the Controller [4], [6].

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

An intent from MNO can be provided via any form: audio input, command line input (CLI), text, or, a graphical user interface (GUI) command. So it is necessary to parse the content of the intent and convert the intent to a predefined structure which is understandable by different network components. This intent, which is 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.

C. IDNAFO

IDNAFO acts as a bridge between ISP and CAN - it takes a formal intent from ISP 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. The steps, along with the name and functionality of each block, are described below:

- After receiving a formal intent from ISP, first task of IDNAFO is to check 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 that an intent cannot be executed by CAN, depending on the implementation design it may forward the intent to some other places within the system 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.

Implementation of each IDNAFO functionality is shown in Fig. 2, other details can be found in [3].

III. Problem of contradiction in RAN management intents

Although we see that IDNAFO with CAN and ISP provide an end-to-end design to execute optimization intents, it does not detect the contradictions arising from the same intent. To elaborate it further, let us consider the following intent as an example: "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 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. Similarly, if MLB and MRO propose different CIO values, that gives rise to a contradiction as well.

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, these network state dependent values are generated by the CFs based on their learning and not shared with 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 framework

To overcome the aforementioned problem, we propose a new functionality called Intent Contradiction Detector and Remover (ICDR). This functionality can be implemented as a separate entity or within the Controller. Main 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 any other type of interface (e.g., command line interface, audio interface) that interacts with MNO, and, (ii) ICDR that interacts with CAN for actual intent execution. The interface provides MNO with two options (Fig. 2):

- 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 does not contain any contradiction, it is recommended to select this option. Usually intent that deals with a single NCP or KPI can be safely selected for execution without fearing contradiction.

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 given 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

In Section IV-A we discussed how ICDR detects a contradiction within an intent. In this Section we discuss how ICDR resolves the contradiction after detecting it.

In the case of a contradiction, ICDR requests all the CFs involved (following the example, MLB and MRO) to send their utility functions (UFs) [6] over the conflicting NCP(s) (in P2, conflicting NCP is CIO, in P3 it is TTT and in P4, both TTT and CIO). Before discussing further about contradiction removal, let us elaborate on the UF.

Objectives of different CFs have different units of measurement. For the Controller to understand and compare among these different objectives, it is beneficial to convert all of them into some identical predefined scale. Example of one such a scale is $[0:10]$, where 0 means the lowest and 10 means the highest objective value. To convert its ICP to this scale, each CF generates a function which is called UF. A UF is denoted by $f(p)$, where $p$ means the configurations and for this $p$, $f(p)$ provides the utility of the CF in the predefined scale.

After obtaining the UFs, ICDR finds the conflicting NCP value for which the product of the UFs is maximum, because, according to Nash Social Welfare Function (NSWF), that value is optimal for the combined interest of all the CFs involved [7]. Let us elaborate further with an example. Let us assume that following the intent we discussed earlier, 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 regarding CIO, 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. 3 and send them to MNO to choose from, highlighting the recommended option. Targets specified in the intent are highlighted in the first two rows, calculated by ICDR, is listed in the last row. After the MNO sees these alternatives, he can choose the one that suits his interests the best.

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 for CAN, it sends the intent back to MNO or to some other networking entities, based on the implementation. If the intent is found to be within the scope of CAN, 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.

| Option | CIO | Load reduction | Handover Success increase |
|--------|-----|---------------|--------------------------|
| 1      | $c_l$ | $y_1$ | $x_2$ |     |
| 2      | $c_r$ | $y_2$ | $x_1$ |     |
| 3      | $c_n$ | $y_n$ | $x_n$ |     |
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. 4.

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], [8]. European Telecommunications Standards Institute (ETSI) proposed an open source orchestrator for slicing networks based on NFV specifications [9]. The Linux Foundation also provides ONAP - an open source E2E network orchestrator of virtual appliances [10]. Apart from the orchestrators, [11] described the network slicing framework and the basic design challenges faced for performing RAN slicing in 5G. In [12] the researchers proposed another such management and orchestration framework for slicing of next generation RAN.

Among the existing research works, FlexRAN [13] 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 [14]. 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 [15] 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 specified 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 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 discuss an IBN based CAN orchestration for RAN management and extend the existing design to enable it for contradiction detection and removal. Along with that, we also discuss the impact of our work on standardization efforts.

In our future work, we plan to implement the proposed design as a standalone Python module and integrate it with any existing networking simulator. Since we already implemented CAN in a simulation environment [6], it’ll be interesting to add the intent based interface on top of that implementation and study how the system behaves.

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). IEEE, 2021.

[4] 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.

[5] Daichi Kominami, Masashi Sugano, Masayuki Murata, and Takaaki Hatauchi. Controlled and self-organized routing for large-scale wireless sensor networks. ACM Transactions on Sensor Networks (TOSN), 10(1):1–27, 2013.

[6] 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.

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

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

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

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

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

[12] R. Ferrús, O. Sallent, J. Pérez-Romero, and R. Agustí. 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.
[13] Mosaic5G. Network sharing using flexran. Available online: https://mosaic5g.io/flexran/.

[14] 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.

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