5-1　ITU-T におけるネットワーク制御自動化技術の標準化活動

5-1　Standardization Activities of Network Control Automation in ITU-T

ベド プラサド カフレ
Ved Prasad KAFLE

人工知能（AI）・機械学習（ML）技術は、Beyond-5G（B5G）・6G システムなどの将来の高度なネットワークとサービスの制御・管理を自動化するために必須である。近年、様々な標準化開発機関（SDO）において、AI/ML 統合ネットワーク制御・管理アーキテクチャの標準化活動が進められている。本稿では、ネットワークサービス設計、リソース制御、障害検知及び障害復旧に焦点を当て、国際電気通信連合（ITU）における AI/ML ベースの制御・管理機構に関する標準化活動について報告する。

Artificial intelligence (AI) and machine learning (ML) techniques are crucial for automating the control and management functions of future networks and services, such as beyond-5G (B5G) or 6G systems. Standardization activities for AI/ML-integrated network control and management architectures are currently advancing in various standards developing organizations (SDOs). This paper provides an overview of the standardization efforts for AI/ML-based control and management mechanisms by the International Telecommunication Union’s Telecommunication Standardization Sector (ITU-T), with a focus on AI/ML-based network architectures, service provisioning, resource control, fault detection and recovery mechanisms.

1 Introduction

Artificial intelligence (AI) and machine learning (ML) techniques are being applied to network control and management functions[1]. Future networks, such as International Mobile Telecommunications (IMT) for 2030 and beyond (IMT-2030), commonly known as 6th Generation (6G) mobile systems, will feature ubiquitous intelligence by integrating AI and communication capabilities into a single platform[2].

The International Telecommunication Union’s Telecommunication Standardization Sector (ITU-T) has developed several standard documents, known as ITU-T Recommendations, outlining the high-level architectures for AI/ML-based network control and management technologies. Building on the research and development of AI/ML-based network automation technologies, NICT has actively submitted numerous contributions to ITU-T for the development of these Recommendations[3].

ITU-T initiated a study and standardization effort on the integration of AI/ML in telecommunication networks in 2017 by establishing Focus Group on Machine Learning for Future Networks including 5G (FG ML5G)[4]. During its lifetime of 2.5 years, FG ML5G produced several deliverables and submitted them to ITU-T Study Group 13 (SG13) for consideration to approve them as Recommendations. Subsequently, ITU-T SG13 further refined the deliverables and approved them as four Recommendations and one Supplement, as listed below.

- ITU-T Recommendation Y.3172 – Architectural framework for machine learning in future networks, including IMT-2020.
- ITU-T Recommendation Y.3173 – Framework for evaluating intelligence levels of future networks, including IMT-2020.
- ITU-T Recommendation Y.3174 – Framework for data handling to enable machine learning in future networks, including IMT-2020.
- ITU-T Recommendation Y.3176 – Machine learning marketplace integration in future networks including IMT-2020.
- ITU-T Supplement 55 – Machine learning in future networks including IMT-2020 : use cases.

Among these Recommendations, Y.3172 serves as the foundational architecture framework for incorporating

AI/ML techniques in network control and management. The other Recommendations focus on enabling components such as evaluating intelligence levels, data handling, and integrating ML marketplace.

ITU-T Supplement 55 describes 30 use cases and their requirements. The use cases are classified into five groups: (1) network slice and service, (2) user plane, (3) applications, (4) signaling and management, and (5) security. The requirements are also mapped to three groups of functions: (1) data collection, (2) data storage and processing, and (3) ML models.

Additional Recommendations were later developed from contributions submitted directly to ITU-T SG13. They specify AI/ML-based technologies related to the various aspects of network control and management such as resource and fault management, and service provisioning.

The remainder of this article is organized as follows. Section **2** describes Y.3172, the base architecture for incorporating AI/ML in networks. Sections **3** describes AI/ML-based architecture for network resource and fault management specified in Y.3177. Section **4** describes AI/ML-based framework for network service provisioning specified in Y.3178. These Recommendations were developed from contributions submitted from Japan. Section **5** provides an overview of the newly approved Recommendation Y.3061, specifying an architecture framework for autonomous networks that extends the concept of design-time and run-time operations specified in Y.3177 to autonomous networks.

2 Architectures for Integrating ML in Future Networks

In this section, we delve into the high-level architecture for incorporating AI/ML in networks specified in Y.3172[5]. Additionally, we briefly outline other Recommendations that delineate the mechanisms facilitating intelligence level evaluation, data handling, and ML marketplace integration.

The high-level architecture, as depicted in Figure 1, comprises four subsystems: the ML pipeline subsystem, the ML sandbox subsystem, the management subsystem, and the ML underlay networks.

The ML pipeline subsystem contains a set of ML models

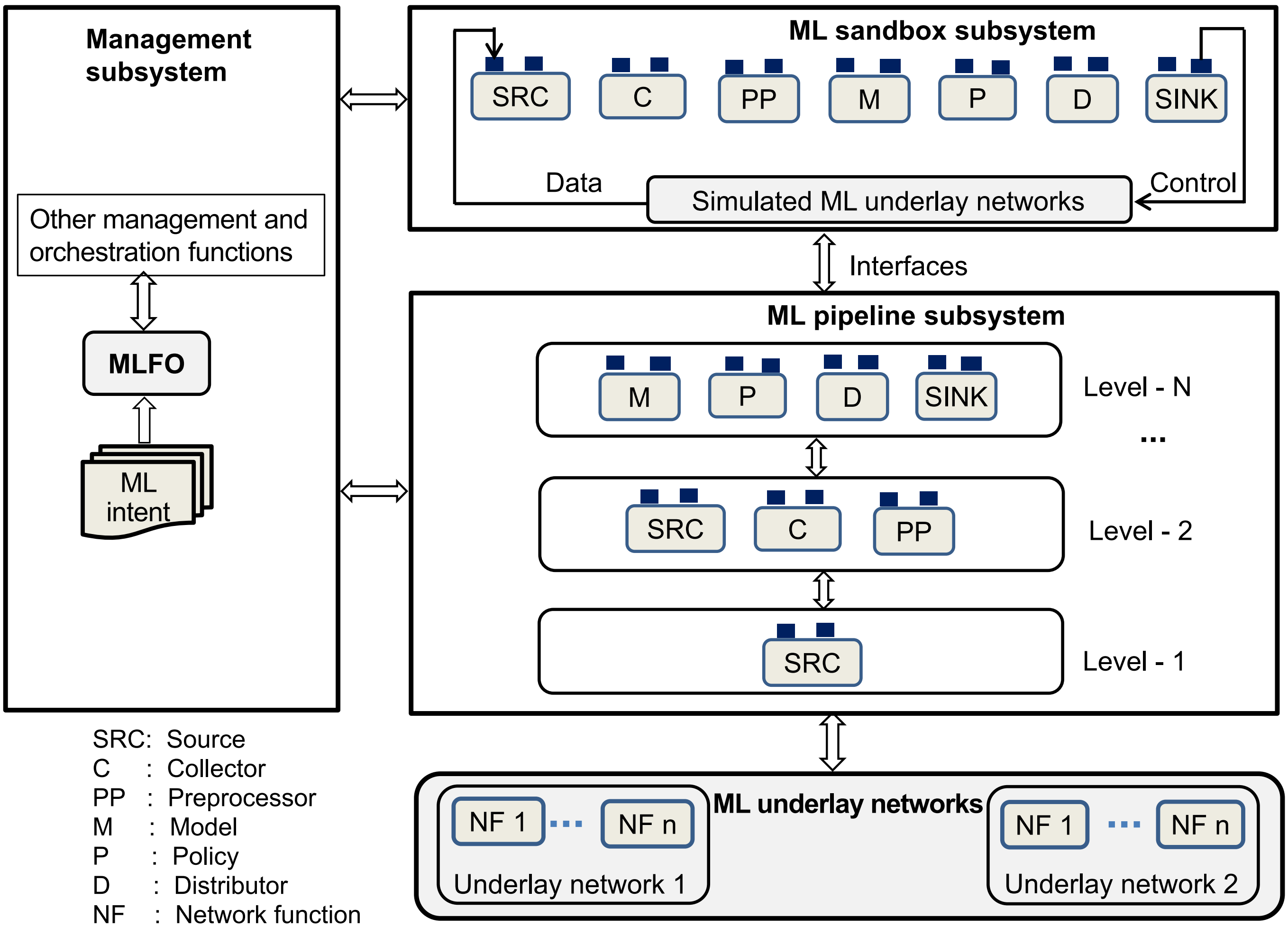

Fig. 1 High-level architecture for integrating ML in future networks [Y.3172]

Table 1　Network intelligence levels [ITU-T Y.3173]

Network intelligence level		Dimensions				
		Action implementation	Data collection	Analysis	Decision	Demand mapping
L0	Manual network operation	Human	Human	Human	Human	Human
L1	Assisted network operation	Human and System	Human and System	Human	Human	Human
L2	Preliminary intelligence	System	Human and System	Human and System	Human	Human
L3	Intermediate intelligence	System	System	Human and System	Human and System	Human
L4	Advanced intelligence	System	System	System	System	Human and System
L5	Full intelligence	System	System	System	System	System

and execution abstraction. It consists of several functional components, such as the input data collector (C), preprocessor (PP), ML models (M), policy (P), and output result distributor (D). The input data are collected through a source (SRC) function, and the ML output results are distributed to the network controllers through a sink (SINK) function.

The ML sandbox subsystem comprises the same functional components as those of the ML pipeline subsystem. It utilizes simulated underlay networks for the training and testing of ML models. Subsequently, these trained ML models are deployed within the ML pipeline subsystem to monitor and manage live ML underlay networks.

The management subsystem comprises a machine learning function orchestrator (MLFO) along with other managerial functions. The MLFO function receives ML intents as input instructions for monitoring and coordinating the operational components within the ML pipeline and ML sandbox. A comprehensive explanation of these functions and interfaces can be found in[5].

Next, we provide a concise overview of enabling technologies of intelligence level evaluation, data handling and ML marketplace integration specified in Recommendations Y.3173, Y.3174, and Y.3176, respectively, produced from the FG ML5G deliverables.

An architecture framework and related methods for evaluating network intelligence levels are specified in Y.3173. The standardized methodologies equip operators, equipment vendors, and other participants in the network industry with a decision-making mechanism for strategizing network technology features and product roadmaps. The intelligence level evaluation encompasses five dimensions: (1) demand mapping, (2) data collection, (3) analysis, (4) decision-making, and (5) action implementation.

Demand mapping involves translating the network configuration or requirements provided by a human operator into specific instructions that network components can comprehend and execute. The dimensions of data collection and analysis pertain to gathering network monitoring and control data and conducting an analysis. The decision and action implementation dimensions involve decision making for the network or service (re)configuration and, subsequently, executing the decision.

To assess the overall intelligence level of an entire network system, the intelligence levels of individual workflows and network subsystems are evaluated across the five dimensions. Workflows encompass tasks such as planning, deployment, maintenance, optimization, and provisioning. Similarly, the network subsystem comprises network elements, management subsystems, and application platforms.

To assess the overall intelligence level of a network, a basic method relying on the intelligence capabilities across the five dimensions is outlined. These capabilities are determined by whether a dimension's function is solely human-based, human and system-based, or solely system-based. From the combinations of these intelligence levels across the dimensions, six distinct intelligence levels are defined, as depicted in Table 1. The lowest level, L0, represents purely manual network operation, where all dimensions are managed by humans. Conversely, the highest level, L5, represents full intelligence, with all dimensions autonomously handled by the system. Intermediate intelligence levels exist where human input is required for certain dimension tasks.

Efficient management of vast amounts of control data gathered from diverse network subsystems and processes is crucial for implementing AI/ML-based control functions in networks. The ML data handling framework, outlined in Y.3174, addresses the diversity in control data. It introduces data models, brokers, and application program interfaces (APIs) in both user and control planes to tackle diversity issues. Additionally, it specifies requirements for input data collection, processing, and output data.

The challenges, motivations, requirements, and architecture for integrating ML marketplace into networks is described in Y.3176. It defines the ML marketplace as a repository of interoperable, trained AI/ML models. Furthermore, it details a method that utilizes ML intent and MLFO to choose suitable ML models from the ML marketplace, along with interfaces for linking the ML marketplace with the ML sandbox and the ML pipeline.

Additionally, aside from the Recommendations developed from the enhancement of the FG ML5G deliverables, ITU-T SG13 has approved several Recommendations based on contributions submitted by members, which are described next.

3 AI/ML-based Architecture for Network Resource and Fault Management

A high-level architecture for AI/ML-based network resource and fault management, as depicted in Figure 2, is described in Y.3177[6]. It extends the base architecture of Y.3172 by introducing additional functionalities for resource and fault management in the management and AI/ML pipeline subsystems.

The management subsystem contains three groups of management functions: resource management, fault management, and other management functions. ML models undergo through three phases of training, evaluation, and deployment. In the training phase, an ML model is instantiated in the sandbox using the dataset based on the intent. In the evaluation phase, the trained ML model is verified if it meets the requirements specified in the ML intent. Finally, in the deployment phase, the ML model is deployed in the AI/ML pipeline.

The AI/ML pipeline comprises six functional components focused on resource and fault management: data collection, fault detection, fault recovery, resource

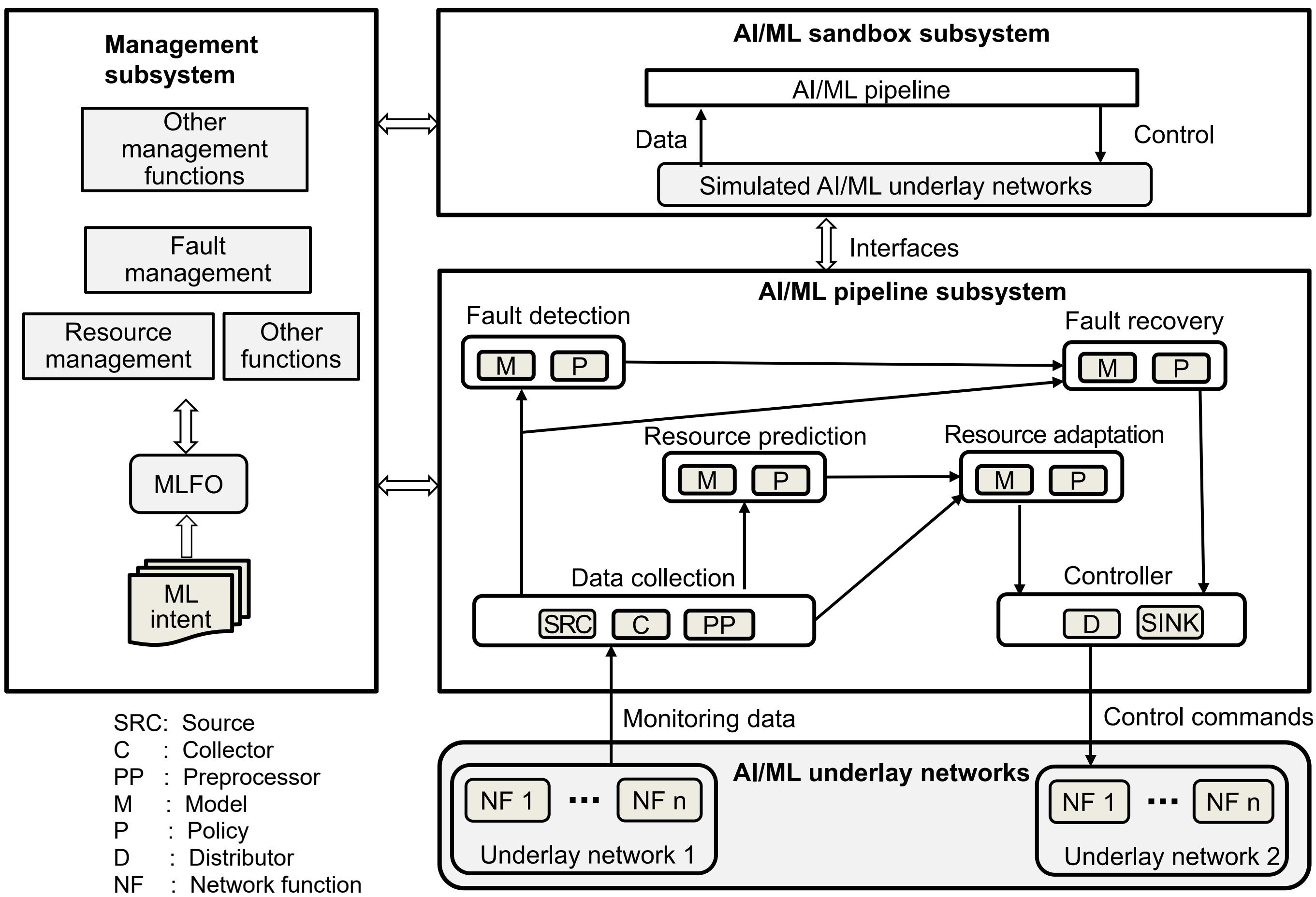

Fig. 2 High-level architecture of AI/ML-based network resource and fault management

prediction, resource adaptation, and controller.

The fault detection function, supported by ML model (M) and prediction (P) functions, identifies network faults by analyzing monitoring data acquired from the data collection function. The M function analyzes data to detect fault occurrences, while the P function determines the root cause of faults and alerts the fault recovery function accordingly. The fault recovery function, also assisted by the M and P functions, instantiates an appropriate recovery action upon receiving fault notifications from the fault detection function.

The resource prediction function, supported by the M and P functions, analyzes network measurement data to infer patterns or trends of resource utilization and predict future resource demands across various network functions. The resource adaptation function, supported by the M and P functions, determines suitable resource adaptation methods, including resource arbitration, network function migration, and network slice reconfiguration. Detailed descriptions of these methods are provided in[6].

The controller function, supported by distributor (D) and sink functions, executes AI/ML action decisions in the AI/ML underlay network. The AI/ML sandbox comprises an AI/ML pipeline designed for training AI/ML models utilizing data obtained from the simulated AI/ML underlay network.

4 AI-based Framework for Network Service Provisioning

A functional framework for AI-driven network service provisioning is specified in Y.3178. It starts with a description of the business role-based model for network service provisioning, then provides a functional framework showing the components and their interactions in AI-driven operation for network service provisioning.

Network services are configured across diverse ICT infrastructures like cloud computing, edge computing, and various networking technologies. For delivering dependable application services, mechanisms are required to ensure an optimal mapping between application service requirements and ICT infrastructure capabilities. Achieving such mapping can be notably intricate, particularly when application services span across diverse networking domains, possibly under the purview of different stakeholders. Given the complexity arising from both the diversity in application services and ICT infrastructures, leveraging AI/ML techniques would prove beneficial for automating the closed-loop mapping process.

Y.3178 identifies several stakeholders involved in network service provisioning, including the application user, application service integrator, communication provider, and network provider. It describes the AI-driven network service provisioning functional components pertinent to the application service integrator and communication service provider.

The application service integrator is responsible for offering application services to the application user. It extracts networking-related application requirements and translates them into communication service-specific requisites, which must be accommodated by the underlying capabilities of a communication service provider. Conversely, the communication service provider, catering communication services to the application service integrator, extracts network-related service requirements and converts them into network service-specific requirements, aligning with the capabilities of underlying network providers.

Figure 3 illustrates the sub-roles of both the application service integrator and communication service provider, along with the message flows between them. The message flows indicate that service requirements move from upper layers to lower layers, while service status information moves in the reverse direction. The sub-roles of service provider and service user do exist in both the application service integrator and communication service provider. However, the roles of mediator and network orchestrator are distinctive and are particularly associated with AI/ML, which will be elaborated next.

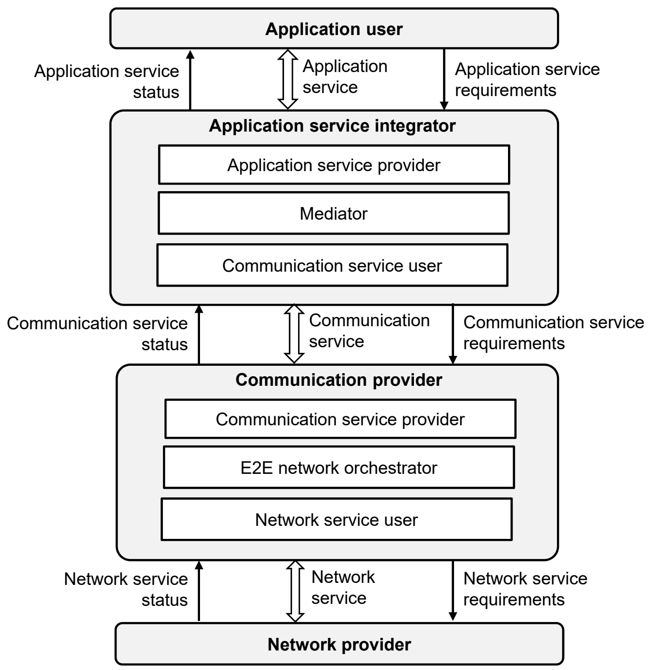

Fig. 3 Roles of application service integrator and communcaiton service provider [Y.3178]

4.1　Mediator

The mediator translates the application's service requirements into communication service requirements. It uses various additional pieces of information such as the application user's resource usage trends and best practices, to meet each application's requirements. The resulting communication service requirements are transmitted to the communication provider through the communication service user.

Figure 4 illustrates the functional components and information flows that support the mediator's activities. It comprises three main functional components: extractor, analyzer, and translator. The extractor is responsible for extracting and validating networking requirements from the application service requirements. The analyzer processes these network requirements to generate abstract communication service requirements, which are independent of specific technologies and implementations. Finally, the translator converts the abstract communication service requirements into capability parameters specific to the underlying communication providers.

4.2　End-to-end network orchestrator

The E2E network orchestrator translates communication service requirements into network service requirements that the underlying network provider must meet by its capabilities. Figure 5 illustrates the functional components and information flows that support the activities of the E2E network orchestrator. These functional components include configuration designer, management procedure planner, and E2E network service manager.

The configuration designer is responsible for creating the design of an end-to-end network service configuration based on the communication service requirements. It produces an end-to-end network configuration that meets these requirements. To simplify the real-time design process, it selects a suitable configuration from the candidate configuration patterns generated by AI/ML models.

The management procedure planner receives the E2E network configuration from the configuration designer and creates the management procedure for the E2E network service as a structured list of network service requirements. This E2E network configuration contains detailed descriptions of specific network services and the relationships (including dependencies) between them.

The E2E network service manager receives this management procedure for the E2E network service and executes it. This process involves fulfilling the network service requirements by making API calls to the appropriate network providers.

During the execution of this procedure, it's not always possible to meet all the network service requirements due to limitations in the availability of network resources. In such cases, based on the status information of the network service received from the relevant network providers, the E2E network service manager gives feedback to the configuration designer. The designer can then adjust the configuration of the E2E network service while ensuring that the communication service requirements are still met. The adjustment result is notified to the communication service provider.

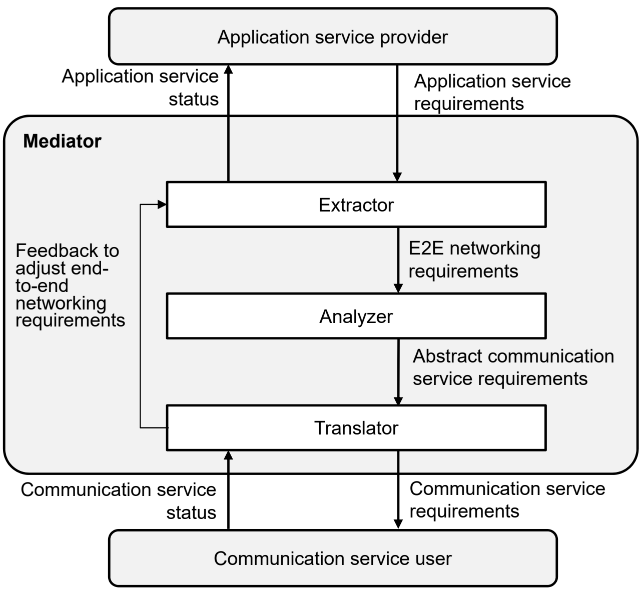

Fig. 4　Functional components and information flows of mediator sub-role [Y.3178]

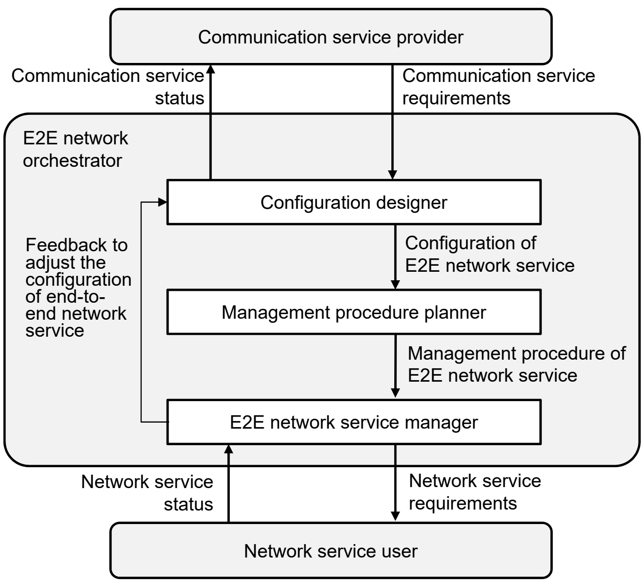

Fig. 5　Functional components and information flows of E2E orchestrator [Y.3178]

5 Autonomous Networks

ITU-T SG13 established the Focus Group on Autonomous Networks (FG AN) in December 2020 to create an open platform for pre-standardization activities related to autonomous networks (AN)[8]. FG AN defined autonomous network as the network system capable of monitoring, operating, recovering, healing, protecting, optimizing, and reconfiguring itself.

FG AN produced several deliverables covering use cases, gap analysis, architecture framework, trustworthiness, and knowledge management. ITU-T SG13 further improved the architecture framework deliverables and approving it as ITU-T Recommendation Y.3061[9].

Figure 6 shows the architectural framework for autonomous networks. The objective of this architecture is to facilitate the continuous evolution, creation, validation, and implementation of a set of controllers that enable the network and its services to operate autonomously. The framework comprises key components of the autonomy engine, dynamic adaptation subsystem, knowledge base (KB) system, AN orchestrator, underlay network, and E2E network orchestrator.

The autonomy engine, which is responsible for creating and validating controllers, consists of two main subsystems: exploratory evolution and online experimentation. The exploratory evolution subsystem facilitates the adaptation of controllers to changes in the underlying network. It uses knowledge stored in the KB subsystem to support continuous evolution and exploration. This process allows for the automated design or modification of controllers and their hierarchies, exploring a range of possible controller logics to adapt to the operational environment.

The experimentation subsystem focuses on validating controllers. It designs, orchestrates, and executes experimental scenarios using the experimental controller and AN sandbox. The experimental controller generates potential scenarios based on controller designs and use

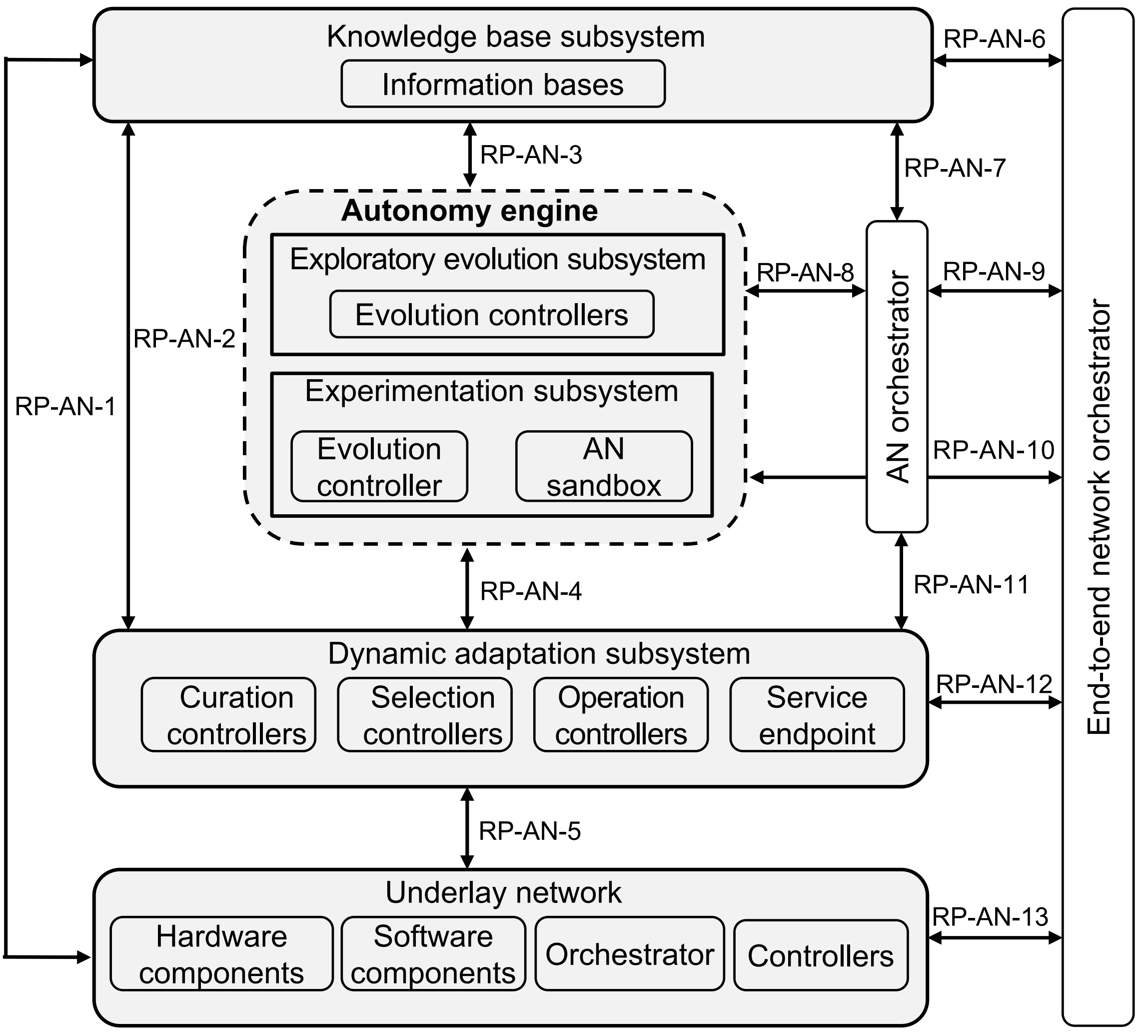

Fig. 6　Architecture framework for autonomous networks [Y.3061]

case representations. The AN sandbox provides an environment where controllers can be deployed and experimentally validated with the assistance of domain-specific models of underlying networks.

The dynamic adaptation subsystem embodies the core concept of dynamically adapting controllers, enabling them to autonomously handle new and unforeseen changes in the underlay networks. It is composed of different controllers, including curation, selection, and operation controllers.

The KB system manages the operations of storage, querying, and updating of knowledge, which includes metadata derived from the capabilities and status of AN components. The AN orchestrator oversees the workflows and processes within the AN, as well as the lifecycle of the controllers. The E2E network orchestrator consists of a set of functions that interface with the subsystems of the AN architecture framework to manage and orchestrate the underlay network functions.

The E2E network orchestrator is tasked with managing and orchestrating control entities within the AN, including those in the underlay networks.

The architecture also specifies several reference points. RP-AN-1, RP-AN-2, RP-AN-3 and RP-AN-6 are reference points between the KB subsystem and underlay network, dynamic adaptation subsystem, autonomy engine, E2E network orchestrator and AN orchestrator respectively, to enable access to the KB from these subsystems. RP-AN-4 is between the autonomy engine and dynamic adaptation subsystem to provide evolutionary exploration and experimentation functionalities to the dynamic adaptation subsystem. RP-AN-5 is between the dynamic adaptation subsystem and underlay network to provide selection and integration of controllers to the underlay network, as the underlay network conditions change at run-time. RP-AN-7, RP-AN-8 and RP-AN-11 are the reference points between the AN orchestrator and KB, autonomy engine and dynamic adaptation subsystem, respectively, to enable the AN orchestrator to manage the workflows and processes in the AN, and the lifecycle of controllers. RP-AN-9, RP-AN-10, RP-AN-12 are the reference points between the E2E network orchestrator and AN orchestrator, autonomy engine and dynamic adaptation subsystem, respectively, which are used by the E2E network orchestrator to manage and orchestrate control network entities. RP-AN-13 is the reference point between E2E network orchestrator and underlay network to manage and orchestrate control network entities in the underlay network.

6 Conclusion

This article describes the current state of standardization for AI/ML-based network and service management architectures developed by ITU-T. These architectures utilize AI/ML model pipelining to enable autonomous operations in control information handling, network intelligence evaluation, network service provisioning, and resource and fault management. Additionally, the article provides a brief overview of the architecture framework for autonomous networks.

References

1 V.P. Kafle, P. Martinez-Julia, and T. Miyazawa, “Automation of 5G network slice control functions with machine learning,” IEEE Communications Standards Magazine, vol.3, no.3, pp.54–62, Sept. 2019.
2 ITU-R Recommendation M.2160 (11/2023) – Framework and overall objectives of the future development of IMT for 2030 and beyond, Nov. 2023.
3 V.P. Kafle, T. Hirayama, T. Miyazawa, M. Jibiki, and H. Harai, “Network control and management automation: Architecture standardization perspective,” IEEE Communications Standards Magazine, vol.5, no.3, pp.106–114, Sept. 2021.
4 ITU-T Focus Group on Machine Learning for Future Networks including 5G, https://www.itu.int/en/ITU-T/focusgroups/ml5g/Pages/default.aspx, visited on 15th May 2024.
5 ITU-T Recommendation Y.3172 (06/2019), Architectural framework for machine learning in future networks including IMT-2020, June 2019.
6 ITU-T Recommendation Y.3177 (02/2021), Architectural framework of artificial intelligence-based network automation for resource and fault management in future networks including IMT-2020, Feb. 2021.
7 ITU-T Recommendation Y.3178 (07/2021), Functional framework for artificial intelligence-based network service provisioning in future networks including IMT-2020, July 2021.
8 ITU Focus Group on Autonomous Networks (FG-AN), https://www.itu.int/en/ITU-T/focusgroups/an/Pages/default.aspx, visited on 15th May 2024.
9 ITU-T Recommendation Y.3061 (12/2023), Autonomous networks – Architecture framework, Dec. 2023.

Ved Prasad KAFLE （ベド プラサド カフレ）

Research Manager,
Network Architecture Laboratory,
Netowork Research Institute
Ph.D.
Network Architecture,
Beyond 5G or 6G Networks
【受賞歴】
2023 年　情報通信技術賞（TTC会長表彰）Information and Communication Technology Award (TTC President’s Commendation)
2020 年　Best Paper Award (First prize), ITU Kaleidoscope Academic Conference 2020
2017 年　Accomplishment Award, The ITU Association of Japan