Hidden Figures: Comparative Latency Analysis of Cellular Networks with Fine-grained State Machine Models

Sangwook Bae*
KAIST
sangwook.bae@kaist.ac.kr

Mincheol Son*
KAIST
mcson@kaist.ac.kr

Sooel Son
KAIST
sl.son@kaist.ac.kr

Yongdae Kim
KAIST
yongdaek@kaist.ac.kr

ABSTRACT

Comparative latency analysis of cellular network control planes is an intuitive and effective method used to determine the superiority/inferiority of one cellular network over others. However, operational policies and network configurations vary across different networks, making it difficult to conduct fine-grained latency comparisons. We present a novel diagnostic method for the comparison of the latencies in processing control plane messages among cellular networks. For each cellular network, we automatically build a fine-grained state machine based on control plane signaling messages collected from user smartphones. From the state machines of multiple network operators, we identify common state transitions consisting of signaling messages. We then compare the latencies in the message intervals for each identified common transition. We discovered 38 bottleneck intervals from three representative control plane procedures by analyzing the state machines of five major operators. Our promising preliminary analysis deserves further research.

CCS CONCEPTS

• **Networks** → **Network performance analysis**; **Mobile networks**; • **Computing methodologies** → *Model development and analysis.*

KEYWORDS

Cellular network, State machine modeling, Comparative study, LTE Control plane

ACM Reference Format:
Sangwook Bae, Mincheol Son, Sooel Son, and Yongdae Kim. 2019. Hidden Figures: Comparative Latency Analysis of Cellular Networks with Fine-grained State Machine Models. In *The 20th International Workshop on Mobile Computing Systems and Applications (HotMobile '19), February 27–28, 2019, Santa Cruz, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3301293.3302352

*Both authors contributed equally to the paper

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
HotMobile '19, February 27–28, 2019, Santa Cruz, CA, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6273-3/19/02…$15.00
https://doi.org/10.1145/3301293.3302352

1 INTRODUCTION

Cellular network operators strive to optimize their core network operations for the provision of high-quality services. A dominant factor in the performance of a cellular network is the control plane [11]. For instance, when a user's smartphone returns to its cellular network from the airplane mode, the control plane latency in an `ATTACH` procedure prolongs the waiting time. The latency in a `ServiceRequest` procedure also undermines the promptness of wireless Internet over LTE. Therefore, carriers focus on analyzing the latencies of the control plane of a target network as well as diagnosing the root causes of any possible bottlenecks.

A typical performance diagnosis involves measuring the latency of each control plane procedure with a local view and vetting whether the measurements satisfy an internal policy. This approach often fails to detect latency problems, thus disregarding the opportunity for performance improvements. Cellular network architectures, optimization logic, and configurations vary with different operators. Therefore, a bottleneck may exist only in one network operator. Taking local measurements without comparing the latency with other operators leads to failure in detecting such problems.

SCAT [6] showed promising results in a comparative study, which discovered six major problems of the control plane by using 17,710 circuit-switched fallback calls. It detected abnormal operations by comparison and then diagnosed the problem by manual inspection of signaling messages and standards. However, this study had two limitations: 1) the conducted analysis was coarse-grained, and 2) a significant portion of the analysis was manual. The first limitation is problematic because the study investigates abnormal control plane operations at the high level. For example, we observed that various operational scenarios performing the `ATTACH` procedure showed different latencies. Because SCAT ignored these individual operation scenarios, the root cause analysis had to be completely manual and speculative, inevitably causing the second limitation.

In this work, we present a novel approach for automatically constructing a fine-grained state machine and utilizing it for the comparative latency analysis of the cellular control plane. We start with automatically building a state machine from the control plane signaling messages from smartphones. We collect uplink/downlink messages as well as non-access stratum (NAS) state information from smartphone chipsets. For each network operator, we model the state machine so that a state becomes an observed NAS state, and a transition between two states becomes a sequence of the signaling messages that cause the change in state. We then compare the latencies between messages on common transition paths across

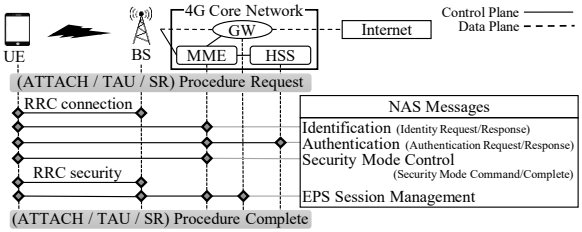


Figure 1: Control Plane Architecture

different state machines from multiple operators.

Our method is distinguished from SCAT by two design choices: **the automatic trace-driven modeling** and **the fine-grained comparative analysis considering various operation scenarios**. Building a full state machine according to multiple 3GPP specifications is a daunting task requiring enormous engineering effort. The proposed trace-driven modeling requires no effort in understanding the 3GPP specifications. By design, our state machine mirrors the operational model of each network operator that is already in service based on the signaling messages from smartphones. For the comparative analysis, we identify the same transition paths across different state machines and compare the latencies on the message intervals on them. That is, each transition path represents a common operation scenario between operators, which provides the ground for fair comparison.

Our state machine representation facilitates various comparative studies, from latency diagnosis to operating logic analysis. In this paper, we focus on a comparative study on the latency of three representative procedures: `ATTACH`, `TrackingAreaUpdate`, and `ServiceRequest`. We compare the shared transition path from the initial state to the normal state of each procedure at the message level. Our system then identifies problematic cases where the latency between two messages is higher than that in the shared transition path of the others' state machines. To demonstrate the effectiveness of our approach, we generate state machines using 390K signaling messages collected from five major carriers in two countries. Through this comparative study, we identified a total of 38 problematic cases. Each case is presented with a specific operation scenario constituted by actual signaling messages, which help us identify causes for latency. Our in-depth analysis discovered entities causing long latency, unnecessary encryption, and the absence of the authentication procedure, each of which is due to different operator-specific operation policies and configurations.

Pinpointing the bottlenecks is essential, but a difficult task without domain- and operator-specific knowledge. This paper highlights the promising results of a comparative analysis using fine-grained state machine representation of the cellular network via trace-driven modeling, thus requiring less effort for understanding of 3GPP standards.

2 CELLULAR CONTROL PLANE

A cellular network consists of user equipment (UE), a base station (BS), and a core network (Figure 1). The UE communicates with a BS using radio resource control (RRC) as well as exchange control plane messages with the LTE (4G) core network. The LTE core network has three entities: HSS (Home Subscriber Server), gateways (GWs), and MME (Mobility Management Entity). The HSS is a database containing subscriber information including phone numbers and service quality profiles. Also, the HSS provides the functionality of user authentication and access authorization. It manages the security information of subscribers and generates their authentication vectors. The GWs provide connectivity between the UE and packet data networks (*e.g.*, Internet). The GWs also assign IP addresses to the UE, and track usage records for billing. The MME provides the UE with EMM (Evolved Packet System (EPS) Mobility Management) and ESM (EPS Session Management) services through various entities (*e.g.*, BS and HSS) and protocols (*e.g.*, RRC and NAS). ESM procedures build a pipeline for data/voice services, called a bearer. EMM procedures control the mobility of users as well as establish user identity and data confidentiality over the bearers. In this paper, we focus on the three essential EMM procedures.

`ATTACH`: The `ATTACH` procedure is the first step for a UE to use any cellular service. It sends the unique identifier of the UE to an MME and establishes a secure channel between the UE and a BS after following the authentication procedure.

`TrackingAreaUpdate`: Cellular operators (in short, ISPs) divide the entire network into multiple tracking areas (TAs) for efficient mobility management. The core entities have the TA information of the UE. The UE updates its TA code through the `TrackingAreaUpdate` (`TAU`) procedure. The TAU procedure is conducted when moving to another TA or when switching between LTE and 3G cellular networks.

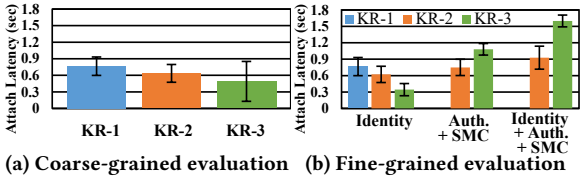
`ServiceRequest`: Any network demands the UE to establish a session for the data service. The UE sends an MME a `ServiceRequest` (`SR`) message for a user data session between the UE and a GW.

To complete the EMM abovementioned procedures, the MME performs the common procedures; identification, authentication and security mode control. In the identification procedure, the MME identifies a UE by exchanging `IdentityRequest` and `IdentityResponse` messages, and requesting its identification parameter. If necessary, the MME initiates an authentication procedure. The MME sends the UE an `AuthenticationRequest` message, derived from the UE's security key in the HSS. The UE then authenticates the network by validating the received message, and sends its response through the `AuthenticationResponse` message to the MME for mutual authentication. Also, if the security key is not present or the MME wants to use a new key, it initiates the security mode control (SMC) procedure by exchanging `SecurityModeCommand` and `SecurityModeComplete` messages. This procedure plays a key role in establishing the secrecy and integrity for the communication between the UE and the MME. Note that those operations are invoked *selectively* according to the status of the UE and the core networks, and the operating policies.

3 COARSE- VS. FINE-GRAINED ANALYSIS

Latency analysis on a cellular control plane is an essential process for analyzing the efficacy of deployed components (*e.g.*, BS, MME) or new optimization configurations (*e.g.*, S1-flex). Furthermore, designing a new core network requires identifying bottleneck points for scalable and robust cellular services [5, 11, 14, 15].

Coarse-grained Analysis. The common approach for this latency analysis is to run a naïve field test. A local tester with a smartphone measures the completion time of the NAS and RRC control plane procedures. The tester then compares the procedure completion times with their service-level standards, which only reflects the local view of the system. The operators measure their network only, without comparing and sharing the detailed results with the



(a) Coarse-grained evaluation (b) Fine-grained evaluation

Figure 2: Attach latency with two evaluation methods

other operators. We emphasize that such procedure-level performance analysis is rough. One procedure entails multiple operation scenarios, each of which involves a different set of NAS/RRC messages and participating entities, such as GWs or HSS. For instance, the TAU procedure covers 118 scenarios affected by the existence of pre-established RRC connections, expired identities, additional processes related to HSS or GWs (*i.e.*, authentication and session establishment), as well as failure recovery [1]. The tester usually knows little about the operation scenarios under testing, and focuses only on collecting the current measurement metrics.

We argue that current procedure-based latency analysis is unfit for comparative analysis. A blind comparison of the procedure completion time does not account for diverse operational scenarios. We observed that the relative order of the performance results of multiple operators differs in each scenario for ATTACH, TAU, and SR. For example, Figure 2 shows the completion times of ATTACH procedures among three ISPs in Korea. Solely based on a blind result comparison, one can conclude that KR-3 shows a better result than the others (Figure 2a). However, the performance of ATTACH varied in different scenarios. Figure 2b shows that KR-3 performs the worst when involving identity, authentication, and SMC messages. Thus, without considering each case, one can easily make the wrong assumption that a given ISP performs faster than the rest. Surprisingly, we observed that the previous works [6, 11] conducted this type of comparative analysis, ignoring diverse operation scenarios.

We also emphasize that local measurements without any comparison to other operators limits the detection of bottlenecks that stem from misconfiguration or unnecessary procedures. Moreover, deciding the bottleneck itself requires a lower bound value, which can be easily obtained from a comparative study. Thus, for an exact diagnosis of the control plane problem, a comparative analysis over multiple ISPs with a fine-grained comparison method is required.

Challenges in Fine-grained Analysis. There exist two technical challenges for an effective fine-grained comparative latency analysis: the difficulty in (1) analyzing the complicated control-plane operations and (2) considering the individual state information.

Identifying bottleneck points and the conditions that cause performance degradation requires an understanding of the standards (*i.e.*, RRC, or NAS) and operator-specific implementations. Note that the conformance-testing document has over 4,200 pages [4], while NAS and RRC standards are 500 and 700 pages, respectively [1, 3]. Therefore, examining the control plane messages and their diagnosis demands excessive engineering cost as well as domain-specific knowledge, which are arduous for the manual analysis process.

Comparative analysis requires exploiting state information including previous states, behaviors, and environmental conditions. An operator may have different latencies for performing the same control plane procedures because their previous procedures may affect the next procedure operations. Indeed, the ATTACH procedure following TAU failure shows different performance results when compared with ordinary ATTACH cases. Therefore, when comparing the two procedures, we should consider the previous state in order to avoid incorrect diagnosis results.

[1]These combinations result in 39 and 166 total cases in ATTACH and SR, respectively, in our dataset with our definition of the path in Section 4.2.

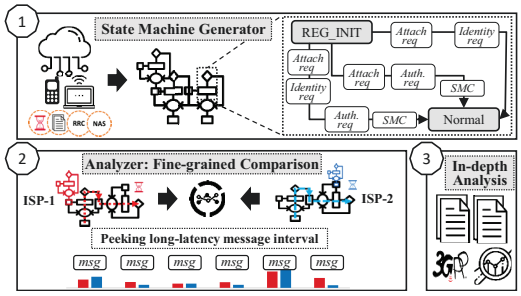


Figure 3: Overview of comparative study

4 DESIGN

The goal of our analysis is two-fold: (1) conducting a fine-grained latency analysis on a target operator's control plane with minimum manual effort, and (2) comparing the latencies across different network operators while considering operator-specific policies and diverse operational scenarios.

To address this goal, we propose an **automated trace-driven modeling** technique. Figure 3 shows the overview of our analysis in three steps: (1) We start by building an operator-specific state machine based on control plane signaling messages. A transition between two states becomes a sequence of signaling messages. Instead of having custom states in the state machine, we leverage the 3GPP standard states. Thus, state machines from different ISPs share some same states and transitions. (2) We then identify the shared transition-paths across different state machines. Each path represents a distinct operation scenario shared across different ISPs. (3) Finally, for each message interval of such a shared transition path, we compute the latency between two messages and identify the long-latency message interval (LMI) whose value is relatively larger than those of other operators. The identified LMIs are valuable information for operators, who are in a dire need of pinpointing latent bottlenecks that may undermine the overall service latency.

4.1 Constructing the State Machine

Trace-driven modeling. We implement a state machine generator that builds a state machine from given control plane signaling messages. For each ISP, we feed its signaling messages collected from the UE to the generator to produce an operator-specific state machine. Therefore, the generated state machines are based on message traces; thus, the model naturally reflects operator-specific configurations and implementations in handling control plane messages. We use diagnostic message monitoring tools to extract the control plane signaling message [6, 13]. They connect to the UE via USB and expose the Diagnostic Message (DM) logs that the UE chipset produces. We selectively collect the DM logs that generate notifications regarding the reception/transmission of control plane signaling messages. Note that each of these messages contains a precise timestamp of the reception/transmission of the message.

States. We define a state as a combination of the EMM state and EMM substate. Note that we leverage the states defined in the 3GPP specification [1]. Such standard states serve as anchoring points

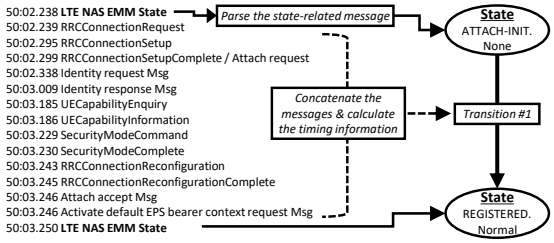


Figure 4: Example of state machine construction

between different state machines, which contributes to identifying comparable transitions across state machines. This also makes the state machine analyzable, as the specification already describes the status and operation of each state. It also reduces the manual effort for building a state machine because the individual messages from the UE chipsets provide explicit state information.

Transitions. We define a transition as a sequence of RRC and NAS messages between two states except broadcasting messages (*i.e.*, Paging). When a sequence of observed RRC and NAS messages causes the state to change from a source to a sink state, the sequence of such message types becomes a transition. Because different operation scenarios involve different sequences of messages, each transition reflects a specific scenario that often involves 1) the existence of a radio connection, 2) the identification, authentication, or security requirements of the MMEs or BSs.

Figure 4 shows an example of building a state machine. The left-hand side shows the diagnostic messages extracted while conducting an `ATTACH` procedure. Based on the extracted logs, we define two states based on the `LTE-NAS-EMM` log messages, which the 3GPP standards define and the DM logs explicitly show. Further, we define the transition between two states to be the sequence of messages seen in the DM logs. We also store the timestamp information to compute the elapsed time to make the state transition. Note that the state machine construction process needs only a UE with a target ISP without any help from the BS or the core network equipment.

4.2 State Machine Comparison Strategy

With the generated state machine for each operator, the analyzer detects the LMIs through comparison. Comparing the state machine is an intuitive process, in which the states in the model are useful delimiters and the transitions represent a comparison target.

Pathwise comparison. From each state machine, we enumerate the list of paths, each of which is a transition sequence from a source to a sink state. For the source states [2], we use the initial states of `ATTACH`, `TAU`, and `ServiceRequest` operations. We set a sink state to be the `REGISTERED.NORMAL` state, where a UE is ready to use its call service. Thus, each path is an overall sequence of signaling messages and states until a target procedure is completed. A path represents how a target procedure is performed by an operator, which includes a service scenario or a failure recovery case.

Observing LMI over the shared path. Among the identified paths from each state machine, we find the shared paths that have the same transition sequence [3]. We then compare the latency of each message interval in the shared paths across different state machines. Latency is calculated by the difference between the timestamp of an uplink message and the timestamp of its next downlink message. Note that the computed latency only measures the elapsed time in processing a control plane message at a target cellular network, not the latency caused by a UE. We further identify the long-latency message interval (LMI), if the latency of a certain message interval is relatively higher than those of other operators. Specifically, we define an LMI as occurring when (1) the latency of one operator exceeds twice that of another operator exhibiting minimum latency, and (2) the latency takes a more substantial portion than the sum of the latencies in the path divided by the total number of messages. We conservatively compared the latencies on the shared paths instead of defining the comparable paths between different state machines, which often demands operator-specific expertise. Comparing the non-shared path and devising a method to determine comparable paths will be performed in our future work.

[2]State format: [EMM state].[EMM substate] `ATTACH`: REGISTERED-INITIATED.None, `TAU`: TRACKING-AREA-UPDATING-INITIATED.None, `ServiceRequest`: SERVICE-REQUEST-INITIATED.None

[3]Two transitions are the same if the order & type of messages are the same.

Table 1: Summary of our dataset

	KR-1	KR-2	KR-3	US-1	US-2
Signaling Msg #	20106	44445	39060	97103	193528
Procedures #	746	2212	1498	5549	9115

Table 2: Statistics of Generated State Machines

Operator	KR-1	KR-2	KR-3	US-1	US-2	Total
Total transitions	464	375	386	332	301	1,352
Shared transitions	161	218	229	99	97	297

(a) Generated Transitions in State Machines

Target Procedure	KR-1	KR-2	KR-3	US-1	US-2
`ATTACH`	15 (3)	11 (6)	16 (6)	8 (4)	3 (2)
`TrackingAreaUpdate`	35 (3)	20 (12)	33 (11)	37 (7)	14 (7)
`ServiceRequest`	11 (7)	38 (16)	51 (17)	65 (15)	44 (11)

(b) Diverse Cases of Target Procedures: ***In each cell, A (B) denotes the following: A - # of transition paths; B - # of comparable cases***

5 COMPARATIVE ANALYSIS

We apply our analysis to five operators in order to demonstrate the effectiveness of our comparative analysis.

5.1 Dataset and Generated State Machine

Dataset. Table 1 provides a summary of our collected data, which includes 394,242 LTE signaling messages and 19,120 target procedures over five carriers. We collected the messages using diagnosis message monitoring tools [6, 13] and five types of UE (Galaxy S4/S5 and LG G2/G3/V10). We conducted various measures for collecting comprehensive signaling messages, such as turning on/off the airplane mode and data/voice service in stationary and mobility scenarios by physically visiting different cities in two countries. All data are either collected by us or extracted from the previous dataset [6]. For the KR dataset, we collect the traces from over 3,000 km movements via high-speed trains and cars between Seoul and Daejeon and driving all around South Korea. We also collected US dataset from several road trips [4] of visiting cities in US west as well as used the dataset from the prior work [6].

Generated state machine. We first demonstrate how the state machine effectively handles diverse scenarios for the fine-grained analysis. Table 2 shows the statistical results of the state machine for each operator. Our automated modeling differentiates 1,352 operational scenarios through transition representation. Surprisingly, only 297 transitions are shared between at least two operators (Table 2a), which implies that the remaining transitions are operated differently. We also observe that 323 scenarios exist across the three target procedures in all, through the pathwise comparison (Ta-

[4]We had to drive this far to collect signaling messages that generated from the scenarios such as moving another location or turning on the phone in a different region.

Table 3: Identified Long-latency Message Intervals (*An ISP with an LMIs is in bold*)

LMI-ID	Path	Uplink message	Downlink message	Appeared operator: ISP (average seconds)
LMI-1	ATTACH-1 [+]	ESM info. response	ueCapabilityEnquiry	**KR-2 (0.439)**, US-1 (0.171), KR-3 (0.124)
LMI-2	ATTACH-1	Attach request[*]	Auth. request	**US-1 (0.095)**, KR-2 (0.052), KR-3 (0.046)
LMI-3	ATTACH-2	Identity response	ESM info. request	**KR-1 (0.262)**, KR-3 (0.042), KR-2 (0.038)
LMI-4	ATTACH-3	NAS-SMC	ESM info. request	**US-2 (0.256)**, KR-3 (0.045), KR-2 (0.023)
LMI-5	ATTACH-3	ESM info. response	RRC-SMC	**US-2 (0.444)**, **KR-2 (0.259)**, KR-3 (0.110)
LMI-6	TAU-1	Identity response	EMM infomation	**KR-3 (0.130)**, KR-2 (0.065)

[*]The message is piggybacked with RRC connection setup complete message [+]The path consists of following messages: Attach request[*] - Auth. request - Auth. response - SMC (NAS) - ESM info. request - ESM info. response - ueCapabilityEnquiry - ueCapabilityInfo. - SMC (RRC) - rrcConnectionReconfig. - rrcConnectionReconfig.Complete - Attach accept - Activate default EPS bearer context request

ble 2b). Each operator has diverse yet unique operation scenarios. The results imply that each operator handles the procedures differently, and that this diversity stems from the operator-specific logic and configurations. These operator-specific logics are also reflected in the frequency of each path. For example, if an ISP turns on a re-authentication option, which always invokes an authentication procedure during the ATTACH procedure, the state machine contains the path containing the authentication request/response messages in the ATTACH procedure. In addition, the path appears more frequently in the state machine from the ISP than that of other ISPs who do not adopt the option [5]. We also compare each state machine from the trace with the one described in the 3GPP standard for completeness. Note that a direct comparison is difficult, because the state diagram of the 3GPP standard does not contain sub-states. Nevertheless, we confirm that all state machines generated from the trace hold the main states and sub-state for our targeted procedures except the states related to the failure states and transitions [6].

5.2 Lessons from Identified LMIs

We have observed 38 LMIs over the shared paths of three target procedures. Table 3 shows the selected LMIs in the shared paths [7]. The first column shows the list of message intervals; each exists in the shared path in the second column. The latency for each message interval is measured between the uplink and downlink messages. The last column shows operators with average latency over the collected observations. We find two implications from the identified LMIs, demonstrating the effectiveness of our approach.

First, every operator has an LMI, and no operator shows the lowest latency at every path. Also, each LMI occurs at different ISPs within the same control procedure according to the path. Table 3 shows that LMI-1 in ATTACH-1 is identified only at KR-2, whereas LMI-3 in ATTACH-2 is identified at KR-1. This implies that KR-2 is not optimized for the ATTACH-1 case, but exhibits low latencies for other cases. From those observations, we conclude that each operator is optimized differently and not fully optimized for all paths.

Second, an LMI exists even in paths having total latency lower than that of other operators. For example, the total latency of ATTACH-1 in US-1 is lower than that of KR-2, but US-1 contains LMI-2 (illustrated in Figure 5). This shows that comparing the total latency of a path alone across operators could result in failure to pinpoint long-latency intervals and their causes. Note that our prior work [6], which only considers the total latency of a target control plane procedure, fails to detect this case. Thus, the comparison requires: (a) consideration of diverse operation scenarios and (b) comparison of latency at the message-level. Our analysis achieves both of the requirements and effectively demystifies hidden performance degradation points.

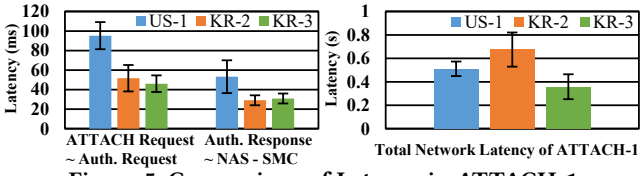


Figure 5: Comparison of Latency in ATTACH-1

5.3 Preliminary In-depth Study

In-depth study of LMIs. Table 3 shows that KR-2 has LMIs 1 and 5, both of which involve handling ESM information response messages. Considering that the ESM information response message in these paths (ATTACH-1,3) is handled by the operations between MMEs and GWs, we conclude that KR-2 has more room for optimization of the operations at two entities. For the remaining LMIs, we can adopt the same analysis approach, investigating the internal operation between the participating entities.

As Table 3 shows, US-1 has LMI-2 in ATTACH-1, performing the ATTACH procedure. Its latency is almost two-times larger than the others. First of all, we can conclude that the operation between MME and HSS of US-1 [8], handling the messages of the LMI, has more overhead than the others. Also, we further investigate all the uplink and downlink messages for LMI-2 by comparing them with those of other ISPs. We have found that only US-1 used encrypted messages for authentication in ATTACH-1, while the others do not cipher the authentication messages. Note that, ciphering the authentication messages is unnecessary in the context of the ATTACH procedure. The authentication procedure is designed to be performed without a security context, and a new security context is generated between the UE and MME after the authentication. We confirm that the encryption over these messages is unnecessary by checking the 3GPP standards [1, 2, 4] [9].

In-depth study of shared path. We have run the further analysis on the observed shared and non-shared paths. One interesting observation is that KR-1 has no shared paths which have the authentication logic. Authentication is a key step in generating a key (K_{ASME}), used for deriving the additional keys for ensuring the confidentiality and integrity of further control messages. This means that the key renewal policy of KR-1 is unique compared with the others, so that KR-1 reuses the pre-established session information aggressively by skipping the authentication procedure. This operational difference mainly stems from the ambiguity in the 3GPP standards, which do not specify the condition for the re-authentication procedure. Thus, it is highly dependent on the operating policy. Finally, this security implication of skipping the authentication deserves further study.

[5]We confirmed the existence of the option by interviewing one MME manufacturer.
[6]To identify these states, an active tester that transmits the manipulated control plane message to the cellular network is required. (We will discuss this in Section 6)
[7]Due to space limitations, we have shown only 6 LMIs in 4 paths.
[8]More specifically, the entities in charge of authentication vector.
[9]Clause 4.4.2.4 in the NAS spec mentions this case. However, it does not describe any underlying reasons for the encryption, and no conformance test case covers this issue.

6 DISCUSSION

Potential Directions: State machine representation facilitates the efficient analysis of the operating logic of the control plane on a cellular network. An interesting example involves analyzing the failure recovery logic, which is highly dependent on the operators and manufacturers, and its faulty design significantly affects the user experience. Promisingly, state machine representation makes it easy to extract and compare the failure recovery logic of each operator. Any path traversing the DEREGISTERED states represents a failure recovery operation scenario. We conducted the comparative latency analysis on such paths, which includes the failure recovery of TAU REJECT. A notable observation is that handling ATTACH REQUEST after TAU REJECT consistently takes 5 or 10s in US-1. Hong *et al.* observed the same issue with significant manual analysis [6], which demonstrates the effectiveness of our state machine representation.

Our approach still has room for improvement in the construction and use of the fine-grained state machine. We believe that the state machine representation considering more information such as the location of collected data, user action, and core equipment vendors, will aid in root cause analysis. Moreover, to improve the accuracy of the diagnosis, the pathwise comparison needs to select the paths exclusively by selecting the source state carefully. The analysis of the non-shared paths is also applicable to the identification of security problems. For example, a path heading toward failure states or time-consuming states could be an effective attack vector. Lastly, our approach, representing the operational scenarios as the path, could be applicable to the stateful fuzzing by executing dynamic testing at each path to discover the potential vulnerabilities.

Limitations of Trace-driven Approach: While the trace-driven approach is effective in reflecting the operating logic, the completeness of the model heavily depends on the collected traces. We believe that a larger dataset obtained through crowd-sourcing would address this limitation and reveal new findings. One solution would be to leverage an active data collector by exploiting the software-defined-radio and open source LTE stack [12]. Our approach relies on the observation of interacting control plane messages at the UE. Unfortunately, the UE has limited access to network-side operations, and finding the root cause of a problem demands a comprehensive understanding of cellular network specifications. Moreover, our approach may not identify an operator-specific configuration that does not produce the signaling messages. However, if such a configuration contains different contents in the signaling message, the fine-grained comparison of the content in the signaling messages may be able to resolve such cases. We emphasize that our work provides a starting point for investigations, which effectively reduces the effort for finding or understanding the cause of a bottleneck.

Related Work: Diagnosing the performance degradation problem in cellular networks has been extensively conducted in control plane [6, 7, 16], services [9], and radio access networks [8]. CNetVerifier [16] and LTEInspector [7] constructed models from the 3GPP specs, but these require extensive efforts to convert the natural language of the specs to the state machine and fails to reflect operating control plane logic. Similar to our work, RILAnalyzer [17] and MobileInsight [10] employ trace-driven modeling. However, RILAnalyzer exploits a probability of the state transitions, based only on the coarse-grained information from the 3G RRC protocol. MobileInsight presents a tool that provides fine-grained information on control plane messages, but it does not provide a way of (a) constructing a model of control plane operating logic, or (b) comparing the models in fine-grained fashion for the latency analysis.

7 CONCLUSION

We presented a novel diagnosis method for comparative analysis across multiple network operators, which is expected to be useful for network operators and manufacturers of cellular equipment. Our approach automates modeling from the trace to reflect the real operating logic, allowing for avoidance of the exhaustive process of standard document analysis. We also conducted a pathwise analysis for the fine-grained comparison over five major operators in two countries and identified 38 LMIs, which deserves further investigations. Our work is the first comparative latency analysis with fine-granularity information with promising results. Our state machine models are also applicable to the identification of problems in other domains including security and operation logic errors, which also deserves further research.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers and our shepherd, Alastair Beresford, for their insightful comments and suggestions for improving the paper. This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (2018-0-00831, A Study on Physical Layer Security for Heterogeneous Wireless Network). We would like to thank the authors of SCAT for the collection of data.

REFERENCES

[1] 3GPP. TS 24.301. Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3, 2017.
[2] 3GPP. TS 33.401. 3GPP System Architecture Evolution (SAE); Security architecture, 2018.
[3] 3GPP. TS 36.331. Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification, 2017.
[4] 3GPP. TS 36.523. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Packet Core (EPC); User Equipment (UE) conformance specification;, 2018.
[5] A. Banerjee, R. Mahindra, K. Sundaresan, S. Kasera, K. V. der Merwe, and S. Rangarajan. Scaling the LTE Control-plane for Future Mobile Access. In *CoNEXT*. ACM, 2015.
[6] B. Hong, S. Park, H. Kim, D. Kim, H. Hong, H. Choi, J. P. Seifert, S.-J. Lee, and Y. Kim. Peeking over the Cellular Walled Gardens-A Method for Closed Network Diagnosis. *IEEE TMC*, 17(10), 2018.
[7] S. R. Hussain, O. Chowdhury, S. Mehnaz, and E. Bertino. LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE. In *NDSS*, 2018.
[8] A. P. Iyer, L. E. Li, and I. Stoica. Automating Diagnosis of Cellular Radio Access Network Problems. In *MobiCom*. ACM, 2017.
[9] Y. J. Jia, Q. A. Chen, Z. M. Mao, J. Hui, K. Sontinei, A. Yoon, S. Kwong, and K. Lau. Performance Characterization and Call Reliability Diagnosis Support for Voice over LTE. In *MobiCom*. ACM, 2015.
[10] Y. Li, C. Peng, Z. Yuan, J. Li, H. Deng, and T. Wang. Mobileinsight: Extracting and Analyzing Cellular Network Information on Smartphones. In *MobiCom*. ACM, 2016.
[11] Y. Li, Z. Yuan, and C. Peng. A Control-Plane Perspective on Reducing Data Access Latency in LTE Networks. In *MobiCom*. ACM, 2017.
[12] openLTE. http://openlte.sourceforge.net.
[13] QXDM. qualcomm-extensible-diagnostic-monitor.
[14] A. S. Rajan, S. Gobriel, C. Maciocco, K. B. Ramia, S. Kapury, A. Singhy, J. Ermanz, V. Gopalakrishnanz, and R. Janaz. Understanding the bottlenecks in virtualizing cellular core network functions. In *LANMAN*. IEEE, 2015.
[15] M. T. Raza, D. Kim, K.-H. Kim, S. Lu, and M. Gerla. Rethinking LTE network functions virtualization. In *ICNP*. IEEE, 2017.
[16] G.-H. Tu, Y. Li, C. Peng, C.-Y. Li, H. Wang, and S. Lu. Control-plane Protocol Interactions in Cellular Networks. In *SIGCOMM*. ACM, 2014.
[17] N. Vallina-Rodriguez, A. Auçinas, M. Almeida, Y. Grunenberger, K. Papagiannaki, and J. Crowcroft. RILAnalyzer: A Comprehensive 3G Monitor on Your Phone. In *IMC*. ACM, 2013.