REINFORCEMENT LEARNING BASED ADAPTIVE ACCESS CLASS BARRING FOR RAN SLICING IN 5G NETWORKS

5G AĞLARINDA RAN DİLİMLEME İÇİN PEKİŞTİRMELİ ÖĞRENME TABANLI UYARLANABİLİR ERİŞIM SINIFI ENGELLEMESİ

ABSTRACT

REINFORCEMENT LEARNING BASED ADAPTIVE ACCESS CLASS BARRING FOR RAN SLICING IN 5G NETWORKS

Master of Science, **Computer Engineering Department**

Machine-to-Machine (M2M) communication is one of the major drivers of 5G networks as M2M traffic might soon surpass Human-to-Human (H2H) traffic. Network slicing is a promising technique for supporting M2M traffic on 5G networks as there is a need to concurrently support varying Quality-of-Service (QoS) requirements of M2M devices. A major bottleneck for M2M traffic is the Random Access Channel (RACH) procedure, which has to be performed for all devices, which results in the same latency for all service types. Due to the event-driven simultaneous access behavior of M2M devices, this procedure can cause severe congestion. Legacy congestion control schemes such as Access Class Barring (ACB) are not adequate to handle the overload in bursty traffic scenarios, which can happen frequently in M2M communications. There is also no clear guideline to adjust ACB parameters dynamically in such situations. Here we propose a multi-rate ACB algorithm using Reinforcement Learning (RL) to tune the barring rates and barring times of different service classes. Our priority-based algorithm not only reduces the congestion but also slices the RACH among different service types. Comprehensive simulation results show that our proposed algorithm maximizes the RACH utilization. In the meantime, based on each service priority, it reduces the delays and increases the access probability even when the connection requests exceed the RACH capacity.

Keywords: Reinforcement Learning (RL), RAN Slicing, Massive Machine Type Communication, Congestion Control, Access Class Barring (ACB)

ÖZET

5G AĞLARINDA RAN DİLİMLEME İÇİN PEKİŞTİRMELİ ÖĞRENME TABANLI UYARLANABİLİR ERİŞIM SINIFI ENGELLEMESİ

Yüksek Lisans, **Bilgisayar Mühendisliği**

Makineler arası (M2M) iletişimi etkin bir şekilde sağlamak 5G ağlarının öne çıkan özelliklerinden biridir. Makineler arası trafiğinin yakın zamanda insandan insana (H2H) trafiğini geçebileceği ön görülmektedir. Ağ dilimleme, makineler arası trafiğini karşılamak için ve aynı zamanda bu cihazların değişen hizmet kalitesi (QoS) gereksinimlerini destekleme ihtiyacı olduğundan umut verici bir tekniktir. Tüm hizmet türleri için aynı gecikmeyle sonuçlanan ve tüm cihazlar için gerçekleştirilmesi gereken rastgele erişim kanalı (RACH) prosedürü, M2M trafiği için muazzam bir darboğazdır. Makineler arası cihazlarının olaya dayalı eşzamanlı erişim davranışı nedeniyle, bu prosedür ciddi tıkanıklığa ve gecikmelere neden olabilir. Erişim sınıfı engelleme (ACB) gibi eski tıkanıklık kontrol şemaları makineler arası iletişimlerde sıklıkla meydana gelebilen, yoğun trafik senaryolarında aşırı yükün üstesinden gelmek için yeterli değillerdir. Ayrıca bu durumlarda ACB parametrelerini dinamik olarak ayarlamak için net bir kılavuz bulunmamaktadır. Biz, bu çalışmada farklı hizmet sınıflarının engelleme oranlarını ve engelleme sürelerin ayarlamak için pekiştirmeli öğrenme (RL) tabanlı, çok oranlı bir ACB algoritması öneriyoruz. Önceliğe dayalı algoritmamız sadece tıkanıklığı önlemiyor aynı zamanda RACH kullanımını farklı hizmet türleri arasında dilimliyor. Yapmış olduğumuz kapsamlı simülasyonların sonuçları, önerilen algoritmamızın RACH kullanımını maksimize ettiğini göstermektedir.

Bununla birlikte, önerilen algoritmamız hizmet önceliğine göre bağlantı isteklerinin RACH kapasitesini aştığı durumlarda gecikmeler azaltmakta ve erişim olasılığının arttırmaktadır.

Keywords: Pekiştirmeli Öğrenme (RL), RAN Dilimleme ,Geniş Ölçekli Nesnelerin Interneti, Tıkanıklık Kontrolü, Erişim Sınıfı Engellemesi (ACB)

CONTENTS

Page

TABLES

FIGURES

ABBREVIATIONS

3GPP	:	**3**rd **G**eneration **P**artnership **P**roject
ACB	:	**A**ccess **C**lass **B**arring
ACs	:	**A**ccess **C**lasses
DRL	:	**D**eep **R**einforcement **L**earning
eNB	:	**e**Node **B**
ETSI	:	**E**uropean **T**elecommunications **S**tandards **I**nstitute
H2H	:	**H**uman-**t**o-**H**uman
ITC	:	**I**nternational **T**elecommunications **U**nion
KPI	:	**K**ey **P**erformance **I**ndicators
M2M	:	**M**achine-**t**o-**M**achine
mMTC	:	**m**assive **M**achine **T**ype **C**ommunications
MTC	:	**M**achine **T**ype **C**ommunication
PBRA	:	**P**riority **B**ased **R**andom **A**ccess
PTR	:	**P**reamble **T**ransmission **R**ate
QoS	:	**Q**uality-**o**f-**S**ervice
RA	:	**R**andom **A**ccess
RACH	:	**R**andom **A**ccess **C**hannel
RAN	:	**R**adio **A**ccess **N**etwork
RAO	:	**R**andom **A**ccess **O**pportunity
RAR	:	**R**andom **A**ccess **R**esponse
RL	:	**R**einforcement **L**earning
SIB2	:	**S**ystem **I**nformation **B**lock **T**ype **2**
TIA	:	**T**elecommunications **I**ndustry **A**ssociation
UE	:	**U**ser **E**quipment
URLLC	:	**U**ltra **R**eliable **L**ow **L**atency **C**ommunications

1. INTRODUCTION

From tigers' stripes to the hexagons that make up honeycombs to the ripples in windblown sand, the natural world is full of order and regularity [2]. Orderliness is an integral part of the world and life which sometimes surprises and sometimes makes humankind think. Even most scriptures point out a natural order and rhythm in the universe and encourage humanity toward the discipline. The orderliness that surrounds us underpins all of our knowledge. From Mathematica to Astronomy, from Physics to Music tell us about the many aspects of orderliness.

Natural or artificial, orderliness lies in the various layer of the world and each aim to provide different goals. Engineering also does prefer to perform any task or operation based on predefined order. For instance, to increase the CPU performance, a comprehensive pipeline has been considered. When it comes to traffic and chaos, order and regulations are the well-known solutions.

New generation cellular networks are meant to deliver higher data transmission speed, ultra-low latency, more reliability and massive network capacity. They aim to provide connectivity for everything, such as machines, objects, devices and vehicles, in coexistence with H2H communications. Besides, extended coverage, security, robust management and lower deployment costs are some of the main advantages and encouragements for preferring this technology in machine type communication. The number of IoT connections is predicted to reach around 27 billion by 2026 [3]. Therefore, there will be a severe challenge to handle the massive number of connections generated by an enormous number of M2M devices. Physical channel resources in the Random Access (RA) procedure have their fundamental limitations, which might result in extreme congestions and chaos.

In another perspective, different services that will be offered in this architecture will have various and variable QoS requirements based on their acceptable delay budget and drop rate. These QoS requirements may vary from ultra-reliable low latency communications to high transmission data rates. Although these requirements have been accommodated

in the data transmission phase, the RACH procedure is not currently suitable for service differentiation. Slicing the RACH resources besides controlling the congestion is essential to meet the desired service requirements.

There are a few congestion control approaches in the literature for the RACH. ACB is one of the accomplished and common ones which is suggested in 3GPP specifications. To alleviate the congestion, ACB redistributes the network access requests of User Equipment (UE) over time. For this purpose, it broadcasts both Barring-Rate and Barring-Time parameters. Yet, there are no manifest instructions to adjust these parameters under highly dynamic scenarios, which is most likely to happen considering the event-driven bursty traffic raised by M2M communications.

There have been several works in the literature for dealing with the congestion issue in the connection establishment process over the RACH. The performance evaluation of the ACB in [4] shows the suitability of this scheme for M2M applications. ACB method parameter optimization is another active research area in [5], [6], [7] and [8]. Although priority requirements have not been concerned, these studies have used different approaches like backlog size estimation, Q-learning, or deep RL to achieve a dynamic ACB parameter adjustment. On the other hand, there have been attempts to implement service differentiation without using artificial intelligent approaches [9]. The work in [10] focuses on the prioritized RA procedure, in which the IoT devices are separated by groups with different priorities using a novel pricing scheme. Chen and Smith [11] present a DQN-based MAC mechanism for managing channel access of heterogeneous radios, in which each radio requires a different QoS. They have defined their RL reward function based on fixed acceptable delay thresholds for each service. However, reasonable thresholds can change due to the channel state. Therefore, to achieve better results, these thresholds should be chosen dynamically.

Some of these works offer complex procedures, use questionable assumptions for getting high performance, or do not conform with LTE-A specifications. Also, in some of these studies, the retry limits have not been considered. In almost all earlier works, the authors

only test scenarios that are below the RACH capacity, which traffic control is not required in these scenarios.

1.1. Scope Of The Thesis

This thesis mainly focuses on congestion control in the course of random access procedures, especially in massive type communications. We start by surveying the RACH capacity under the suggested network configuration in the 3GPP specification [12]. Then we continue by determining congestion thresholds under various traffic models. We study the ACB capability to control these congestions, even in bursty traffic scenarios. Then, we implement reinforcement learning to adjust the ACB parameters intelligently, dynamically and continuously. In the next step, we try multiple reward functions to achieve maximum RACH utilization. The most important stage is implementing the multi-rate mechanism powered by the RL agent. Our study continues by adjusting the barring time parameter. Finally, we compare our results with the theoretical optimum approach and previous works to evaluate the performance.

1.2. Contributions

In this research, we cover these deficiencies by proposing a novel, simple and efficient approach. The main contributions of this paper can be summarized as follows:

- We propose a dynamic multi-rate ACB scheme tuned by RL. Our novel reward function makes it possible to use the RACH effectively while considering the QoS priorities.
- Unlike most of the previous works, we consider scenarios significantly exceeding channel capacity and show the effectiveness of the proposed algorithm under heavy load.

- For the first time, we attempt to tune the barring time parameter with constant or dynamic schemes. We use our multi-rate RL-based agent to adjust the barring time dynamically.
- We introduce a perfect control method to compare our proposed approach.
- Our simulation results show that the proposed technique approaches the perfect-control outcomes, confirming that the proposed scheme not only maximizes the RACH utilization but also minimizes the delay for higher priority classes through its slicing feature.

1.3. Organization

The organization of the thesis is as follows:

- Chapter 1 presents our motivation, contributions and the scope of the thesis.
- Chapter 2 provides the general background about random access procedures and access class barring.
- Chapter 3 gives a brief overview of previous works and related studies on congestion issues over the RACH and different approaches to overcoming this problem.
- Chapter 4 introduces our novel RL-based mult-rate approach in detail.
- Chapter 5 demonstrates achieved simulation results and evaluates the performance of our proposed method in detail, considering various KPIs.
- Chapter 6 states the summary of the thesis and possible future directions.

2. BACKGROUND OVERVIEW

2.1. Random Access Channel

In LTE-A networks, data transmissions require a connection established by UE through the RA procedure. For high priority and regular equipment, two different modes, contention-free and contention-based, are operated in this procedure. Performing a RA procedure is essential under any of these situations [1]:

- to acquire initial access to the network.
- to re-establish the connection after the failure of a radio link.
- to hand over from one eNB to another.
- to update the user equipment location.
- to make scheduling requests.

In this study, our focus is on contention-based mode, which consists of the following four steps in figure 2.1:

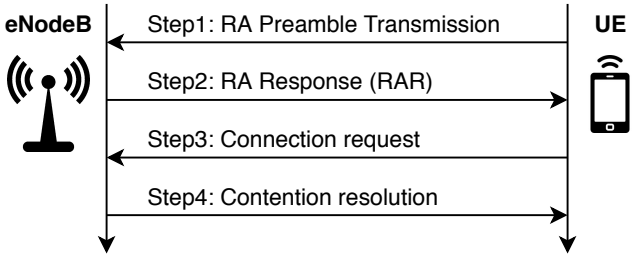

Figure 2.1 Random Access Procedure

1. RA preamble transmission: Each UE transmits a preamble as an access request to the eNodeB(eNB) for a dedicated time or frequency resource block in the upcoming RA Opportunity (RAO). This preamble is selected randomly from a pool of up to 64 orthogonal preambles known to both UEs and eNBs.

2. RA Response (RAR): The eNB acknowledges the successfully detected preambles by the identification of the related preamble and uplink grant for the next step.

3. Connection request: UE requests a connection with its ID using the uplink resource guaranteed in the corresponding received RAR. If multiple UEs select the same preamble in the first step, they would each receive the RAR, so all will be granted the same resource block for the connection request, which will result in a collision in the multiple selected preambles.

4. Contention resolution: The eNB broadcasts contention resolution message including the ID of related UE. Then eNB allocates the required data resources for UE. UE restarts the procedure if it does not receive a response to a preamble or a request. Each UE repeats this procedure until establishing a connection or reaching the maximum number of preambles retransmissions.

The RA procedure ends up successfully for a UE when the eNB can detect the received connection request properly, and it is possible if and only if the preamble is selected by just one UE in an RAO.

2.1.1. Backoff Procedure

In case of failure during the RACH procedure due to collisions or insufficient transmission power, the UE must perform a backoff step before re-transmit a new chosen preamble in the next RAO. 3GPP recommends this method to improve the success chance of establishing a connection [13]. In order to reduce the collision, UEs chose the backoff times T_{BO} randomly based on the LTE-A standard as follows.

$$T_{\text{BO}} = U(0, B_i), B_i \in [0, 960]\, ms \tag{1}$$

where $U(.)$ stands for uniform distribution, and B_i is the backoff indicator broadcasted by the eNodeB in the RA Response. The RA Response is read by all UEs which transmit a

preamble in the previous RAO. There is also an exponential backoff scheme to deal with the congestion problem [10].

2.2. Access Class Barring

ACB is one of the well-known solutions for the congestion problem in the RACH. 3GPP introduced this control method to alleviate RACH pressure by limiting the maximum number of UEs that simultaneously access the eNB. Sixteen different Access Classes (ACs) 0 to 15 are defined based on service requirements. Each UE is categorized into one out of the first 10 ACs (0 to 9). Also, it can be assigned to some of the five special categories (11 to 15). In this approach, the eNB broadcasts a Barring Rate $P_{\text{ACB}} \in \{0.05, 0.1, ..., 0.95\}$ and Barring time $T_{\text{ACB}} \in \{4, 8, 16, ..., 512s\}$ through System Information Block Type 2 (SIB2) for the upcoming RAO.

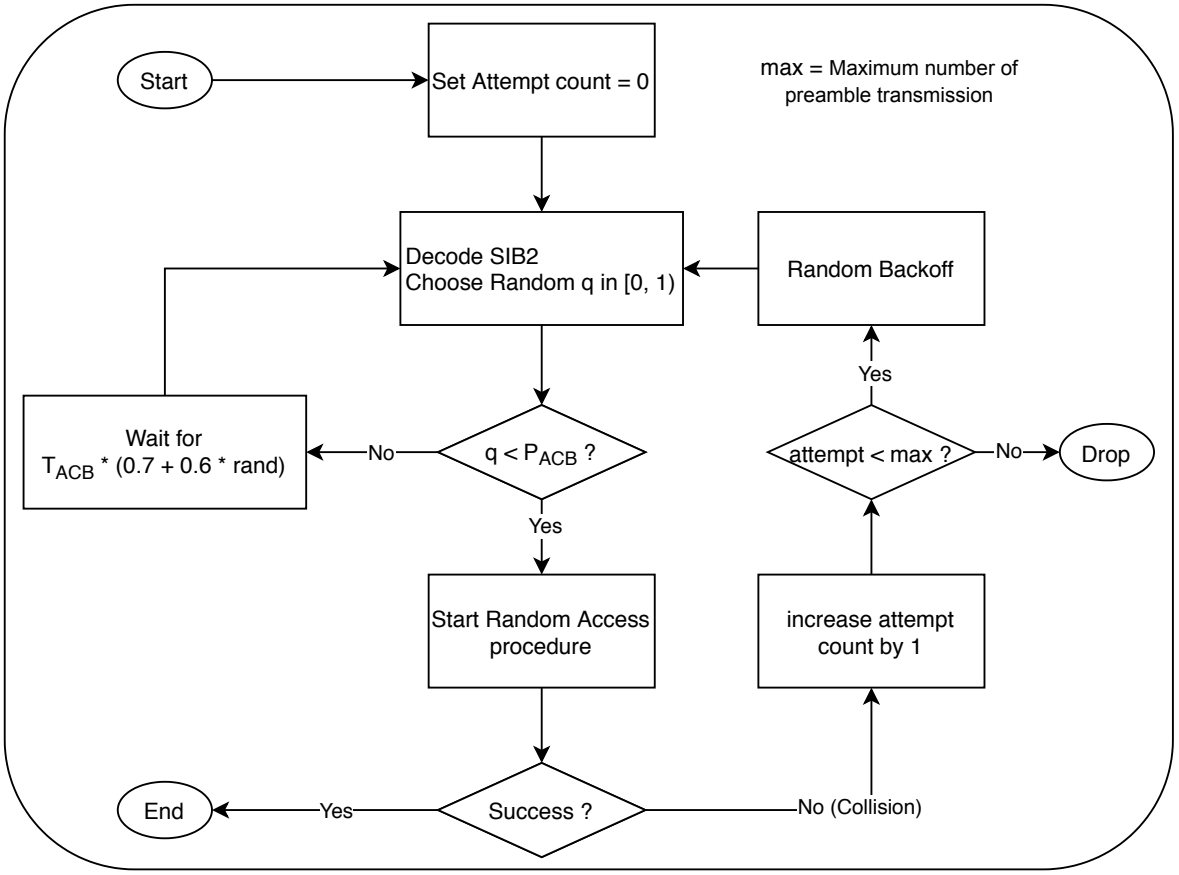

Figure 2.2 Access Class Barring flow chart

Barring factors are commonly applied to ACs 0-9, while the special categories are exempted from the barring process. In the first step of the RACH procedure, each UE generates a random number q, between 0 and 1. If q is less than P_{ACB} , then a random preamble will be selected, and the remaining steps will carry on. Otherwise, the UE is barred temporarily for a random time calculated as follows:

$$T_{\text{barring}} = (0.7 + 0.6 \times \text{ rand }) \times T_{\text{ACB}} \text{ , rand } \in [0, 1] \tag{2}$$

Figure 2.2 visually demonstrates the ACB flowchart.

This process is repeated until the UE generates a random number lower than P_{ACB} and sends its preamble. In this way, ACB reduces the number of access requests per RAO.

3. RELATED WORK

With the spread of machine-to-machine communication on cellular networks, numerous leading standardization organisms have stepped in and updated their standards toward MTC requirements. International Telecommunications Union (ITU), the European Telecommunications Standards Institute (ETSI), 3GPP and the Telecommunications Industry Association (TIA) have announced standards and protocols for insuring coexistence fo M2M and H2H communications.

Several studies have also been conducted on congestion and overload solutions for M2M applications in wireless communication. The number of these studies is so large that various surveys have been done in this context. Laya et al. investigate the suitability of random access channels for MTC by surveying current and alternative solutions [14] They compare multiple approaches based on four main KPIs, including access success probability, preamble collision rate, access delay and device energy consumption. They also categorize different methods under the main eight categories, which the access class barring scheme is one of them.

Althumali and Othman survey M2M communications over LTE networks, covering M2M architectures, LTE structure, deployment challenges, and access control requirements on their survey [15]. They divided random access techniques into three main categories. They made a comparison of different methods based on various KPIs.

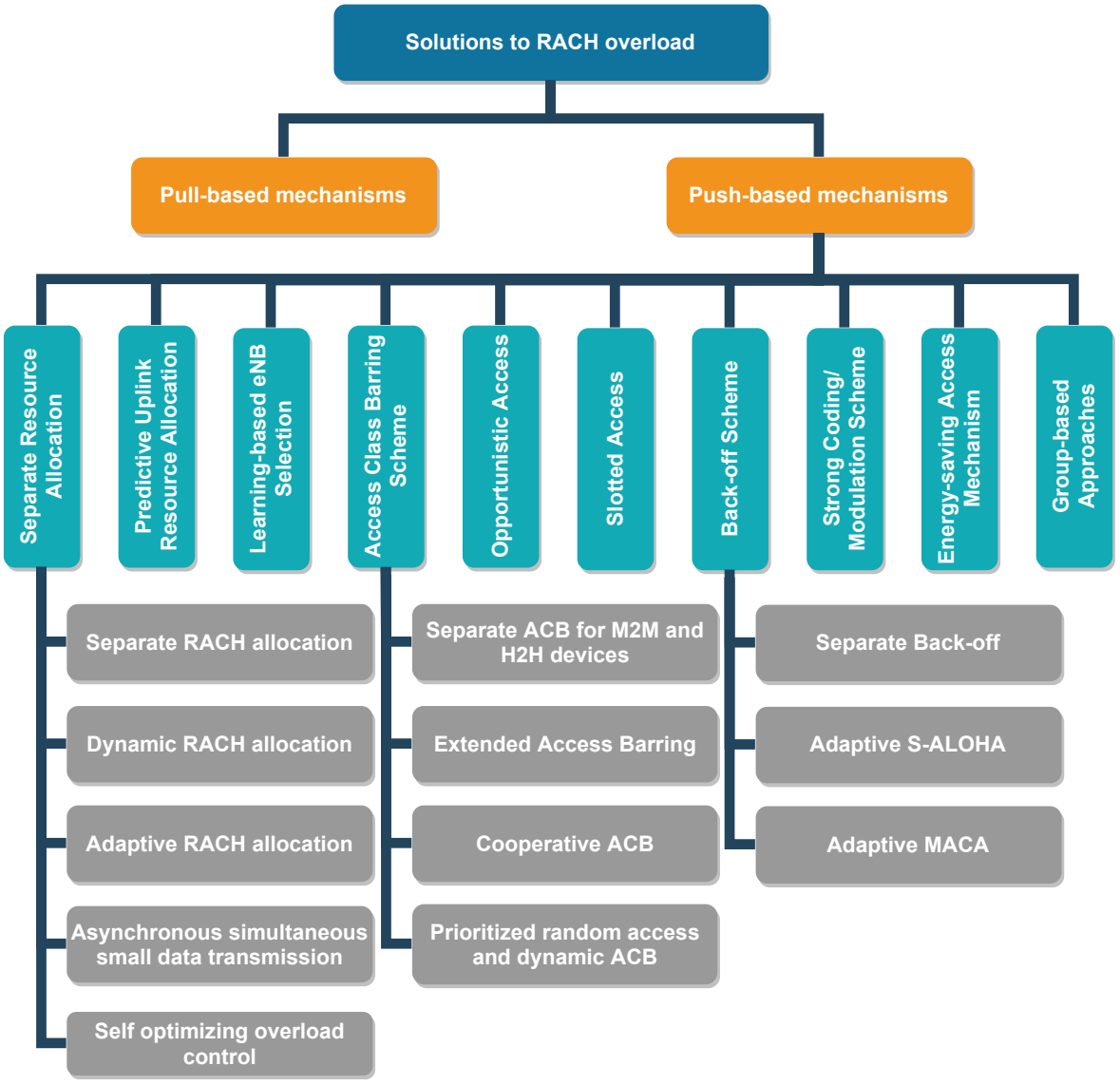

Figure 3.1 Classification of different mechanisms suggested to solve the access traffic by [1]

The severity of traffic issues raised by MTC made Soltanmohammadi et al. perform a comprehensive survey in this respect [1]. They have looked at the different aspects of problems, proposed solutions, pros and cons of each method, and left challenges that need

to be addressed in order to empower and enhance MTC. Their congestion control solutions classification is provided in figure 3.1

In this section, we cover some of these categories and give some examples of each kind. However, most of the covered studies can be characterized under estimation-based or RL-based approaches.

3.1. Load Estimation Based Approaches

The optimal barring rate of ACB can be determined if the competitive UE count is known. Based on this fact, Tavana et al. defined and derived a function called conditional probability distribution function (PDF) to reduce the overall service time [16]. They used both slot information and the system dynamics to predict the number of backlogged MTC devices within their scheme. To regulate the ACB factor adaptively, they contributed a maximum likelihood estimator by using the PDF. Lastly, to enhance the accuracy of prediction, they used the Kalman filter. The performance of their method further gave optimal results since it was observed that the results were quite close to the optimal situation.

From an innovative perspective, some studies aim to estimate the number of contending UEs in each RAO (named Preamble Transmission Rate (PTR) herein). Tello-Oquendo et al. [17] designed a series of maximum likelihood and Bayesian estimators for this purpose. Guan et al.[18] propose a Priority-Based Random Access (PBRA) approach to provide service differentiation over the RACH. They classified the UEs into the three different priorities inspiring from access delay requirements suggested in [19] and [9]. Then they split the preambles and assigned each group to the declared priority classes. Using a primitive load estimator and a classic ACB mechanism, they have tried to provide different QoS in the RACH. It is noticeable that they have applied ACB to only UEs of the low priority class. Assigning a larger slice of the preamble for higher priority classes reduces the RACH utilization. Maximizing the RACH utilization and handling the congestions are the main missing issues with this approach. Besides, PBRA is not applicable for a various number of priority classes.

Bui et al. underline the lack of prioritization in baseline ACB for urgent traffic during the RACH procedure [20]. They recommend using Distributed Queueing (DQ) with a novel load estimator method for MAC-layer. Furthermore, they offer dynamic access prioritizing mechanism based on information from the DQ process about congestion levels.

Another estimation-based approach is called the adaptive attractor-selection-based congestion control scheme, which lies on resource separation [21]. The ASCC technique calculates the base station–selection probability to be selected using the predicted load value and available preamble sequences.

3.2. RL Based Approaches

With the proliferation of artificial intelligence, particularly reinforcement learning, and the success of these methods in handling complicated issues, RL has made a quick entry into the field of communication. RL is initially used for resource management and various resource slicing. We survey the most thriving applications of this approach in this subdivision.

Raza et al. propose an RL-based slice admission strategy to reduce the penalty a RAN infrastructure provider faces while providing services with varying latency requirements [22]. One of the recent researches in this area is [23], which offers a combined energy-efficient subchannel assignment and power management strategy. They use a multi-agent RL to tackle the massive access management challenge considering energy efficiency. To make the optimization issue, they turned the delay requirement into a data rate constraint. For heterogeneous wireless networks, Yu et al. introduce DLMA, a universal MAC protocol based on DRL [24]. Nassar and Yilmaz suggest a network slicing model based on a cluster of fog nodes to utilize the restricted resources at the network edge optimally [25]. In order to learn the optimal slicing policy adaptively, A DRL beside a Markov decision process for problem formulation is used. They target to achieve efficient resource allocation using grade of service as a key performance indicator in their study.

Another RL-based RAN slicing approach is suggested by Raza et al [26]. In the presence of services with varying priorities, they offer a slice admission method based on RL. A 5G

flexible radio access network (RAN) is being proposed, in which slices of multiple mobile service providers are virtualized over the same RAN infrastructure.

A Q-learning-based method is proposed by El-Hameed and Elsayed to take care of the congestion issues over the RACH [27]. Their main target is to ensure that the H2H communications and its QoS requirement are not affected by the coexistence of MTC. Their approach assigns the available preambles between M2M and H2H devices such that the H2H devices receive adequate service while the number of active M2M devices is maximized. Their reward function is based on actual calculated blocking probability and blocking probability threshold value. Based on this reward value, the RL agent changes the number of assigned preambles for M2M.

They proposed Q-learning-based uplink resource allocation techniques in work [28] to optimize the number of serviced IoT devices in NB-IoT networks in real-time. For the single-parameter single-group situation, they initially created tabular-Q, LA-Q, and DQN-based methods, which exceed the traditional LE-URC and FSI-URC approaches in terms of the number of serviced IoT devices. They studied the multi-parameter multi-group scenario provided in the NB-IoT standard to handle traffic with various coverage needs, which presented the high-dimensional configurations challenge.

3.2.1. ACB Parameters Optimization Using RL

One of the most suitable applications of RL is ACB parameters optimizations to improve the RACH performance. General targets of all these approaches are increasing the success probability and reducing the delay. In this subsection, we try to cover each research perspective and also its advantages and disadvantages. One of the pioneers in this regard is Chen et al. In the first phase of their study, authors in [11] propose a novel DRL algorithm to tune the barring rate factor dynamically. In the second phase, they enhance their algorithm to manage RACH for heterogeneous machine-type communication devices requiring various levels of quality of service. The details of this study also are discussed in section 5.2.1.

Authors in [29] have used the RL to cope with the congestion problem arising from MTC. They have targeted to minimize both the collision rate and the access delay in their study. Their Q-learning algorithm makes use of a reward function based on the collision rate and the delay in the previous RAO. Although they reduce the collision rate however the low success rates are striking at first glance.

For tuning ACB parameters, Jiang et al. proposed using deep deterministic policy gradients for continuous action selection [30]. They also use a deep Q-network to handle the back-off mechanism's discrete action selection. The DRL-based solutions significantly outperform traditional heuristic solutions, especially in heavy traffic scenarios, and the DRL-based ACB scheme outperforms the DRL-based BO and DQ schemes, according to their results.

Tuning both barring rate and barring time parameters of the ACB mechanism is suggested in work [31]. Bui and Pham feed the RL agent with channel state, and their reward function is based on a ratio of successful preambles. Their RL method, known as the dueling deep Q-network, is equivalent to our single rate approach, while their main concern is energy consumption. The conspicuous lack in this study is that they did not consider service differentiation among the UEs.

3.3. Other Approaches

The authors in [32] investigate the aggregations of UEs under different groups to coordinate the RACH procedure via group coordinators. The suggested model recalls a gateway in the networking design concept. They also proposed a local group update mechanism to overcoming the RA traffic. The group-based approach reduces the collision chance in the channel; however, it introduces coordination cost.

Vural et al. presented a "multi-preamble random access" algorithm to dynamically discover the size of the preambles in distinctive load situations[33]. By doing so, they aimed to reduce the collision probability and increase successful random access probability. At the end of the study, they conclude that the adapted algorithm can efficiently provide preamble discrimination in various service classes and support a high rate of RA load. The authors also

put forward a concept called ”vPreamble” as a virtual network function, and various ideas in this concept that each vPreamble example can be instantiated and configured for service classes such as IoT and mobile services.

Klymash's technique is unique in that it chooses the primary mobile node using the Voronoi diagram and fuzzy logic approaches [34]. They cluster, aggregate and classify UEs to enables more efficient use of the cellular network's radio resource. The main node delegates the communication, resulting in a reduction in signal load on the eNB.

Nwogu et al. deal with congestion problems in heterogeneous LTE via a three-way partitioned resource booking method [35]. Through a mixed static/dynamic resource utilization strategy, their technique reduces random access latency considering various QoS requirements. They also suggest a dynamic resource sharing scheme for a class between the URLLC and non-URLLC priorities.

Phuyal et al. compared the performance of both ACB and E-ACB within their simulations based on models suggested by 3GPP [36]. While most of the works concentrate on single-cell scenarios, Lien et al. present the cooperative ACB for global stability and access load sharing to remove significant flaws in the usual ACB method [37]. The Strong coding scheme is one of the overload control categories, and the work in [38] uses this approach to cope with the MTC through RACH attempts. The authors compare their results against E-ACB.

In another approach, A novel ACB configuration based on the ratio of idle preambles in the channel is proposed to feed an adaptive filtering algorithm [39]. Real-time ACB parameter adjustment is used to deal with MTC congestion issues in this research.

4. PROPOSED METHOD

Efficient RACH congestion control through the ACB mechanism requires making sequential decisions about the barring rate and barring time at each RAO. The control problem is potentially intricate and uncertain, considering the randomness factor in the fundamental design of the ACB. Above all, the effect or feedback of the decision taken may appear at any time interval since each UE independently and randomly can select a backoff period. Therefore, the reaction of a decision taken mixes with other decisions. The major complexity of the RACH congestion control is due to this distributed and mixed feedback structure. Besides, not the competing UE count nor their access time distribution is available to the eNB. Channel state is the only available information to eNB, which includes; idle, collided and successful preamble count. On the other hand, there is no explicit accuracy evaluation method for these consecutive decisions. All these characteristics make RL suitable for ACB parameter optimization. In specific for this purpose, we have implemented the Trust Region Policy Optimization [62] approach, which is an iterative procedure for optimizing policies with guaranteed monotonic improvement.

4.1. Priority Assignment

Service providers and regulators are meant to deliver a wide variety of QoS to their customers. Priority classification is either in favor of meeting these QoS requirements or based on internal regulation policies. Prior studies on prioritised random access divide the UEs based on their access delay sensitivity. [18], [19], [9] are used high, medium and low priority traffic in their work, while authors in [11] prefer to use high, low and scheduled priorities in their categorization. All these works aim to separate the emergent and high priority traffic like healthcare applications, fire sensors, public safety devices. Although this level of emergency shows a rare frequency of occurrence, it requires minimum delay and maximum channel access success chance. On the contrary, low priority class represents regular and settled data transmitters whose data loss and delay are tolerable. The medium category stands in between based on its QoS requirement.

Also, some works suggest five different categories including, H2H, low priority, high priority, scheduled and emergency [63], [64]. These works also follow the same idea and attempt to provide priority-based access in more diverse levels.

We support multiple levels of prioritization to cope with different QoS concurrently. Our proposed priority classification is configurable for any service provider and regulator to meet their requirements since there is no limitation on the number of classes. Therefore service differentiation is applicable in desired degrees. Assuming that n separate classes are required, the class number should assign form 1 to n from low to high priorities, respectively. The higher priority classes will provide low latency and high success rates in respect to lower priority classes.

4.2. Reinforcement Learning

Reinforcement learning is an area of machine learning concerned with how software agents evolve by interacting with the environment by observing it, taking actions and receiving immediate reward feedback to maximize some notion of cumulative reward.

Figure 4.1 represents the RL approach in which the RL agent observes the environment through the state space, while the reward value acts as a feedback mechanism for the agent.

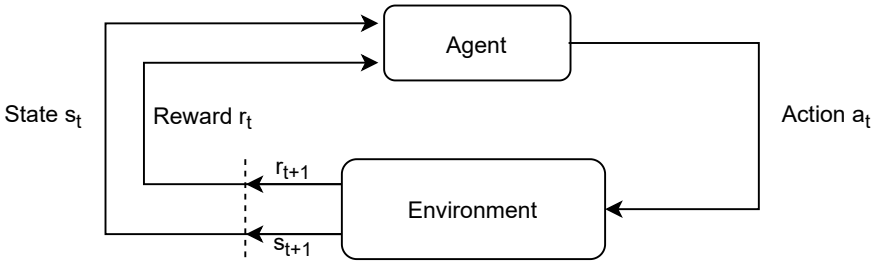

Figure 4.1 Typical framing of the RL scenario

We propose three different strategies to determine the barring time. In the first strategy, we use a single constant barring time powered by RL to dynamically and continuously tune the barring rates. In the second strategy, we introduce our constant multiple barring times scheme besides our multi-rate RL agent. And in the last strategy, we use the RL to tune the barring rates and barring times concurrently.

4.3. Multi-Rate Dynamic ACB Method with Constant Single Barring Time

We describe the environment via channel state at each RAO, including the number of idle N_I, collided N_C and successful N_S preambles individually for each priority class. The priority class of successful UEs is retrievable since their ID already has been decoded in the fourth step of the RACH process.

$$s = \{N_I, N_C, N_{\text{S}}^{PC_1}, N_{\text{S}}^{PC_2}, ..., N_{\text{S}}^{PC_n}\} \tag{3}$$

RL agent controls the traffic of each priority class by broadcasting multiple $P_{\text{ACB}}^{PC_n} \in [0, 1]$ rates simultaneously. Therefore, the action space count is equal to n, which is the number of defined priority classes in the system.

$$a = \{P_{\text{ACB}}^{PC_1}, P_{\text{ACB}}^{PC_2}, ..., P_{\text{ACB}}^{PC_n}\} \tag{4}$$

The eNB informs the UEs about changes in the barring rates through the SIB2 messages before each RAO. In the constant single barring time phase, we choose $T_{\text{ACB}} = 0.3s$ based on exhaustive optimization search results while we have considered the delay requirements of 5G Networks for massive type communications.

The reward function describes how the RL agent ought to behave. Since we aim to increase the success possibility of higher priority classes, we define the immediate reward for previous RAO based on the successful UE count of each priority class as follows:

$$r = N_{\text{PC}_1} \times C_1 + N_{\text{PC}_2} \times C_2 + ... + N_{\text{PC}_n} \times C_n \tag{5}$$

N_{PC_n} represents the number of successful preambles that belong to each priority class, while C_n is the coefficient of each priority class which we choose to be equal to the priority class

number. The priority classes should be assigned from 1 to n, where n indicates the highest priority class. The RL agent aims to achieve a higher reward in each iteration and the whole episode. Therefore, the coefficient differences encourage the agent to give priority access to classes with higher coefficients.

Our RL-based method only benefits from available information at eNB. Multiple barring rates in the action set and priority-based coefficients in the reward function grant the desired privileged slicing over the RACH usage. The provided results in the section 5. confirm the efficiency of our approach.

4.4. Multi-Rate Dynamic ACB Method with Constant Multiple Barring Times

Our study shows that the Barring Rate alone is not capable of controlling the congestion over the RACH. In the fact of the matter, a non-zero Baring Time is essential alongside the Barring Rate to form well-organized traffic management over the channel. However, determining this Barring Time is so critical. Assigning a long period as Barring Time may cause high delays, while a short period may cause an overload in the channel.

One of the principal considerations in this study is service differentiation. Although our proposed multi-rate approach is quite capable of this purpose, different priority classes still have to wait the same amount of time regardless of randomness in the barring time calculation process. By definition, each UE enters a backoff period after a collision occurs for that UE preamble transmission. We propose to send different barring times to reduce the delay for UE in higher priority classes. Therefore, it is possible to send a lower amount of barring times for higher priority classes.

The RL agent still benefits from the same state definition set, the reward function and the action set. The constant Barring Times for each priority class is calculated using the equation (6).

$$T_{\text{PC}_n} = \frac{CC - C_n + 1}{CC} \times 0.6 \tag{6}$$

In this equation, CC represents the total priority class count in the system, and C_n is the priority class number from 1 to n, where n indicates the highest priority class. The constant value of 0.6 seconds is the result of the exhaustive optimization search. This value is achieved with the same comprehensive search approach in the Constant Single Barring Time phase. For instance, in a scenario consisting of three different priority classes, the barring times will be respectively 0.2, 0.4 and 0.6 seconds for high, mid and low priority classes.

Provided results in the evaluation section prove that using multiple barring rates alongside the multi-rate RL-based approach can reduce the delay for higher priority classes while the RACH utilization is maximized.

4.5. Multi-Rate Dynamic ACB Method with Dynamic Multiple Barring Times

In the context of using individual barring time for each priority class, we use the RL capabilities to tune the baring times dynamically. Therefore, the RL agent dynamically adjusts not only the barring rates but also the barring times. Since using a dynamic barring rate increases the complexity of the congestion control problem, the RL agent needs to be fed more detailed information about the environment.

$$s = \{N_I, N_C, N_{\text{S}}^{PC_1}, D^{PC_1}, N_{\text{S}}^{PC_2}, D^{PC_2}, ..., N_{\text{S}}^{PC_n}, D^{PC_n}\} \tag{7}$$

In addition to the channel state information, D^{PC_n} the mean delay of each priority class is used to explain the environment to the RL agent. The RACH delay of UE is also another retrievable information for the eNB after decoding the id of the successfully received preamble.

The action space consists of both barring rates and baring times separately for each priority class. Each pair of the ACB parameters are meant to control UE in the different priority classes. In this way, the RL agent can control the behavior of each priority class without affecting other service types. This control infrastructure allows the RL agent to give full access to higher priority classes while partially banning the lower priority classes.

$$a = \{P_{\text{ACB}}^{PC_1}, T_{\text{ACB}}^{PC_1}, P_{\text{ACB}}^{PC_2}, T_{\text{ACB}}^{PC_2}, ..., P_{\text{ACB}}^{PC_n}, T_{\text{ACB}}^{PC_n}\} \tag{8}$$

After trying various reward functions, including the threshold restriction in counting the successful preambles, we use the same coefficient-based reward function in the first phase, as it can encourage the RL agent to properly prioritize classes.

$$r = N_{\text{PC}_1} \times C_1 + N_{\text{PC}_2} \times C_2 + ... + N_{\text{PC}_n} \times C_n \tag{9}$$

Achieved and presented results in the evaluation section emphasize the capability of the RL approach to handling the congestion control problem. The RACH utilization maximization and service differentiation goals are achieved. Moreover, the mean delay of higher priority classes is reduced.

5. EXPERIMENTAL RESULTS

A comprehensive simulator is designed for observing the congestion in the network. It can simulate a simplified version of the RACH procedure in different conditions. In this study, we implement the network configuration and the traffic models suggested in the 3GPP specification [12]. A test scenario, also called an episode, consists of 2000 RAOs of 5ms long, so each episode duration is 10 seconds. Only 54 preambles from a 64 shared preamble pool are specified for the contention-based RA procedure. The successive unsuccessful preamble transmission limit, preambleTransMAX, is set to 10. Table 5.1 lists additional parameters used throughout the network configuration.

Parameter	Setting
Cell bandwidth	5 MHz
Periodicity of RAOs	5 ms
Subframe Length	1 ms
PRACH Configuration Index	6
Total number of preambles	54
Maximum number of preamble transmission	10
Number of UL grants per RAR	3
Number of CCEs allocated for PDCCH	16
Number of CCEs per PDCCH	4
Ra-ResponseWindowSize	5 subframes
mac-ContentionResolutionTimer	48 subframes
Backoff Indicator	20ms
HARQ retransmission probability for Msg3 and Msg4	10%
Maximum number of HARQ TX for Msg3 and Msg4	5

Table 5.1 RACH configuration suggested in 3GPP specifications

Suggested traffic models for Machine Type Communication (MTC) in 3GPP specifications are presented in table 5.2 ”Traffic model 1” represents a uniform distribution of UEs over the time to simulate a non-synchronized access behavior of UEs in the network. As the extreme scenarios ”traffic model 2” can be granted, in which a massive number of UEs request to access the network simultaneously in a highly synchronized manner.

Characteristics	Traffic model 1	Traffic model 2
Number of MTC devices	1000, 3000, 5000, 10000, 30000	1000, 3000, 5000, 10000, 30000
Arrival distribution	Uniform distribution over T	Beta distribution over T
Distribution period (T)	60 seconds	10 seconds

Table 5.2 Suggested traffic models for MTC in 3GPP specifications

We define Preamble Transmission Rate (PTR) as the count of UEs that simultaneously trigger the handshake process. PTR determines the connection success rate at each RAO. Low PTR increases the success probability of an RAO, while high PTR reduces it. The average results

of one billion independent random preamble selection are provided in figure 5.1. Achieved outcomes are similar to the study in [8]. The success rate probability has a parabolic behavior in respect to PTR change. The maximum chance is reachable when the PTR is equal to the available preamble count (54), which is in line with the theoretical optimum calculations. However, the collision rate is also critical due to the drop mechanism.

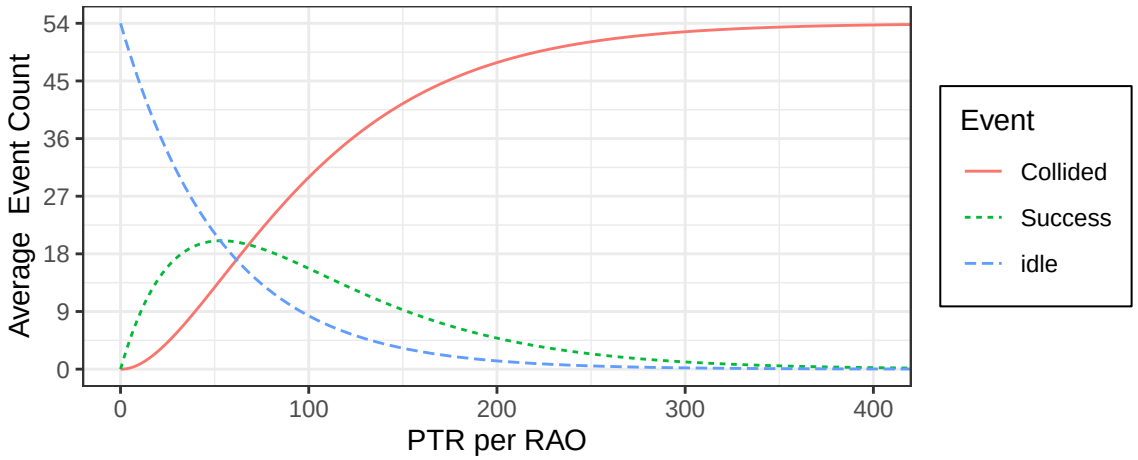

Figure 5.1 Channel state under different PTR

In the best case, only 54 UEs can establish a connection simultaneously, provided that each UE has chosen a different preamble in the handshake process. However, as nodes choose their preambles randomly, it is impossible, and according to [8], the highest successful UE count is about 21. The higher PTR, the higher the collision chance in the channel. By exceeding a critical PTR level, congestion starts due to this inherent limitation in the RACH. As it can be seen in figure 5.1, in a channel with PTR higher than 300, the success share is almost zero, while all 54 preambles face collision.

In a simple experiment, the number of UEs is increased gradually to examine these intervals in a network without any congestion control policy. There is also no backoff mechanism while the drop mechanism is active. Achieved results are demonstrated in figure 5.2. Our results are consistent with the results given in the 3GPP specification [12] provided in table 5.3.

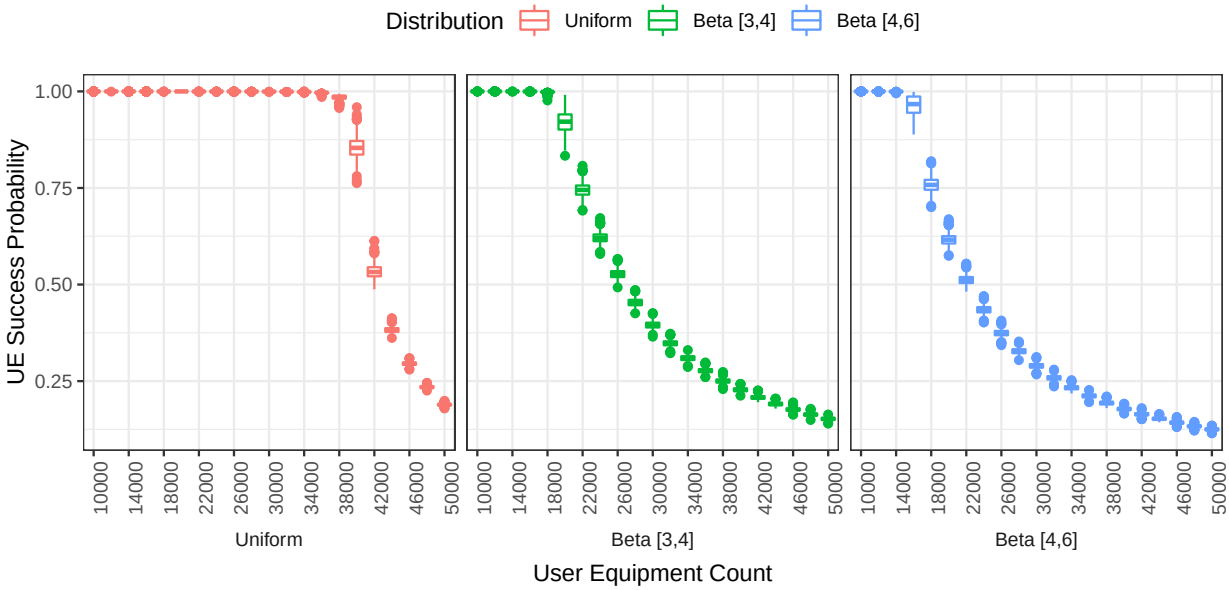

Figure 5.2 Congestion starting intervals in the different distributions over a pure network, without any congestion control policy

Traffic Model	**Performance measures**	**Number of MTC devices per cell**		
		5000	**10000**	**30000**
1	Access Success Probability	100%	100%	100%
2	Access Success Probability	100%	100%	29.5%

Table 5.3 3GPP Simulation results for RACH capacity for LTE FDD

Congestions in the episodes and drops in success rates start when UE count exceeds about 38000, 18000 and 14000 respectively for Uniform, Beta(3, 4) and Beta(4, 6) distributions. The achieved results emphasize that only one test scenario faces congestion in the traffic models suggested in the 3GPP specification [12] and also almost all previous works. The remaining cases can be handled even without any congestion control policy. We extend the 3GPP traffic models to make the traffic more challenging and to prevent overfitting in the learning process. Therefore, we add Beta(4, 6) distribution and increase the UE count in the episode. Each episode is constructed with a random distribution and UE count to make the simulation more realistic. Besides, UEs are assigned to different priority classes randomly, so an almost equal number of each priority class exists in an episode.

5.1. Perfect Control Benchmark

At each RAO, the optimal value for the Barring Rate can be calculated as

$$P_{\text{ACB}} = min(1, M/n) \tag{10}$$

according to [5], [11] and [65]. In this equation, M is the number of available preambles, while n is the candidate count for preamble transmission in the upcoming RAO. In the theoretical optimum approach, the main target is to limit the PTR equal to the available preamble count $M = 54$, assuming that eNB is aware of the value of n. In practice, due to the essential probability properties, PTR is fluctuating around the desired optimum value. Since benchmarking is the only purpose here, we relax the assumption for the first improvement. Therefore, we assume that eNB knows all random numbers (q) generated by candidate UEs in each priority class in the upcoming RAO. In this way, using a simple sort operation, the eNB can determine the P_{ACB} such that exactly 54 UEs have the opportunity of preamble transmission.

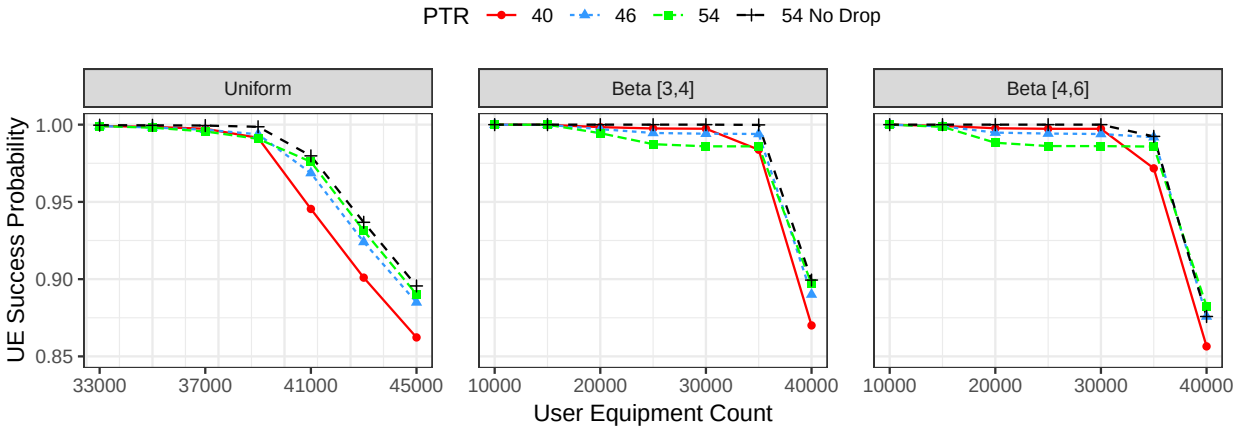

Figure 5.3 Performance comparison of the perfect control mechanism applying different fixed PTR policies

When the value of the broadcasted barring rate is equal to one, the eNB grants access to all contending UEs, and basically, there is no restraint for the upcoming RAO. If n is equal or less than M, the equation returns one as the barring rate value. Regarding these facts,

the theoretical optimum approach essentially tries to limit the number of contending UEs to the available preambles count, which is 54. In a single RAO, the theoretical optimum approach results in the maximum success rate in the RACH procedure. However, there is also a noticeable collision probability. Considering the sequential RAOs, this collision probability may result in connection request failures and packet losses due to the drop mechanism. Based on the standard, each UE stops preamble transmission after ten successive unsuccessful connection attempts. We have applied an exhaustive search over the PTR value to minimize the drops and define the perfect control mechanism. As figure 5.3 shows, the theoretical optimal (PTR = 54 No Drop) achieves the highest success rate, while the drop mechanism is disabled. However, by enabling and applying the drop mechanism in the simulation, the mean success rate of the theoretical optimal method decreases, especially in the range 20000 and 30000 in Beta distributed scenarios. The PTR = 46 accomplishes a higher overall performance in various test cases. It is worth mentioning that these details have not been noticed in the previous works since they rarely test heavy traffic scenarios, and the drop mechanism has not been considered in most of these studies.

Our novel perfect control approach not only outperforms the theoretical optimum approach but makes it possible to implement multi-rate perfect control. To do so, starting from the highest priority class, eNB allocates the preambles by choosing the proper barring rate. Then remaining capacity is used to specify the barring rate for the next priority class, and so on.

5.2. Performance Evaluation

In evaluating our proposed multi-rate ACB scheme, many Key Performance Indicators (KPI) have been considered: throughput, success probability, PTR, drop, backlog, number of attempts, delay and delay standard deviation. Although success probability and delay are the most important KPI factors for our study, we observe other KPIs as well. Our dynamic proposed method can tune the barring rate for each ACs 0 to 9 individually. We have tested the proposed method with different numbers of priority classes like 3, 5 and 10 levels. In all of these cases, the multi-rate approach has shown robust performance and consistent channel

usage, even under heavy loads. Nevertheless, due to lack of space, we present the result of three and five different priority class scenarios under a limited UE count. We compare multi-rate RL-based simulation average results with a benchmark with no ACB mechanism as the lower bound, a dynamic single rate ACB adjusted by RL, DRL approach (suggested by Chen et al.), and the perfect multi-rate control as the upper bound. Our multi-rate RL-based outperforms the theoretical optimum results due to the drop mechanism. Therefore, as it is explained in the previous section, we developed perfect control inspired by the theoretical optimum approach.

5.2.1. Chen's DRL Approach

Among the previous studies, the DRL approach suggested by Chen et al. is one of the pilot RL-based methods which has considered the service differentiation in the RACH procedure. Therefore we also compare our method against their DRL approach. There are some fundamental configuration differences between our and Chen's research. First of all, we assume that there can be any arbitrary priority class in the scenario, while their work is limited to three different priority classes. Secondly, in their configuration, a constant value of UEs exists in the test case. In our work, the UE count and distribution are chosen randomly just before each episode. They consider a lower ratio of total UEs in higher priority classes, while in our study, each priority class has an equal number of UEs, which makes the control more challenging. Above all, our configuration includes test cases with higher UE than models suggested in 3GPP. We have implemented their proposed method using their recommended threshold values 1, 5 and 10 seconds from high to low priority classes.

In the following subsections, we start by presenting the efficiency of our method in tuning the barring rate parameter in an environment with three coexistence priority classes. For this purpose, all KPI factors and statistics achieved from simulations are presented in detail. Results are presented in both figures and tables to benefit the simplicity of visualization and numbers precision. Then we demonstrate the assessment of our multi-rate RL-based dynamic approach under five various classes. Finally, we explain our developments on adjusting

barring time via constant and dynamic methods. For uniform distribution, we have only presented scenarios with more than 33000 UEs in them. The scenarios with a lower number of UEs result in all UEs being successful regardless of the control mechanism, as explained in figure 5.2.

5.2.2. With 3 Various Priority Classes

5.2.2.1. Success Rate Results Figure 5.4 and table 5.4 demonstrate the overall success rate probability in Uniform, Beta(3, 4) and Beta(4, 6) distributed scenarios for varying numbers of UEs. The overall mean success rate probability is the foremost important KPI in our study. Achieved results confirm that using RL or DRL is an efficient method to maximize the RACH success rate. Thus all the RL-based methods, including single-rate, Chen and multi-rate, have very close results to the perfect control approach. As opposed to what might be expected, using a multi-rate ACB rather than a single rate ACB does not reduce the performance, as the results prove the fact. For instance, when the UE count is equal to 40000 and arrival distribution is Beta(3, 4), the overall success rates are 0.22, 0.86, 0.87, 0.87 and 0.88 for No-ACB, Chen DRL, RL(single rate), RL(3-rate) and perfect control approaches, respectively.

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Chen	Overall	99.98	99.98	99.68	98.66	98.09	96.30	86.36	99.99	99.90	98.98	98.51	98.27	95.51	84.80	99.78	99.68	99.38	98.25	93.86	89.43	85.47
RL (Single Rate)		100	99.99	99.58	98.52	98.38	97.98	87.87	100	99.87	98.68	98.41	98.34	97.35	86.35	99.90	99.80	99.55	98.80	95.59	91.05	87.02
RL (3-Rate)		100	99.99	99.79	99.58	99.29	97.85	87.33	100	99.92	99.62	99.29	98.75	96.84	85.70	99.88	99.76	99.39	97.67	93.61	89.47	85.70
Perfect Control		100	99.99	99.71	99.45	99.40	99.39	88.99	100	99.89	99.48	99.41	99.39	99.12	87.55	99.91	99.82	99.67	99.38	96.89	92.43	88.34
Chen	Low	99.93	99.95	99.66	98.52	97.81	96.65	86.52	99.97	99.88	98.89	98.31	97.98	95.85	85.45	99.65	99.54	99.15	98.03	94.05	90.21	85.88
RL (Single Rate)		100	99.99	99.58	98.53	98.37	97.99	87.88	100	99.87	98.68	98.41	98.35	97.35	86.36	99.90	99.80	99.55	98.80	95.59	91.04	87.01
RL (3-Rate)		100	99.99	99.79	99.65	99.60	96.86	65.93	100	99.92	99.70	99.62	99.53	94.46	61.70	99.83	99.60	98.74	94.07	82.05	69.65	58.29
Perfect Control		100	99.99	99.71	99.47	99.45	99.43	68.27	100	99.89	99.50	99.46	99.44	98.64	63.93	99.91	99.83	99.67	99.30	92.17	78.80	66.54
Chen	Medium	100	99.99	99.69	98.61	98.14	96.34	88.75	100	99.91	98.98	98.51	98.27	96.17	86.47	99.87	99.76	99.52	98.48	94.85	90.76	87.87
RL (Single Rate)		100	99.99	99.58	98.52	98.38	97.97	87.89	100	99.88	98.68	98.41	98.34	97.34	86.35	99.90	99.80	99.55	98.80	95.59	91.05	87.01
RL (3-Rate)		100	99.99	99.78	99.55	99.13	98.37	98.20	100	99.92	99.58	99.12	98.45	98.33	98.33	99.91	99.84	99.71	99.46	99.38	99.39	99.40
Perfect Control		100	99.99	99.70	99.44	99.38	99.37	99.37	100	99.89	99.47	99.39	99.36	99.36	99.37	99.91	99.82	99.67	99.42	99.25	99.24	99.24
Chen	High	100	99.99	99.70	98.86	98.32	95.91	83.82	100	99.91	99.09	98.70	98.55	94.52	82.49	99.84	99.73	99.48	98.25	92.69	87.33	82.64
RL (Single Rate)		100	99.99	99.58	98.51	98.38	97.98	87.85	100	99.87	98.67	98.41	98.34	97.36	86.35	99.90	99.80	99.55	98.80	95.57	91.06	87.03
RL (3-Rate)		100	99.99	99.78	99.54	99.14	98.34	97.84	100	99.92	99.58	99.11	98.26	97.73	97.07	99.91	99.84	99.71	99.47	99.39	99.40	99.41
Perfect Control		100	99.99	99.70	99.45	99.38	99.36	99.34	100	99.89	99.48	99.39	99.36	99.34	99.34	99.91	99.82	99.67	99.42	99.25	99.24	99.25

Table 5.4 Numeric presentation of success percentages of different congestion control mechanisms in a network consisting of 3 various priority classes

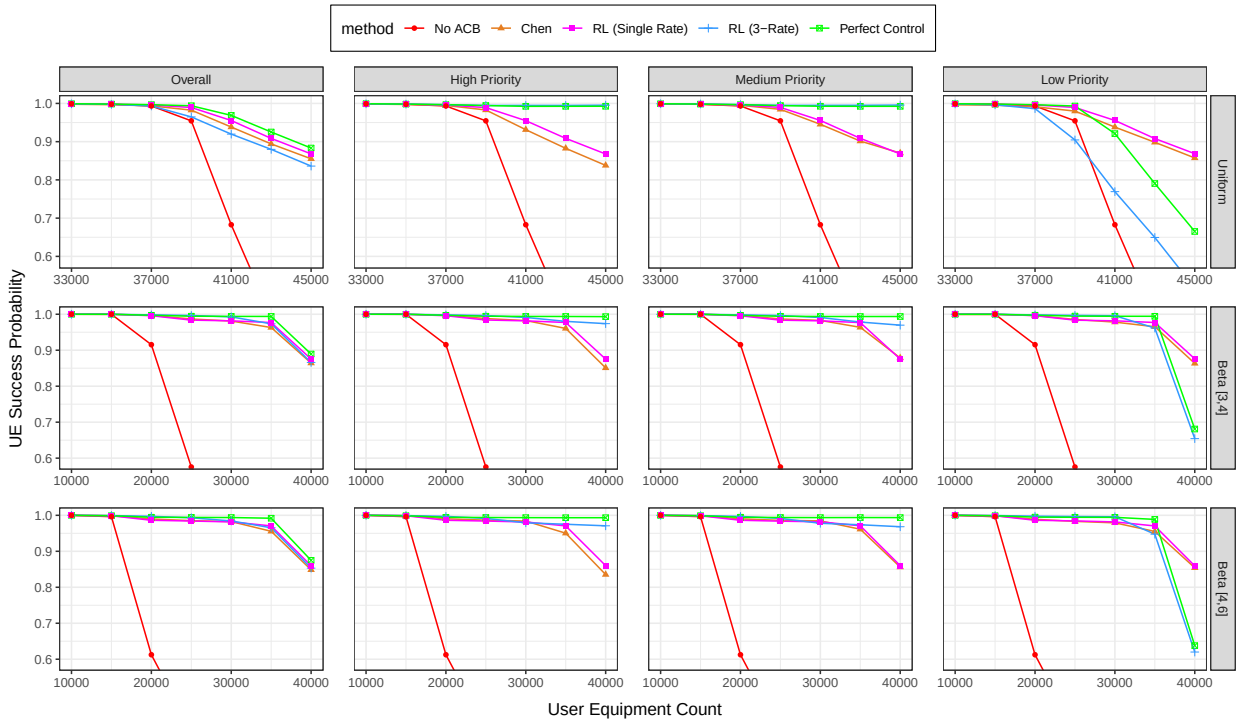

Figure 5.4 Overall and priority-based success rate comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

On the other hand, analyzing priority-based results exhibit our multi-rate method's difference. Since there is no service differentiation in No-ACB and RL(single-rate) approaches, their Class-based results are the same as their overall results. Although Chen's DRL method can improve RACH utilization, it does not show suitable functionality in service differentiation, especially in episodes with a high UE count. In contrast, in the multi-rate approach, higher priority classes have almost complete success rates. In High UE counts like 40000, in which the PTR is high, and congestions occur, the multi-rate method chooses to restrict only the lower priority class, and therefore after exceeding the capacity, only the success rate of the low priority class falls.

Continuing with our exemplary 40000 Beta(3, 4) scenario, in the RL(3-rate) approach, success rates are 0.97, 0.98 and 0.65 for high, medium and low priority classes, respectively. Simulation results are close to the perfect control results, which are 0.99, 0.99 and 0.68. The results of the single rate method are 0.87, 0.87 and 0.87, which is equal to the overall success rate of the single rate approach. Figure 5.5 presents a closer look at the success share

of each priority class under mentioned methods. The bars represent the overall success of these approaches, while red, yellow and blue colors represent the high, mid and low priority classes. The bars almost have the same height, indicating a near success rate. However, in the Perfect control and multi-rate bars, the red color has a larger portion, which symbolizes the higher success share of higher priority classes.

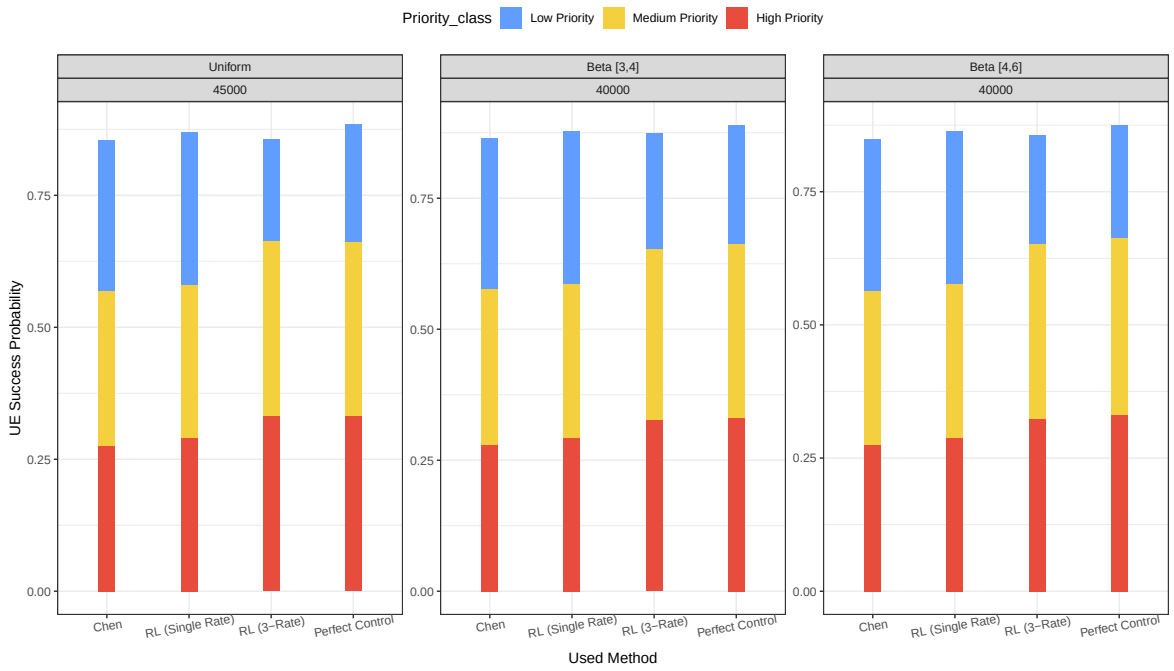

Figure 5.5 Priority-based success-share distribution of different congestion control mechanisms

The same priority-based results are achieved with different UE counts and distributions. As it can be seen, our multi-rate method maintains the maximum channel utilization, while access opportunities are distributed based on priority classes.

Scrutinizing the multi-rate results shows that in some scenarios, the success ratio of the mid-class is slightly higher than the high priority class. However, this negligible difference does not literally mean that the mid-class achieves a higher priority in the RACH procedure. The delay results prove that higher priority classes have superiority over the medium priority class.

As a matter of fact, not only our multi-rate RL-based approach but also the perfect control method faces the same situation. The drops are the keys to explain these abnormal result

points. The priority-based approaches intend to increase the success rate of higher priority classes. Meanwhile, this policy raises the collisions chance in these classes too. Although the collision chance is not high enough to reduce the success chance, it may result in undesired drops due to the maximum attempt count constraint. Therefore we closely trace the drop results, which are presented in figure 5.6. The drop ratio is one of the information that eNB can not collect throw the standard RACH mechanism. Therefore we do not use this data for training purposes.

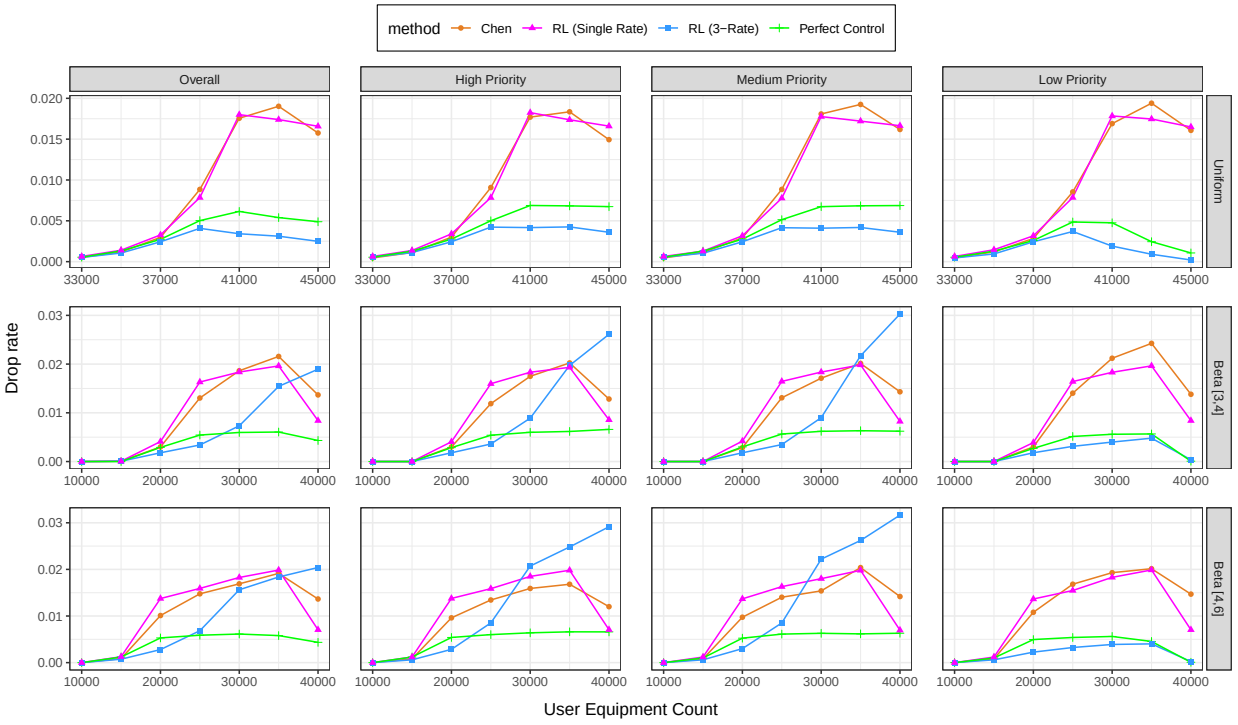

Figure 5.6 Overall and priority-based drop rate comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

An example of this situation can be seen in a Beta(3, 4) distributed 40000 UEs test case. The Success ratio of the high priority class is 0.9784, while the mid-priority achieves a chance of 0.9820 in the channel. However, the mean delays are 13 milliseconds against 865 milliseconds for high and mid classes, respectively. In fact, according to the delay results, a UE of a higher priority class manages to establish connection 66 times faster than mid-class equipment in this scenario.

5.2.2.2. Delay Results In the priority-based access control, the delay importance is as high as the success rate. Observed delay statistics are in line with the success rate outcomes, as figure 5.7 shows. The overall average mean delay of the listed methods exposes a negligible difference. Analyzing RL(single rate) and Chen's methods priority-based, results do not show a considerable difference between priorities. UE of any priority classes senses the same delay before connection establishment under these methods. On the contrary, in our multi-rate strategy, except for the low priority class of the uniform distribution, delay results overlap with the perfect control approach. For higher priority classes, the delay is reduced as much as possible to the perfect control benchmarks.

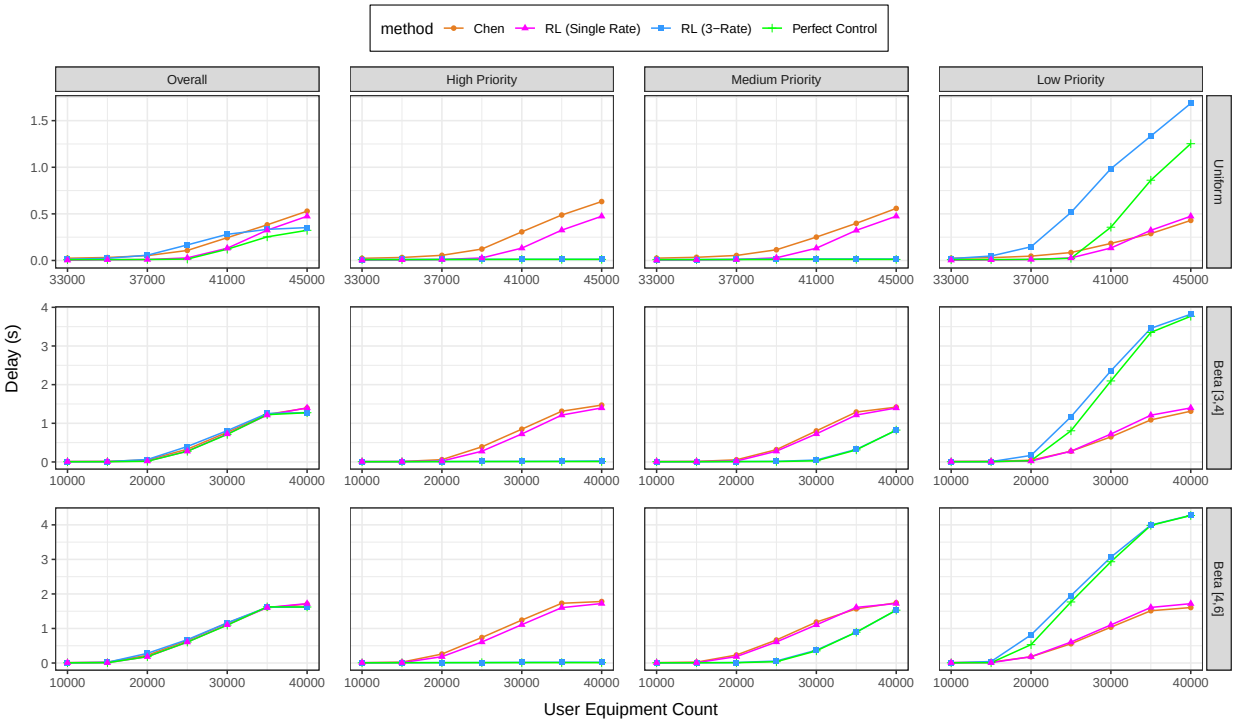

Figure 5.7 Overall and priority-based delay comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

Table 5.5 provides a closer look at obtained delay results. The stable delay result of both the perfect and RL(3-rate) approaches is noticeable in the uniform distribution. Regardless of the existent UE count, the delay is in the range of 8 to 11 ms for the high priority class. In the medium priority class, this range is slightly higher, 8 to 20 ms. However, the RL(single rate) and Chen's methods can cause a delay up to 600 ms depending on the traffic rate for medium

and high priority classes. Analyzing the low priority class validates that the RL(single rate) and Chen's methods manage the traffic with the same delay for higher priority classes. While perfect and RL(3-rate) approaches keep UE of low priority classes on hold for a higher period.

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Chen	**Overall**	9	13	53	328	768	1228	1390	10	25	224	654	1157	1599	1704	23	31	51	106	242	378	522
RL (Single Rate)		5	6	24	280	720	1215	1397	6	9	185	606	1105	1611	1721	8	10	14	32	129	307	472
RL (3-Rate)		6	7	45	350	774	1227	1260	6	15	239	648	1138	1599	1610	10	16	33	100	216	292	324
Perfect Control		5	6	17	277	720	1225	1273	6	8	184	603	1104	1627	1622	8	9	10	15	117	254	321
Chen	**Low**	12	15	43	273	647	1097	1325	12	24	177	558	1038	1525	1617	24	29	45	82	174	278	422
RL (Single Rate)		5	6	24	280	720	1215	1398	6	9	185	606	1105	1611	1720	8	10	14	32	129	307	472
RL (3-Rate)		6	7	116	1021	2236	3338	3698	6	28	690	1854	2981	3931	4197	14	29	80	286	707	1086	1379
Perfect Control		6	6	34	811	2119	3349	3769	6	8	531	1762	2944	3995	4278	8	9	11	24	347	866	1247
Chen	**Medium**	8	13	52	294	765	1226	1377	10	25	213	640	1111	1492	1718	25	33	52	108	232	372	521
RL (Single Rate)		5	6	24	280	719	1215	1397	6	9	185	606	1105	1611	1722	8	10	14	32	129	307	472
RL (3-Rate)		6	6	10	18	67	361	865	6	8	18	72	400	936	1562	8	9	10	14	17	18	20
Perfect Control		5	6	8	10	30	313	820	6	8	10	36	360	892	1522	8	9	9	10	11	11	11
Chen	**High**	7	11	63	415	885	1367	1494	8	26	281	761	1314	1792	1793	22	31	56	128	328	517	664
RL (Single Rate)		6	6	24	280	720	1215	1397	6	9	185	606	1105	1610	1721	8	10	14	32	129	307	471
RL (3-Rate)		6	7	9	10	11	13	13	6	8	10	11	13	13	14	8	9	10	11	11	11	11
Perfect Control		6	6	8	10	10	11	11	6	8	10	10	11	11	11	8	9	9	10	11	11	11

Table 5.5 Numeric presentation of mean delay of different congestion control mechanisms in a network consisting of 3 various priority classes (Millisecond)

The same policy is trackable for the Beta(3, 4) and Beta(4, 6) distributed scenarios. In a test case built of 35000 UEs with Beta(4, 6) pattern, the overall delay is approximately 1600 milliseconds using any of the advanced algorithms in the comparison. Also, with these methods, the mean delay of low, medium and high priority classes slightly fluctuates around 1600 milliseconds. With significant improvement, our RL(3-rate) scheme delay results are 13, 936, 3931 for high, medium and low classes, respectively. The results are competitive with perfect control results, which are 11, 892 and 3995 milliseconds.

The delays are shifted from higher priority classes to lower priority classes, while the overall mean delay is the same for both the single rate and multi-rate approaches. We can conclude that the UEs in the high priority class manage to connect immediately and experience no delays, while the highest waiting period belongs to the lowest priority.

Diagram 5.8 demonstrates the delay standard deviation. The overall delay standard deviations of all intelligent approaches are close. Desired service differentiation is reflected

in success rate and delay results. Desired service differentiation is strongly reflected in the success rate and delay results. Delay standard deviation, especially in high priority class, supports this outcome. As it can be seen, in the high priority class Delay standard deviation of RL(3 Rate) and Perfect Control schemes are almost zero. On the other hand, Delay STD shows a rising trend in the single rate and Chen's outcomes. This increase is directly related to the number of equipment.

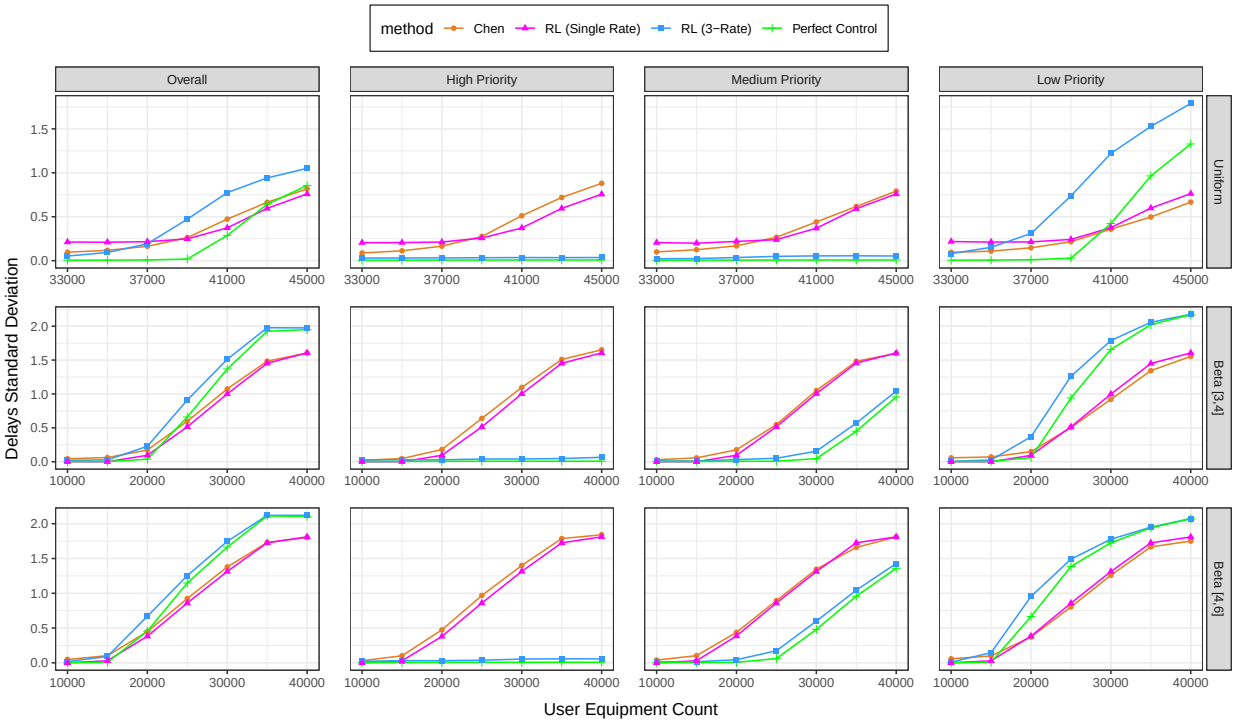

Figure 5.8 Overall and priority-based delay standard deviation comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

5.2.2.3. Mean Attempt Count Results Efforts on reducing energy consumption rely on minimizing the attempt count of UEs during the connection establishment process. In other words, these studies interest in increasing the success probability at the first attempt. For this reason, the total attempt count of UEs during the RACH is another important KPI. We analyzed the attempt count and frequency of both successful and unsuccessful equipment. However, since the only retrievable information in eNB gathers from decoded preambles, we do not include this extra information in our reward development or training process. Figure

5.9 represents the mean attempt count of successful UEs, while figure 5.10 demonstrates the mean attempt of all UEs in the scenario.

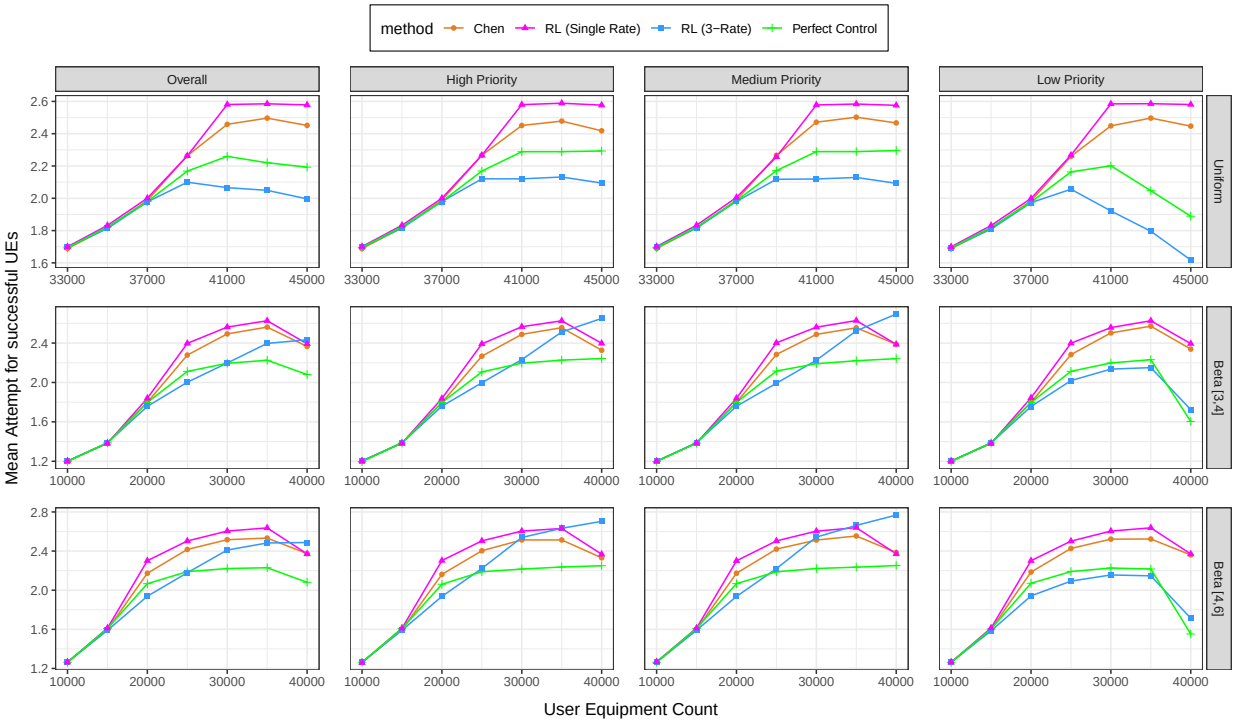

Figure 5.9 Overall and priority-based successful UE attempt count comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

With a glance at the attempt count charts, it can be said there is a rising trend, which has a direct relation with the UE count in the environment. However, after a threshold, a certain decline can be observed in the diagrams. We can explain this abnormal behavior considering the backlog statistics in figure 5.11. Since the episodes are restricted to 2000 RAO, therefore in some test cases, UEs may remain in the backlog. Each UE at backlog has already had at least one unsuccessful RACH attempt. These turning points of the attempt count curves exactly overlap with the start point of accumulation in the backlog. For instance, the Beta distribution attempts counts are in the range of 1.2 to 2.6. The curve turns down at 35000 UEs, which is equal to the increasing point of backlog ratio in figure 5.11.

There is a trade-off between minimizing the attempt count and maximize RACH utilization. This study targets to control congestion, maximize RACH utilization, and provide service

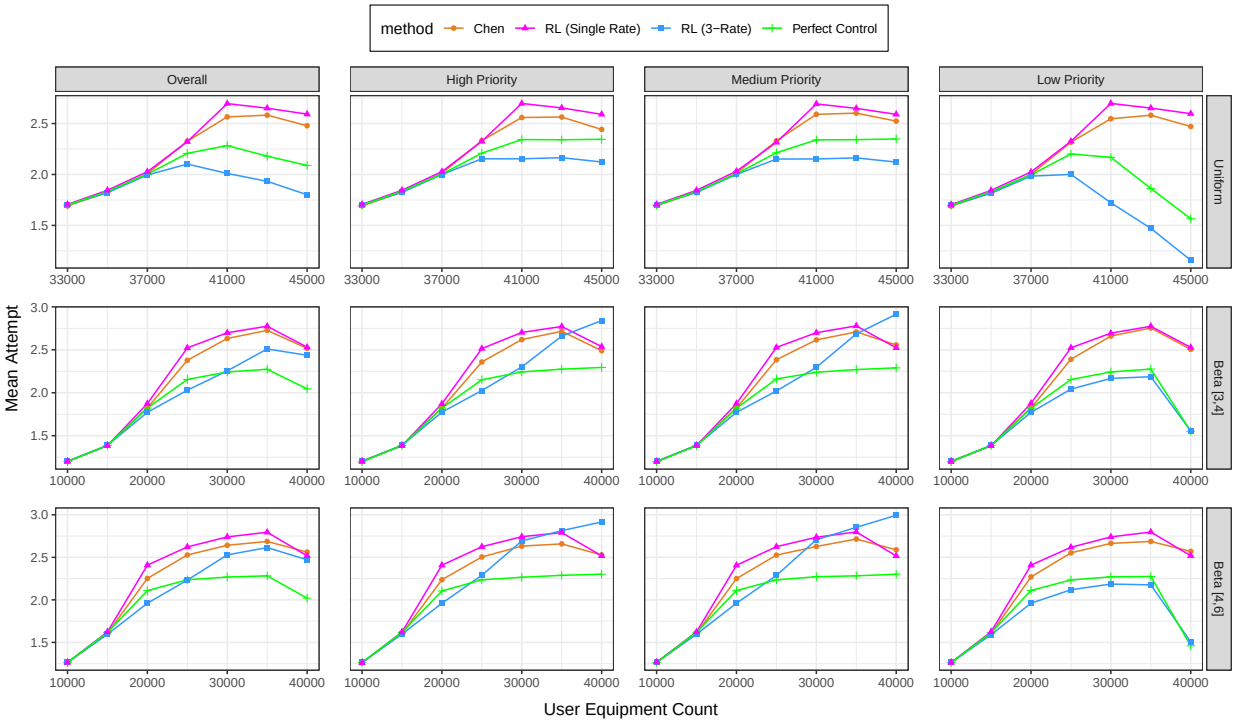

Figure 5.10 Overall and priority-based attempt count comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

differentiation. Therefore we do not aim to train the agent to push the attempt count toward one.

The second inference from the graphs is that the attempt count of unsuccessful UE is higher than the successful UE, which results in an upward shift in the entire UE results of figure 5.10, comparing to figure 5.9. Since the attempt count of dropped UEs is ten by definition, and since the total mean attempt count includes the dropped UEs too, this upward shift can be explained.

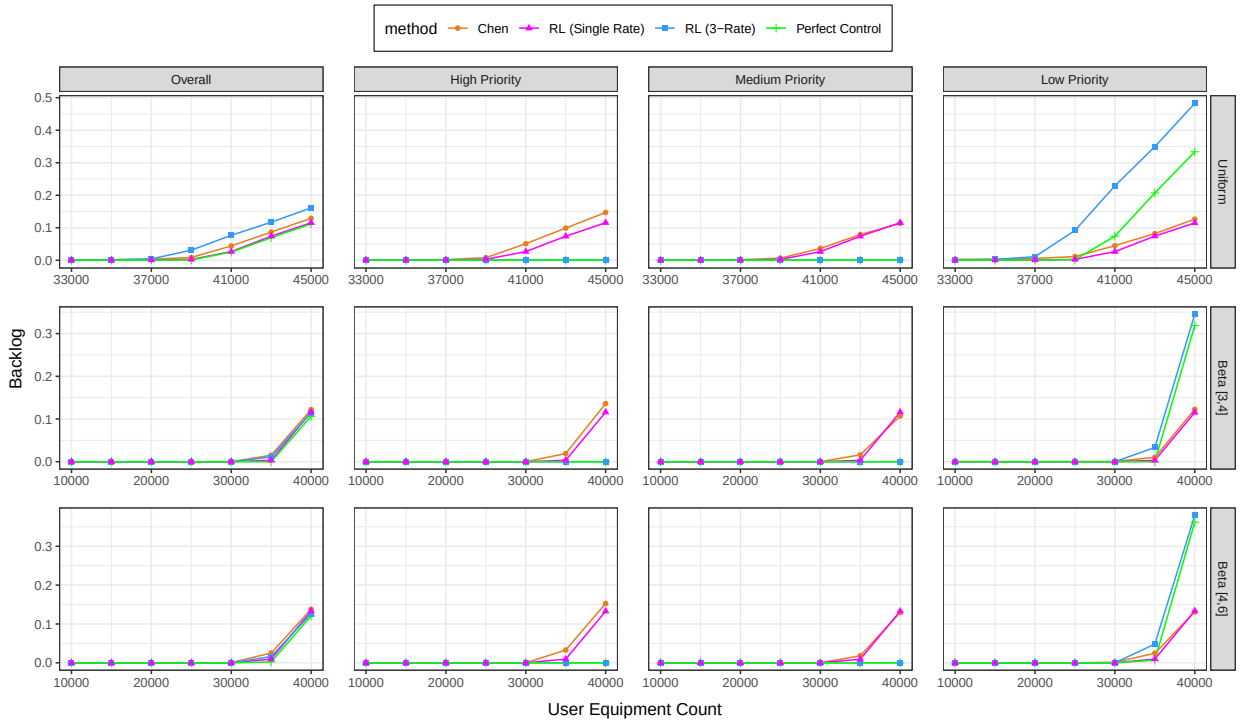

Figure 5.11 Overall and priority-based Backlog rate comparison of different congestion control mechanisms in a network consisting of 3 various priority classes

5.2.2.4. Exemplary Episodes Analyzes In the final stage of the performance evaluation of our RL-based multi-rate approach under three different priority classes, we demonstrate details of some sample episodes to demonstrate our service differentiation strategy. These sample episodes (figures 5.12, 5.13, 5.14, 5.15, 5.16, 5.17) help to explore the operation of each control mechanism.

At each RAO of these episodes, various principal parameters such as arrival count, PTR, barring rate change, channel state, and success share are demonstrated. Some episode statistics are provided as well in the last row of these diagrams. These visual episode presentations are helpful in examining the following questions.

- How an approach handles the congestion using the barring rate, and how does barring rates change?
- What is the instant effect of this barring over the channel state?

- What is the success share of each priority class at each RAO?

Figure 5.12 represents a sample episode under different control mechanisms. In this exemplary episode, there are 35000 UEs distributed by Beta(3, 4). In all four methods, since the UE count and their distribution are identical, the arrival rate is the same, and all episodes are faced with an equal load (figure 5.12.a). In the first column, where there is no ACB or any other control approach, the channel is severely congested. The sharp increase in PTR and collisions in the channel state diagrams are indicators of this heavily congested situation. In the RL(single rate) case, channel utilization has been maximized by the RL agent. However, the success shares of all priority classes are the same for this standard method. The 3-rate RL-based mechanism takes the channel control one step further. In this way, our approach provides service differentiation in addition to the utilization maximization goal. As mentioned, ACB controls the channel traffic through the barring rates. In our method, the lower priority class is barred in the first place after the start of the congestion. Then as the traffic increases, the middle priority class is also started to be barred. During this process, there is no barring for high-priority classes. The barring rate adjustments show similar behavior in our method and the perfect control method (figure 5.12.b). The channel state row of the diagram (figure 5.12.c) shows how the proposed methods can keep the channel stable and maximize utilization, while the success share row (figure 5.12.d) demonstrates the service differentiation properties. It shows how each approach distributes the load for different priority classes, considering that all priority classes have an equal share in the arrival phase.

Figure 5.13 displays an excellent example of a highly congested channel. In this test case, 40000 pieces of equipment compete to establish a connection during 10 seconds. This scenario includes 10 thousand more equipment than the most crowded model suggested in 3GPP. The dynamics and policies are still the same as in the previous example. Congestion control, channel utilization maximization, and priority-based access requirements are being met.

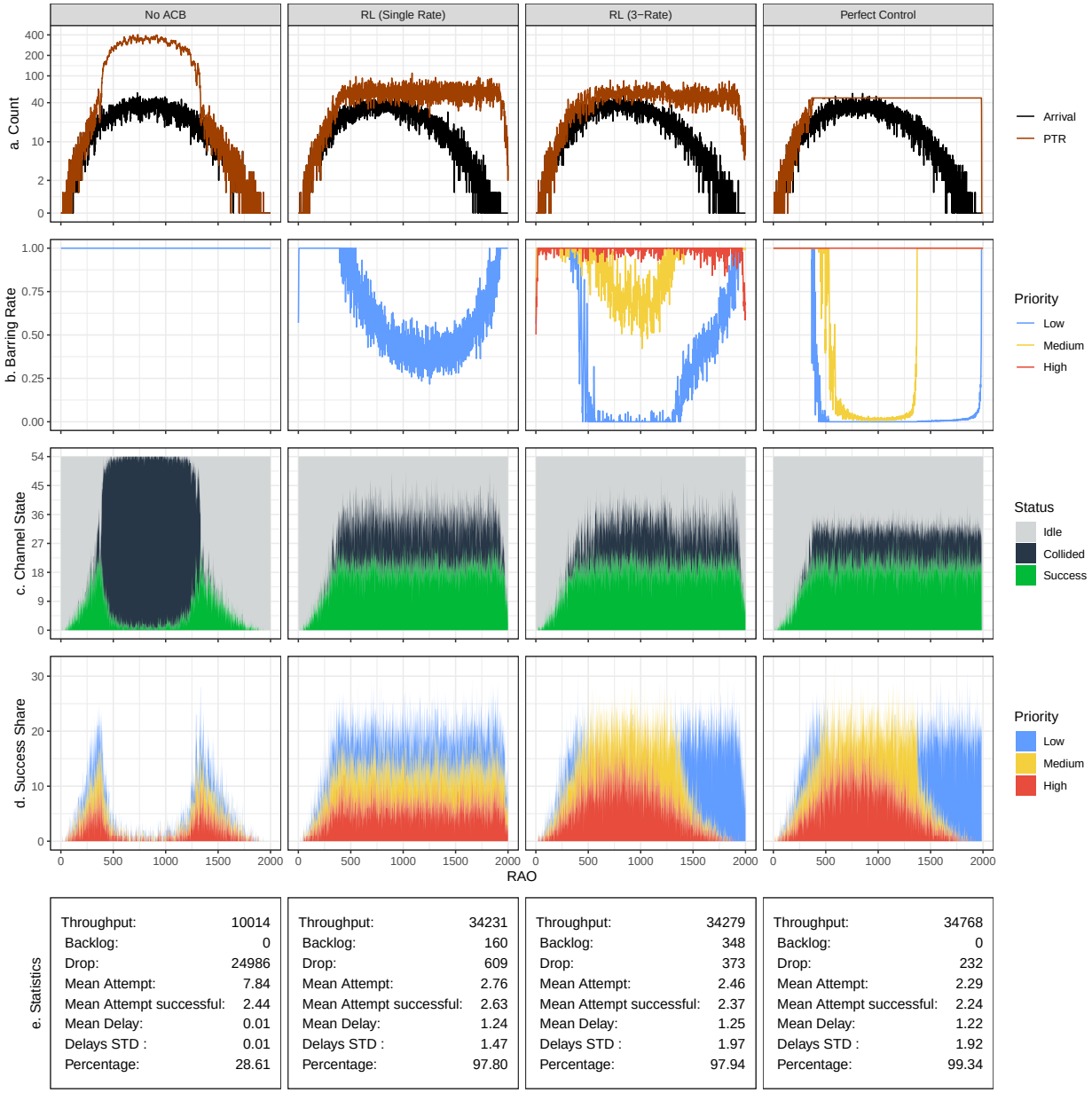

Figure 5.12 Visual comparison of different congestion control mechanisms operation over an episode which is consists of 35000 UE Beta [3, 4] distributed, with three various priority classes

We tried to prevent the overfitting problem by adding the Beta(4, 6) distribution to our random scenario creator. The Beta(4, 6) distribution also causes a high PTR in respect to the other patterns, which results in a higher degree of congestions to the channel. As can be seen in figures 5.14 and 5.15, in the NO ACB method, the PTR exceeds 500 and 600 at their peak points. Therefore handling the extra pressure imposed by this type of distribution

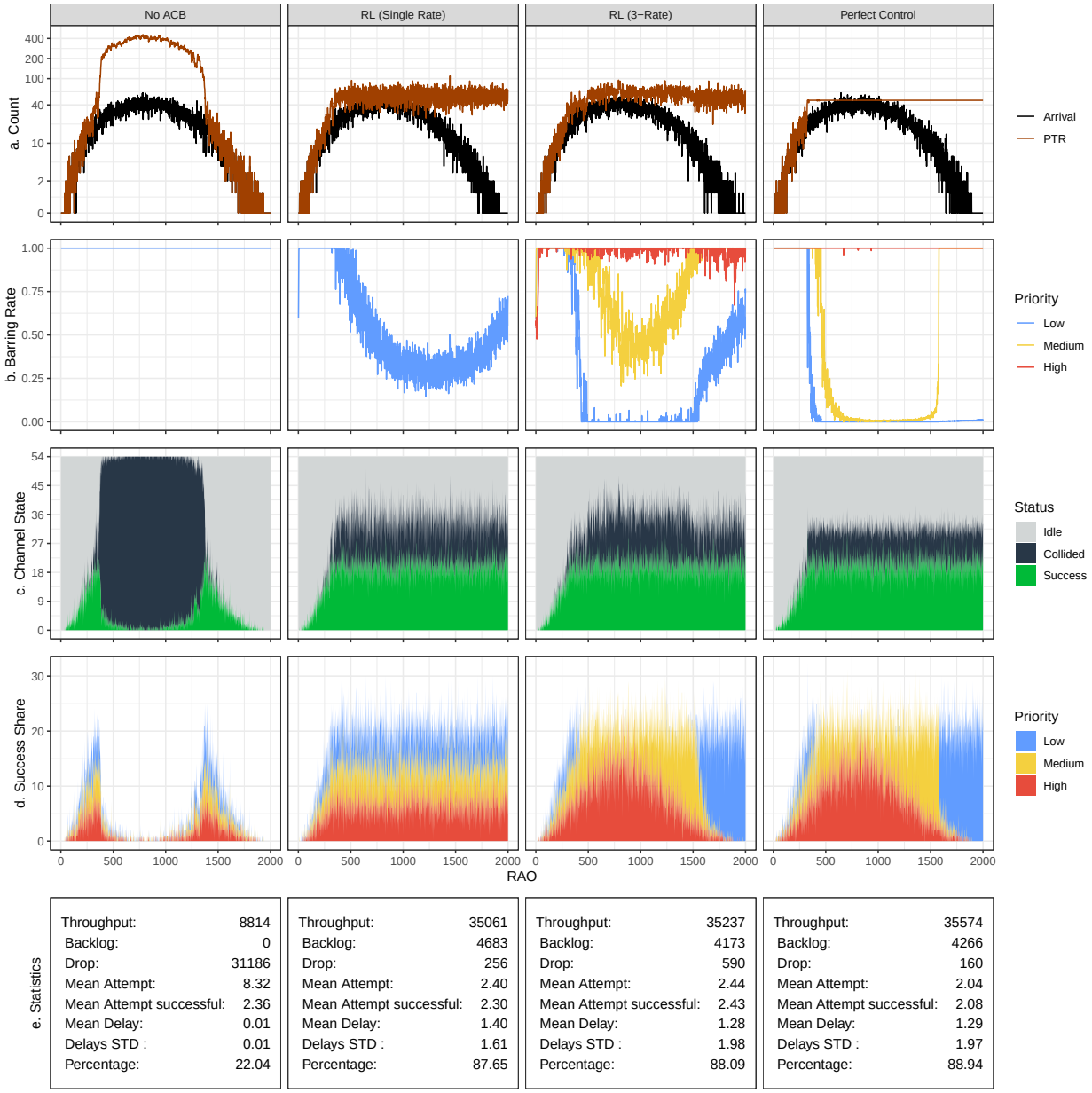

Figure 5.13 Visual comparison of different congestion control mechanisms operation over an episode which is consists of 40000 UE Beta [3, 4] distributed, with three various priority classes

challenges the efficiency of the control method in an event-driven scenario. Using the Beta(4, 6) distribution as the arrival pattern of user equipment reduces the success rate of the channel without a congestion control policy.

RL-powered congestion control mechanisms still are able to keep the maximum channel

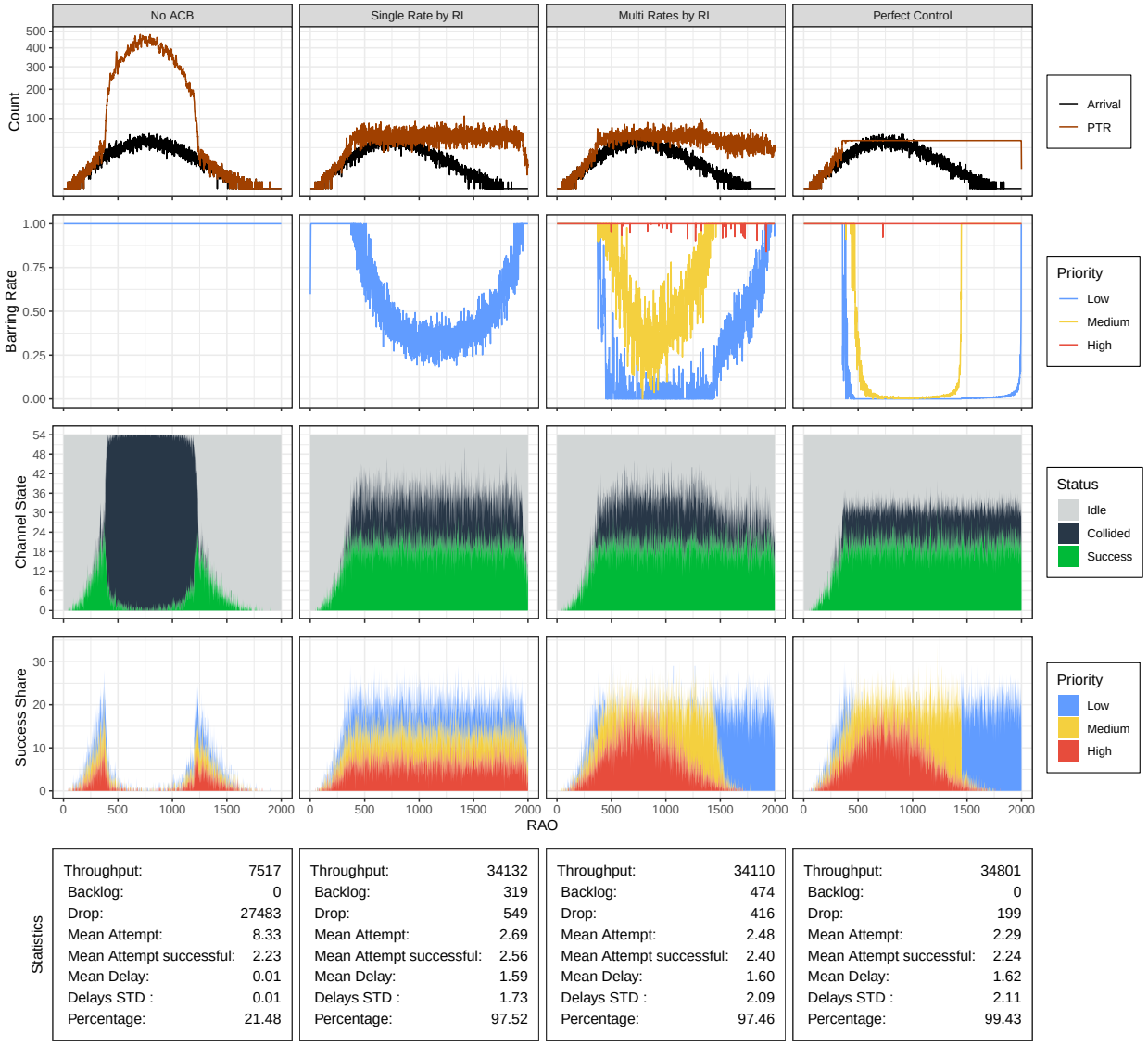

Figure 5.14 Visual comparison of different congestion control mechanisms operation over an episode which is consists of 35000 UE Beta [4, 6] distributed, with three various priority classes

usage under this pattern. Especially our proposed multi-rate approach shows persistent priority-based management. It maximizes channel utilization like the single-rate approach. Besides, it slices the access permission among the priority classes like the perfect control approach.

We also consider extremely congested scenarios under the uniform distribution, which consist of more than 40000 UEs. In the ideal case in which only 54 UE transmits preamble

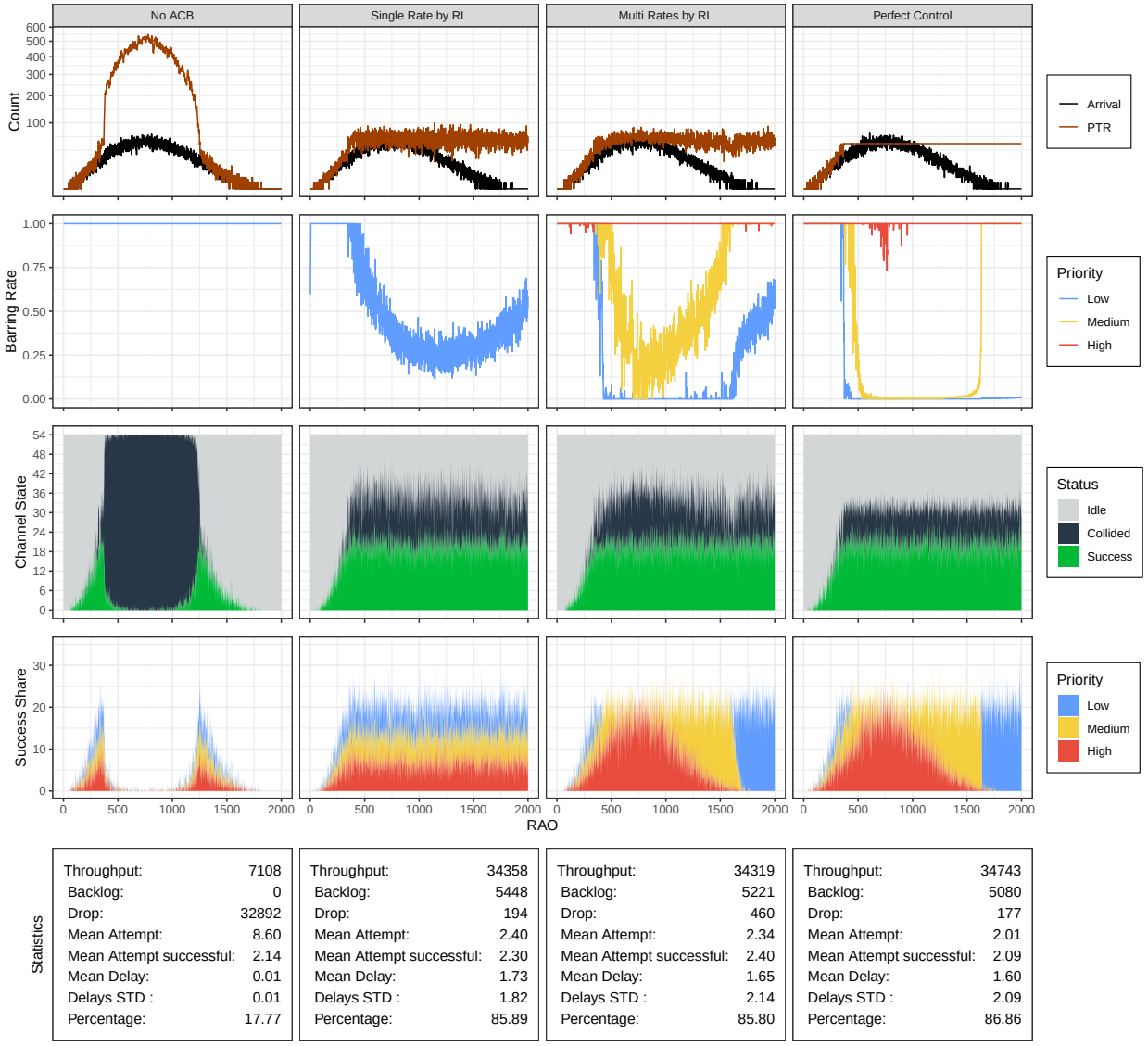

Figure 5.15 Visual comparison of different congestion control mechanisms operation over an episode which is consists of 40000 UE Beta [4, 6] distributed, with three various priority classes

simultaneously, the mean success attempt count is about 20. According to the network configuration, each episode consists of 2000 RAO. Thus in the best case, only 40000 UE can successfully establish a connection under this configuration. Figures 5.16 and 5.17 show the operation of the different control approaches under permanent overloads.

Since the failed UEs keep repeating the RACH procedure, the PTR cumulatively increases and reaches around 200 in the NO ACB scheme. Congestion control policy-based approaches

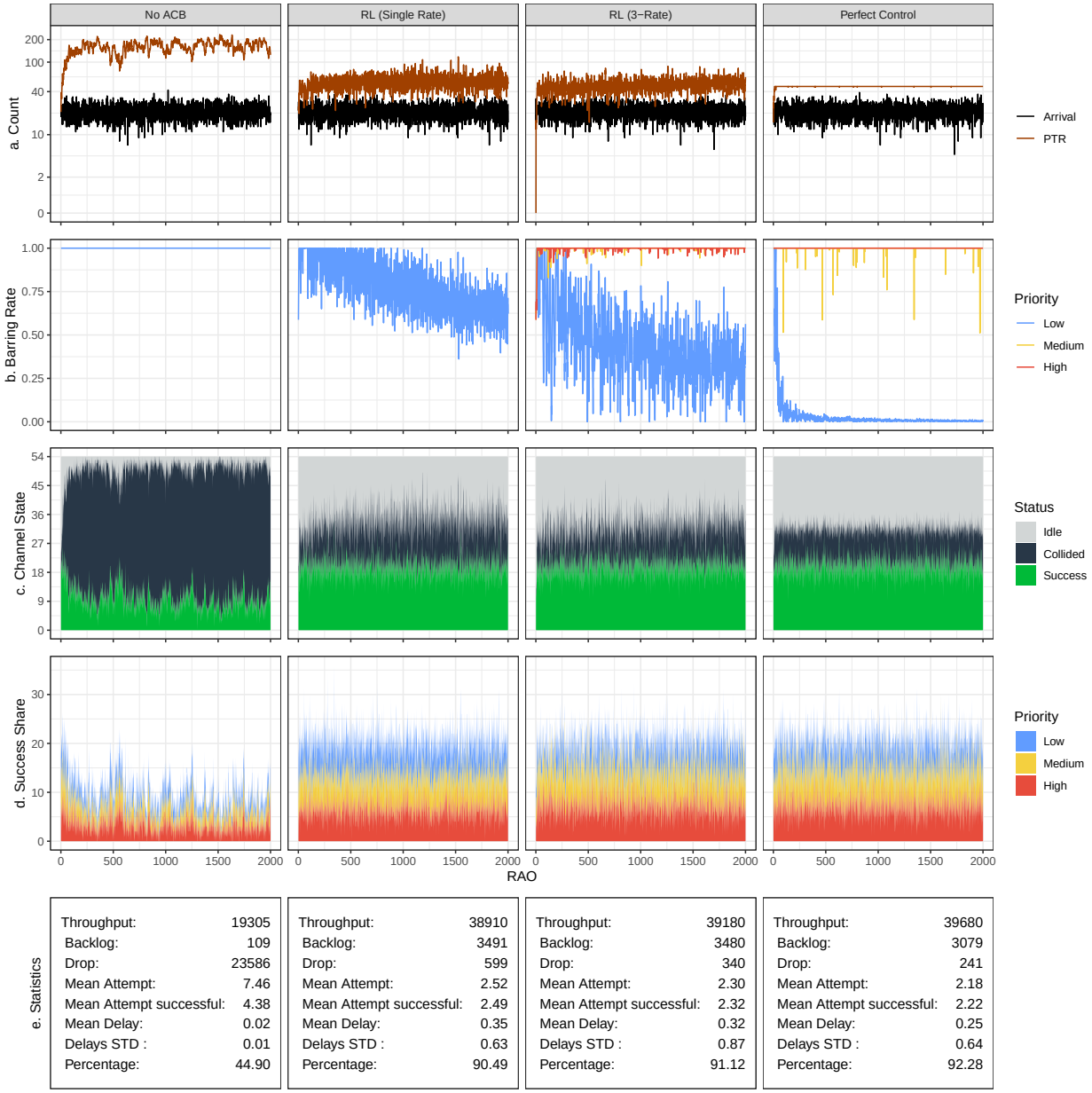

Figure 5.16 Visual comparison of different congestion control mechanisms operation over an episode which is consists of 43000 UE Uniform distributed, with three various priority classes

manage to remain the PTR in a safe boundary. In the service-differentiated schemes, only the barring rate of the low priority class is adjusted. The RL agent and the perfect control grant full access to other priority classes by announcing their barring rate as 1.

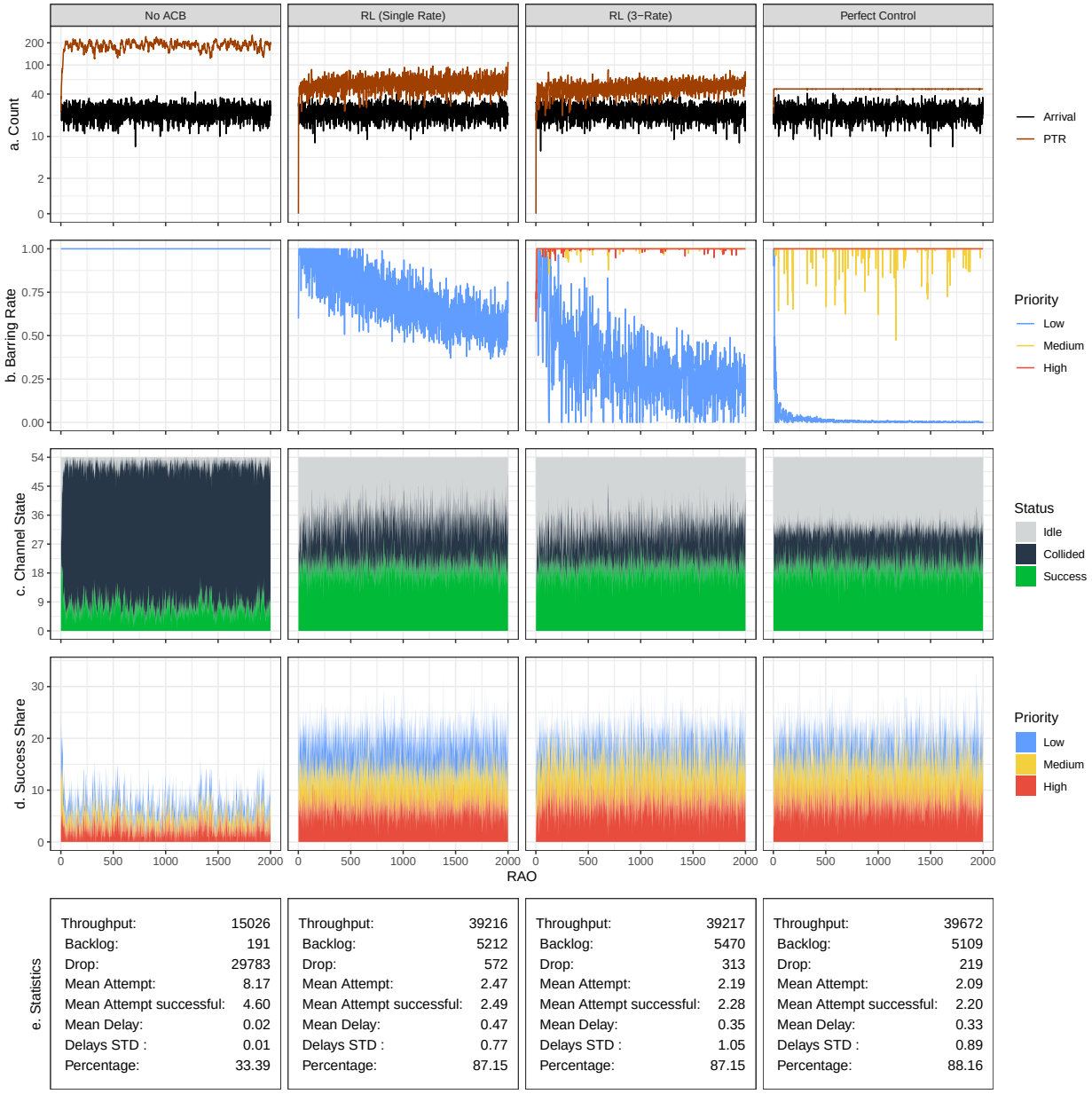

Figure 5.17 Visual comparison of different congestion control mechanisms operation over an episode which is consists of 45000 UE Uniform distributed, with three various priority classes

5.2.3. With 5 Various Priority Classes

Since service differentiation is one of the main concerns of this study, it is critical to evaluate the performance of our multi-rate approach under different numbers of coexistence priority classes. So far, we have presented the results of our RL-based ACB under an ecosystem of 3 priority classes. The multi-rate approach is used over 5, 7 and 10 distinct priority classes. In this section, we present results for a network with five coexistence priority classes. By definition, these classes are numbered from 1 to 5, from the lowest to the highest priority classes. These class numbers are used as the class coefficient in the reward function as well (Equation 5). The perfect control approach is able to manage an arbitrary number of classes in the channel, as is explained in section 5.1. While the perfect control method simulates the the-state-of-art in the comparisons, a single RL-based approach is helping to demonstrate the service differentiation.

We are content to describe only success rates and mean delays in this section since all KPIs show similar behavior to a system with three priorities. Graphical and numerical presentation of the success rate is provided in figure 5.18 and table 5.6.

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Perfect Control	**Overall**	100	99.99	99.69	99.46	99.41	99.39	89.00	100	99.87	99.48	99.40	99.39	98.96	87.70	99.90	99.82	99.69	99.40	96.78	92.43	88.28
RL (5-Rate)		99.82	99.91	99.85	99.58	99.18	97.51	86.80	99.85	99.89	99.67	99.21	98.56	96.25	83.33	99.82	99.63	97.76	93.87	90.29	87.14	84.46
RL (Single Rate)		100	99.99	99.57	98.52	98.28	97.93	87.81	100	99.87	98.69	98.42	98.35	97.27	86.24	99.90	99.80	99.56	98.88	95.68	91.02	87.03
Perfect Control	**Lowest**	100	99.99	99.69	99.52	99.49	99.46	47.56	100	99.88	99.49	99.46	99.49	97.48	41.12	99.91	99.82	99.67	99.25	86.93	65.24	44.50
RL (5-Rate)		100	99.99	99.93	99.88	99.77	94.72	44.05	100	99.96	99.92	99.83	99.71	91.55	38.66	99.62	98.86	89.88	70.60	52.83	37.46	24.57
RL (Single Rate)		100	99.99	99.56	98.52	98.27	97.93	87.77	100	99.87	98.67	98.40	98.36	97.25	86.26	99.90	99.80	99.55	98.88	95.68	91.02	87.03
Perfect Control	**Low**	100	99.99	99.69	99.42	99.42	99.39	99.39	100	99.88	99.48	99.39	99.41	99.35	99.37	99.90	99.82	99.71	99.43	99.22	99.25	99.24
RL (5-Rate)		99.08	99.57	99.64	99.33	98.80	98.21	98.07	99.26	99.64	99.36	98.83	98.67	98.20	96.60	99.90	99.82	99.68	99.59	99.55	99.43	99.19
RL (Single Rate)		100	99.99	99.57	98.53	98.30	97.92	87.84	100	99.86	98.69	98.43	98.35	97.28	86.27	99.90	99.79	99.57	98.90	95.66	91.02	87.04
Perfect Control	**Medium**	100	100	99.69	99.45	99.35	99.37	99.37	100	99.86	99.50	99.41	99.32	99.34	99.40	99.90	99.83	99.70	99.44	99.25	99.24	99.28
RL (5-Rate)		100	99.99	99.90	99.57	99.11	98.19	97.27	100	99.95	99.69	99.15	98.16	97.43	96.79	99.92	99.86	99.76	99.72	99.69	99.63	99.50
RL (Single Rate)		100	99.99	99.58	98.50	98.28	97.92	87.84	100	99.87	98.70	98.44	98.36	97.26	86.21	99.90	99.80	99.56	98.87	95.67	91.00	87.03
Perfect Control	**High**	100	99.99	99.67	99.45	99.41	99.35	99.33	100	99.88	99.45	99.38	99.35	99.32	99.35	99.91	99.83	99.69	99.43	99.27	99.24	99.24
RL (5-Rate)		100	99.99	99.90	99.56	99.12	98.22	97.28	100	99.95	99.68	99.14	98.11	97.03	92.37	99.77	99.73	99.73	99.71	99.68	99.61	99.50
RL (Single Rate)		100	99.99	99.58	98.52	98.27	97.93	87.82	100	99.87	98.69	98.43	98.33	97.27	86.28	99.91	99.80	99.55	98.89	95.68	91.04	87.05
Perfect Control	**Highest**	100	100	99.70	99.42	99.40	99.39	99.34	100	99.86	99.47	99.35	99.39	99.32	99.34	99.91	99.81	99.68	99.43	99.26	99.24	99.22
RL (5-Rate)		100	99.99	99.90	99.56	99.12	98.20	97.30	100	99.95	99.68	99.11	98.12	97.03	92.24	99.91	99.85	99.76	99.71	99.69	99.63	99.51
RL (Single Rate)		100	99.99	99.57	98.53	98.29	97.92	87.80	100	99.86	98.70	98.42	98.36	97.27	86.21	99.90	99.80	99.55	98.88	95.71	91.02	87.00

Table 5.6 Overall and priority-based success rate comparison of different congestion control mechanisms in a network consisting of 5 various priority classes

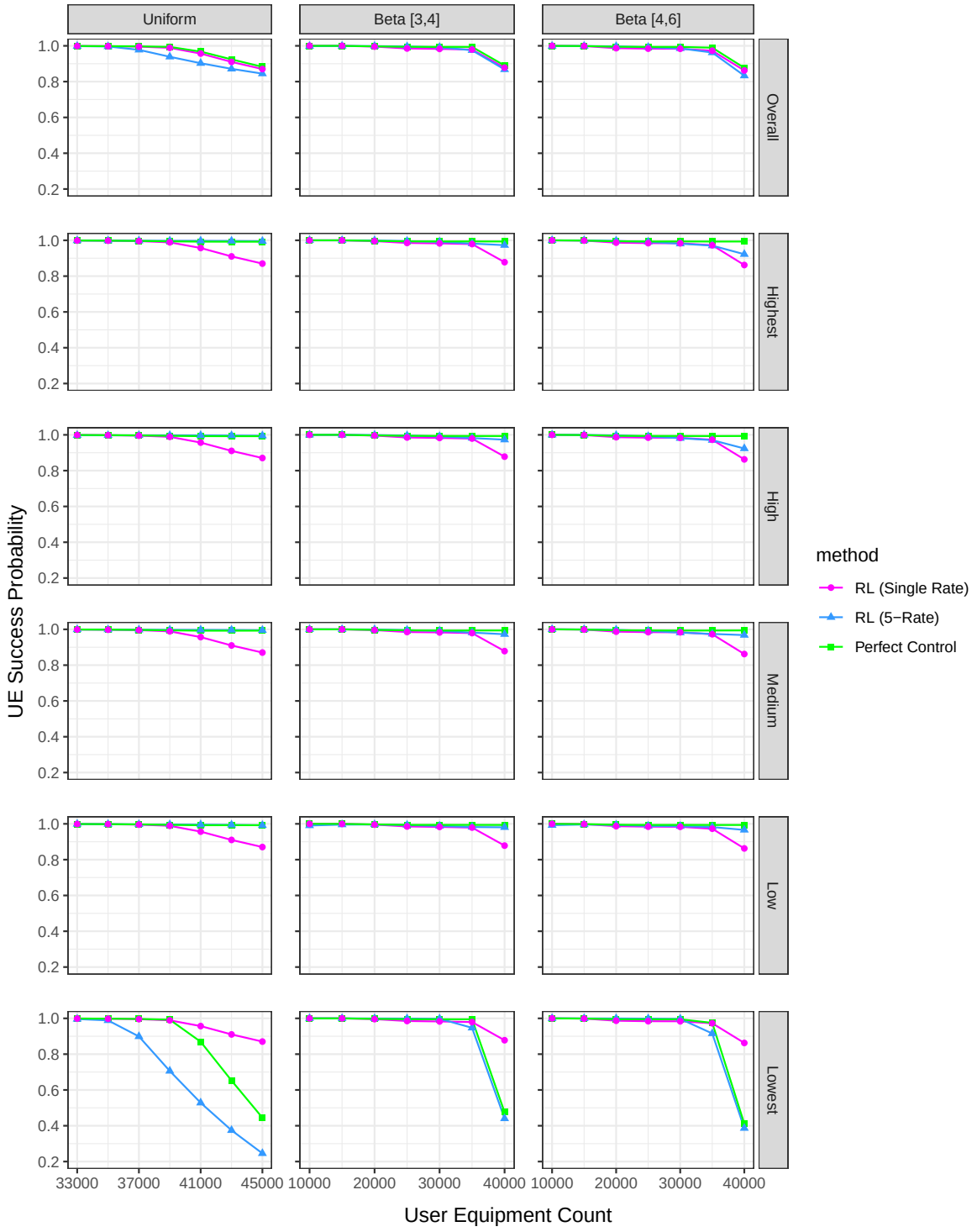

Figure 5.18 Overall and priority-based success rate comparison of different congestion control mechanisms in a network consisting of 5 various priority classes

According to the results, our Muti-Rate policy provides priority-based connection establishment as is expected. The first four higher priority classes are guaranteed full access

chance in almost all scenarios. The drop mechanism causes a negligible loss, as explained in section 5.2.2.1. with details. Meanwhile, the lowest priority class is restricted in overloaded situations like 40000 UE in Beta(3, 4) and Beta(4, 6). In comparison, in a single rate RL-based tuned approach, the success rate of all priority classes is equal.

Recorded mean delay results in the same test cases corroborate the capability of the proposed approach in priority-based traffic control over the RACH. As it can be seen in figure 5.19, the overall delay of the multi-rate scheme is close to single-rate and perfect control approaches. Class base results reveal the difference between the single rate and multi-rate methods. The UEs in the highest and high priority classes can establish a connection in less than 20 milliseconds, even in the most crowded scenarios (table 5.7). This achievement plays an important role in QoS guaranteed services. RL agent shifts the delays to lower priority classes, like in the three coexistence priority scenarios in the previous section.

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Perfect Control	**Overall**	5	6	19	275	724	1226	1264	6	8	184	604	1107	1627	1616	8	9	10	14	113	191	189
RL (5-Rate)		13	14	99	394	808	1244	1266	12	35	283	684	1160	1613	1649	22	49	148	225	233	207	165
RL (Single Rate)		6	7	24	281	720	1214	1396	6	10	186	609	1106	1609	1720	8	10	14	32	133	310	472
Perfect Control	**Lowest**	6	6	59	1298	2784	3981	4152	6	9	870	2287	3477	4568	4608	8	9	12	31	577	1284	1765
RL (5-Rate)		6	14	431	1758	2997	4081	4209	7	121	1258	2490	3613	4593	4604	70	201	755	1428	1883	2220	2450
RL (Single Rate)		6	7	24	282	720	1213	1398	6	10	188	608	1105	1609	1724	8	10	14	32	133	310	473
Perfect Control	**Low**	5	6	9	42	797	2002	3063	6	8	21	704	1849	2845	3716	8	9	9	11	12	12	14
RL (5-Rate)		22	14	27	157	954	1997	2965	19	19	109	835	1831	2739	3562	13	18	26	29	33	41	62
RL (Single Rate)		6	7	24	281	719	1213	1395	6	10	185	609	1106	1608	1721	8	10	14	32	133	310	471
Perfect Control	**Medium**	6	6	9	10	12	111	592	6	8	10	12	174	755	1482	8	9	9	10	11	11	11
RL (5-Rate)		20	14	15	23	48	215	723	17	14	21	54	282	856	1612	9	10	11	11	12	13	15
RL (Single Rate)		6	7	24	281	720	1212	1395	6	10	186	610	1107	1609	1719	8	10	14	32	133	310	473
Perfect Control	**High**	5	6	9	10	10	11	11	6	8	10	10	11	11	18	8	9	9	10	11	11	11
RL (5-Rate)		10	21	13	13	14	14	16	11	13	12	13	15	15	23	13	11	11	11	11	12	12
RL (Single Rate)		6	7	24	281	721	1214	1396	6	10	186	607	1107	1610	1719	8	10	14	32	133	311	472
Perfect Control	**Highest**	6	6	9	10	11	11	11	6	8	10	10	11	11	11	8	9	9	10	11	11	11
RL (5-Rate)		7	7	9	12	14	15	17	7	8	11	14	17	19	30	9	10	10	11	11	11	12
RL (Single Rate)		6	7	24	281	721	1215	1396	6	10	187	609	1107	1608	1718	8	10	14	32	133	310	471

Table 5.7 Overall and priority-based delay comparison of different congestion control mechanisms in a network consisting of 5 various priority classes

Our results in a network configured by 7 and 10 priority classes show the same consistency, and it can be concluded that our multi-rate RL-powered congestion control approach provides priority-based service differentiation besides utilization maximization in the RACH procedure.

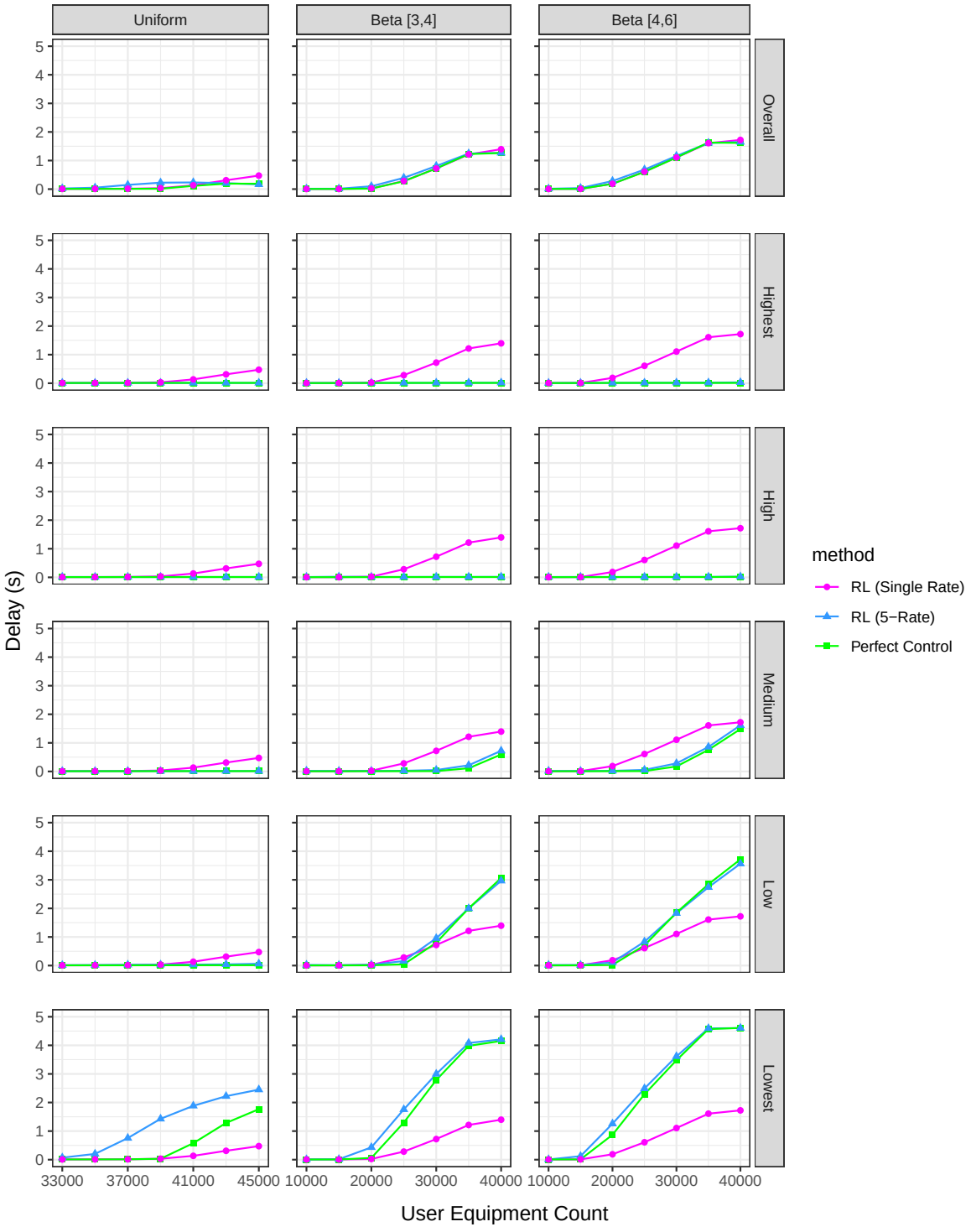

Figure 5.19 Overall and priority-based delay comparison of different congestion control mechanisms in a network consisting of 5 various priority classes

5.2.4. Barring Time Adjustment Results

The key to control the congestion is to make some of the UEs wait long enough to prevent the traffic. On the other hand, the key to minimizing the pre-connection delay time is to reduce the backoff processes time without causing an overload on the channel. Striking a balance between these two goals is of utmost importance, which brightens the significance of choosing a proper barring time. In this part of the study, we assume that three distinguished priority classes coexisted in the network. Under this assumption and the prementioned network configurations, we provide the experimental results of the following barring time adjustment approaches:

1. RL-Based Multi-Rate approach with Single Constant Barring Time
2. RL-Based Multi-Rate approach with Multi Constant Barring Time
3. RL-Based Multi-Rate approach with Multi Dynamic Barring Time
4. Multi-Rate Perfect Control

The first competitive approach is using a single constant barring time in the entire episode and all test cases. This fixed duration is determined using an exhaustive search and is applied for every three priority classes. The Multi Constant scheme uses equation 6 to determine the barring times for each priority class separately. Based on our scenario configuration, the values are calculated 0.2, 0.4 and 0.6 seconds for high, mid and low priority classes, respectively. As explained in section 4.5 , the RL agent is used to tune the barring rate and barring time parameters concurrently in the Multi Dynamic method.

Although there is no manifest theoretical approach to determine the optimum barring time at each RAO, the perfect control method can be accepted as a benchmark in this matter. In fact, the barring time in the perfect control approach is zero, and the backoff mechanism is applied using the barring rate. All the UEs are allowed to transmit their preamble if they satisfy the barring rate condition. The UEs do no enter the backoff phase at all. Instead,

the eNB broadcasts a barring rate such that exactly desired number of UEs have a preamble transmission turn, as explained in the perfect control section.

5.2.4.1. Success Rate Results Figure 5.20 demonstrates the success rate comparison of these four methods, and the detailed results are provided in table 5.8. The performance of these approaches shows diversity in different UEs distributions. In the uniformly distributed scenarios, the highest results are achieved by Multi Constant scheme after the perfect control method. For instance, in a test case of 45000 UEs, the overall success rates are 0.85, 0.86, 0.84 and 0.88 in the listed order. This slight outcome preponderance comes from the low priority class, as the high and mid priority classes results are very close. The low priority class results are 0.58, 0.60, 0.54, 0.66, while the other classes' results are higher than 0.99.

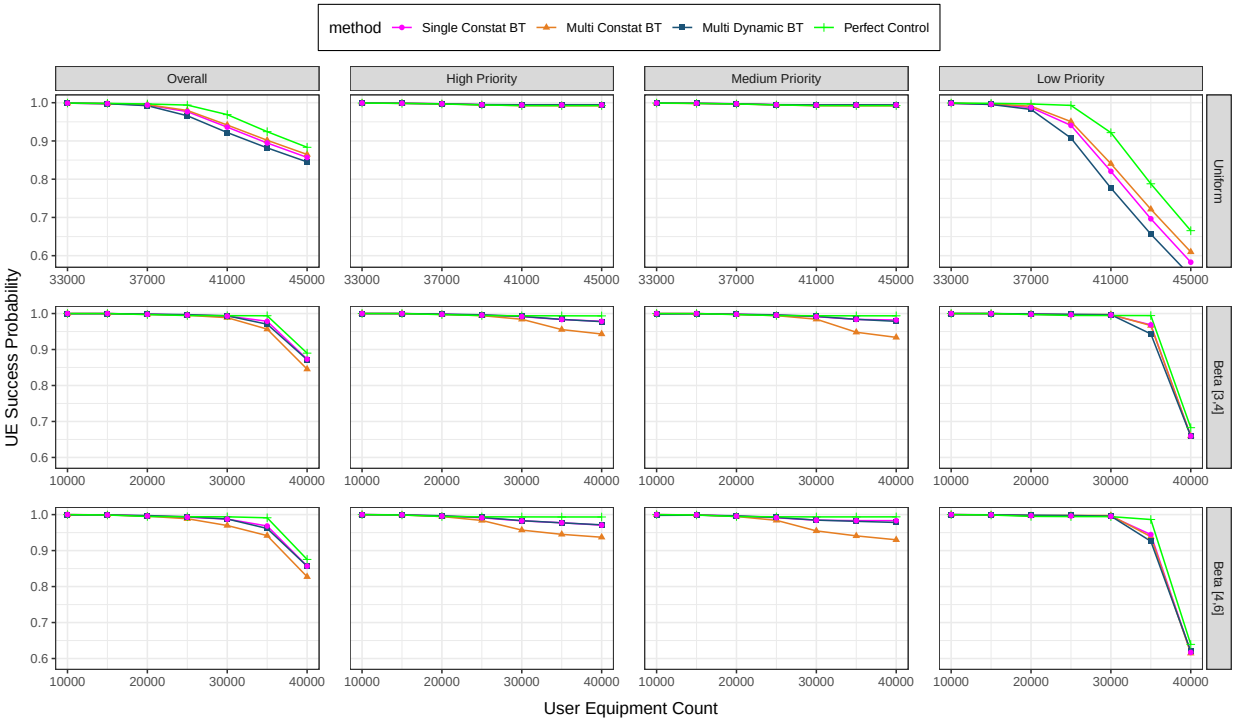

Figure 5.20 Overall and priority-based success rate comparison of different BT adjustment approaches in a network consisting of 3 various priority classes

On the other hand, Multi Constant scheme does not perform in Beta(3, 4) and Beta(4, 6) distributions as adequately as other methods. Figure 5.20 shows that, In test cases formed with over 30000 UEs, there is an explicit gap between the success rate of the Multi Constant

scheme and the other approaches. Based on the numerical results, we can conclude that the Multi Dynamic method is more successful in Beta(4, 6), while in Beta(3, 4) distribution, the Single Constant scheme works better.

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Single Constat BT	**Overall**	100	99.99	99.79	99.58	99.29	97.85	87.33	100	99.92	99.62	99.29	98.75	96.84	85.70	99.88	99.76	99.39	97.67	93.61	89.47	85.70
Multi Constat BT		100	99.99	99.78	99.50	98.85	95.67	84.55	100	99.92	99.56	98.85	96.97	94.16	82.73	99.88	99.78	99.49	97.98	94.16	90.18	86.46
Multi Dynamic BT		99.95	99.98	99.85	99.70	99.37	97.03	87.17	99.93	99.91	99.74	99.40	98.79	96.15	85.66	99.89	99.74	99.22	96.59	92.20	88.19	84.52
Perfect Control		100	99.99	99.71	99.45	99.40	99.39	88.99	100	99.89	99.48	99.41	99.39	99.12	87.55	99.91	99.82	99.67	99.38	96.89	92.43	88.34
Single Constat BT	**Low**	100	99.99	99.79	99.65	99.60	96.86	65.93	100	99.92	99.70	99.62	99.53	94.46	61.70	99.83	99.60	98.74	94.07	82.05	69.65	58.29
Multi Constat BT		100	99.99	99.81	99.68	99.72	96.66	65.96	100	99.93	99.73	99.77	99.74	93.88	61.47	99.87	99.68	99.09	95.06	84.04	72.14	60.98
Multi Dynamic BT		100	99.99	99.89	99.79	99.68	94.33	65.93	100	99.95	99.85	99.76	99.57	92.59	62.01	99.84	99.54	98.24	90.75	77.69	65.63	54.67
Perfect Control		100	99.99	99.71	99.47	99.45	99.43	68.27	100	99.89	99.50	99.46	99.44	98.64	63.93	99.91	99.83	99.67	99.30	92.17	78.80	66.54
Single Constat BT	**Medium**	100	99.99	99.78	99.55	99.13	98.37	98.20	100	99.92	99.58	99.12	98.45	98.33	98.33	99.91	99.84	99.71	99.46	99.38	99.39	99.40
Multi Constat BT		100	99.99	99.76	99.41	98.42	94.80	93.37	100	99.91	99.47	98.41	95.48	94.09	93.00	99.88	99.81	99.69	99.43	99.21	99.19	99.19
Multi Dynamic BT		99.86	99.94	99.80	99.64	99.21	98.39	97.83	99.79	99.84	99.68	99.22	98.46	98.13	97.82	99.91	99.84	99.71	99.49	99.44	99.46	99.44
Perfect Control		100	99.99	99.70	99.44	99.38	99.37	99.37	100	99.89	99.47	99.39	99.36	99.36	99.37	99.91	99.82	99.67	99.42	99.25	99.24	99.24
Single Constat BT	**High**	100	99.99	99.78	99.54	99.14	98.34	97.84	100	99.92	99.58	99.11	98.26	97.73	97.07	99.91	99.84	99.71	99.47	99.39	99.40	99.41
Multi Constat BT		100	99.99	99.76	99.41	98.40	95.54	94.31	100	99.91	99.46	98.38	95.70	94.52	93.71	99.91	99.84	99.70	99.44	99.24	99.21	99.20
Multi Dynamic BT		100	99.99	99.85	99.66	99.22	98.37	97.75	100	99.94	99.69	99.22	98.32	97.71	97.16	99.91	99.85	99.72	99.52	99.47	99.48	99.47
Perfect Control		100	99.99	99.70	99.45	99.38	99.36	99.34	100	99.89	99.48	99.39	99.36	99.34	99.34	99.91	99.82	99.67	99.42	99.25	99.24	99.25

Table 5.8 Numerical success results of different Barring Time adjustment approaches in a network consisting of 3 various priority classes

It is worth underlining that some of the test cases are exaggerated, and in suggested traffic models by 3GPP, the Multi Dynamic method achieves the closest scores to the perfect control approach. Nevertheless, other KPIs like delay have a critical impact in specifying the most suitable control approach.

5.2.4.2. Delay Results Figure 5.21 compares the mean delay results between the four barring time adjustment approaches. At first glance, it can be said that the high-priority class does not face any latency, while delays are more noticable in low priority class.

This outcome is due to service differentiation among the priority classes. The statistical results in table 5.9 show that all RL-based approaches introduce almost the same delay as the perfect control method in the high priority class.

In the mid-priority class, delay results of the Multi Constant scheme and Multi Dynamic method are lower than the Single Constant approach. For instance, in Beta(3, 4) and 40000 UEs, the perfect control has a delay of 820 ms. The Single Constant approach result is 865 ms, while Multi Constant scheme and Multi Dynamic method results are 813 and 839 ms.

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Single Constat BT	**Overall**	6	7	45	350	774	1227	1260	6	15	239	648	1138	1599	1610	10	16	33	100	216	292	324
Multi Constat BT		6	7	44	335	767	1241	1280	7	16	230	643	1155	1619	1627	11	17	35	107	218	297	332
Multi Dynamic BT		7	8	70	402	815	1217	1275	8	25	281	685	1163	1587	1622	14	29	71	172	274	327	342
Perfect Control		5	6	17	277	720	1225	1273	6	8	184	603	1104	1627	1622	8	9	10	15	117	254	321
Single Constat BT	**Low**	6	7	116	1021	2236	3338	3698	6	28	690	1854	2981	3931	4197	14	29	80	286	707	1086	1379
Multi Constat BT		6	7	114	976	2222	3352	3743	6	31	664	1848	2997	3957	4241	14	31	84	308	706	1081	1373
Multi Dynamic BT		6	8	192	1181	2381	3394	3792	6	58	819	1986	3074	3961	4270	25	68	194	528	949	1287	1548
Perfect Control		6	6	34	811	2119	3349	3769	6	8	531	1762	2944	3995	4278	8	9	11	24	347	866	1247
Single Constat BT	**Medium**	6	6	10	18	67	361	865	6	8	18	72	400	936	1562	8	9	10	14	17	18	20
Multi Constat BT		7	8	10	15	47	324	813	7	9	15	51	370	892	1514	10	10	10	12	14	14	15
Multi Dynamic BT		11	9	11	14	45	334	839	12	11	14	54	378	918	1539	8	9	11	13	14	14	14
Perfect Control		5	6	8	10	30	313	820	6	8	10	36	360	892	1522	8	9	9	10	11	11	11
Single Constat BT	**High**	6	7	9	10	11	13	13	6	8	10	11	13	13	14	8	9	10	11	11	11	11
Multi Constat BT		6	7	9	11	12	15	19	7	8	10	12	16	20	24	8	9	10	11	11	11	11
Multi Dynamic BT		6	7	9	10	11	12	13	6	8	10	11	13	13	14	8	9	10	10	10	10	10
Perfect Control		6	6	8	10	10	11	11	6	8	10	10	11	11	11	8	9	9	10	11	11	11

Table 5.9 Numerical delay results of different Barring Time adjustment approaches in a network consisting of 3 various priority classes

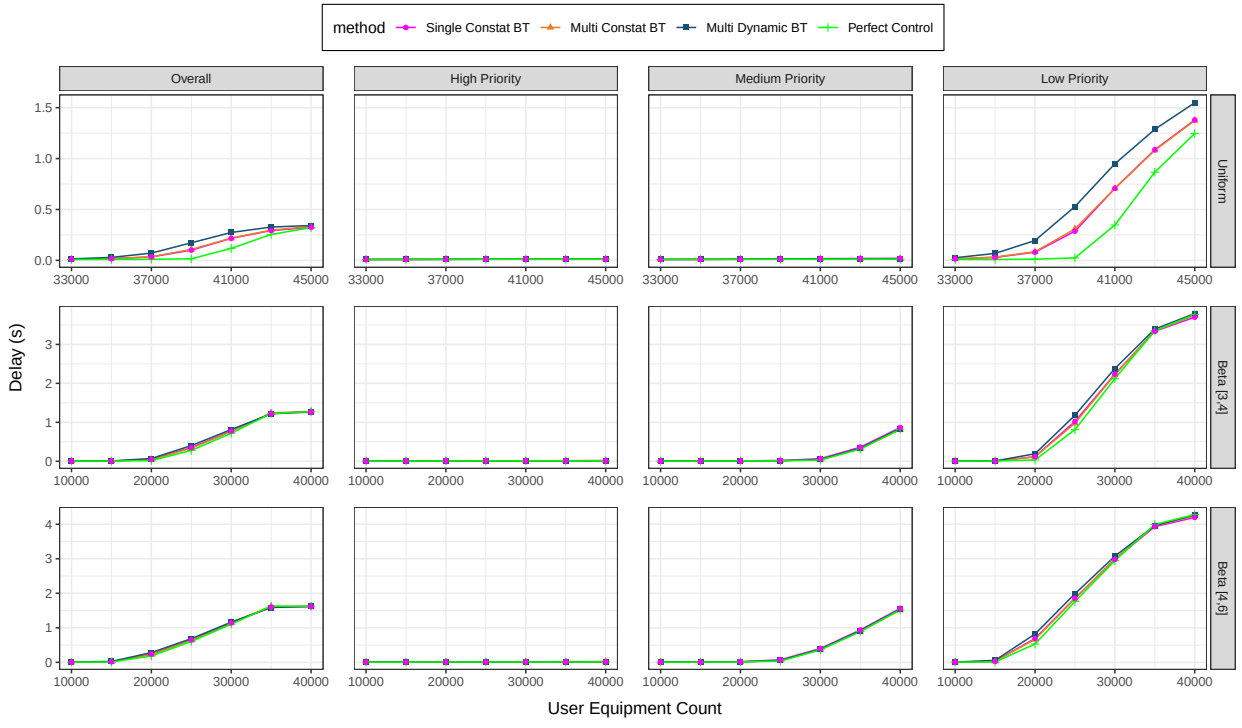

Figure 5.21 Overall and priority-based delay comparison of different BT adjustment approaches in a network consisting of 3 various priority classes

Although Multi Constant scheme causes lower delay than the others, however, the success rate of this approach was lower than the others. There is the same pattern in the Beta(4, 6) distribution with 40000 UEs. The delay results are respectively 1562, 1514, 1539 and 1522

ms. Again considering both success rates and the delay results, the multi dynamic method preforms a better control over the channel.

5.2.4.3. Delay Standard Deviation Results The delay standard deviation results are provided in figure 5.22 and table 5.10 to support the mean delay results and future examination.

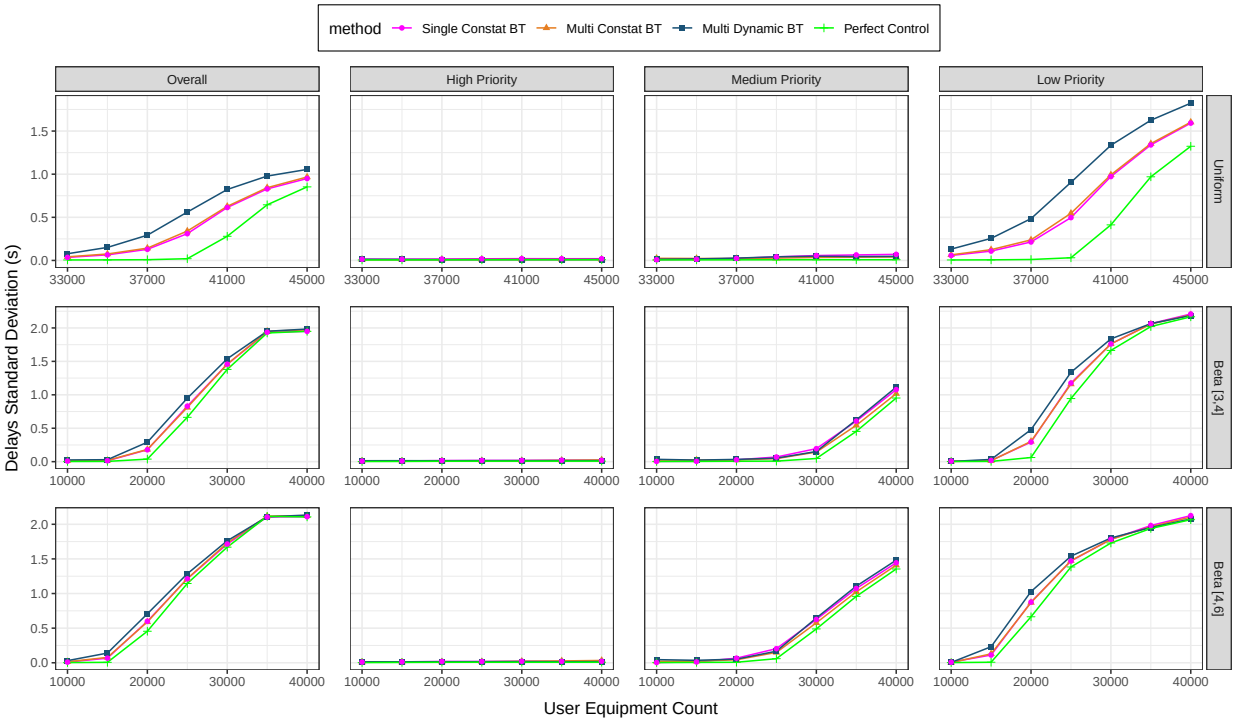

Figure 5.22 Overall and priority-based delay standard deviation comparison of different BT adjustment approaches in a network consisting of 3 various priority classes

Method	Priority	Beta [3,4]							Beta [4,6]							Uniform						
		10K	15K	20K	25K	30K	35K	40K	10K	15K	20K	25K	30K	35K	40K	33K	35K	37K	39K	41K	43K	45K
Single Constat BT	**Overall**	8	13	177	829	1457	1936	1949	9	66	599	1211	1714	2104	2109	35	64	128	310	613	829	949
Multi Constat BT		16	19	182	810	1457	1950	1971	15	77	589	1215	1729	2117	2123	41	74	142	340	627	843	967
Multi Dynamic BT		25	29	290	949	1538	1950	1983	30	140	707	1285	1761	2107	2132	77	150	292	560	822	978	1055
Perfect Control		2	3	38	663	1379	1924	1949	3	6	455	1145	1669	2114	2106	5	6	8	19	279	644	852
Single Constat BT	**Low**	8	17	293	1176	1760	2064	2208	8	111	876	1469	1782	1981	2124	57	108	214	496	972	1339	1592
Multi Constat BT		2	14	303	1160	1761	2057	2195	3	128	869	1474	1781	1964	2103	65	124	238	546	992	1354	1603
Multi Dynamic BT		3	33	478	1336	1836	2064	2185	4	231	1028	1538	1802	1953	2076	131	255	484	905	1335	1625	1823
Perfect Control		2	3	62	944	1663	2020	2166	3	7	664	1383	1729	1939	2066	5	6	10	30	412	969	1323
Single Constat BT	**Medium**	4	5	27	69	194	606	1077	4	12	66	203	625	1072	1443	8	13	23	41	56	63	70
Multi Constat BT		23	23	25	46	143	543	1017	21	22	43	149	577	1024	1409	24	22	21	27	34	37	39
Multi Dynamic BT		35	23	33	52	151	624	1114	45	35	52	166	646	1106	1481	14	17	26	42	48	45	46
Perfect Control		2	3	6	7	47	453	951	3	5	7	59	488	958	1354	5	6	6	7	8	8	8
Single Constat BT	**High**	8	10	16	18	19	22	20	10	13	19	18	21	20	21	14	14	16	18	20	19	19
Multi Constat BT		9	8	9	12	14	19	26	8	10	12	15	21	27	33	8	8	8	9	10	10	10
Multi Dynamic BT		13	14	14	15	15	15	15	13	14	15	16	17	14	16	13	11	9	9	10	10	10
Perfect Control		2	3	6	7	7	7	7	3	5	7	7	7	7	8	5	6	6	7	8	8	8

Table 5.10 Numerical standard delay deviation results of different Barring Time adjustment approaches in a network consisting of 3 various priority classes

6. CONCLUSION

By considering the rising trend of IoT devices, especially within the 5G networks, this work has targeted the congestion issue in the RACH procedure. We have proposed an RL-based Adaptive ACB to slice the RACH resources among the UEs of different priority classes. We aimed to control the congestion by intelligently adjusting the barring parameters of ACB to reduce the instant overload. In addition to the RACH utilization maximization, we considered the priority-based passage using the available information in eNB.

Our proposed method has scored 99.6 percent similar values to the results of the perfect control approach, even in extremely congested scenarios. In this way, the maximum number of handshake processes take place without causing congestion in the channel. On the other hand, provided service differentiation has been proved through the mean delay results. Regardless of the existent UE count, the delay is in the range of 8 to 11 ms for the high priority class, which overlaps with prefect control results. Using our novel approach UEs in higher priority classes can establish a connection up to 100 times faster than a single rate approach. In the medium priority class, this range is slightly higher depending on the simulation load. However, delays in mid-class are still lower than in the lower priority

class. This apparent difference among the class-based delay results proves the priority-based access control mechanism in our approach. Reducing the delays, attempts count by each UE, and therefore energy consumption has been other advantages of our proposed method. Our approach is following the standard design named Extended ACB introduced in 3GPP [12]. Consequently, easy implementation and future cellular network applicability are other encouragements.

REFERENCES

[1] Erfan Soltanmohammadi, Kamran Ghavami, and Mort Naraghi-Pour. A survey of traffic issues in machine-to-machine communications over lte. *IEEE Internet of Things Journal*, 3(6):865–884, **2016**. doi:10.1109/JIOT.2016.2533541.

[2] Philip Ball. *Patterns in nature: why the natural world looks the way it does*. The University of Chicago Press, Chicago, **2016**. ISBN 9780226332420.

[3] Ericsson. Ericsson mobility report. *https://www.ericsson.com/mobility-report*, **2020**.

[4] I. Leyva-Mayorga, L. Tello-Oquendo, V. Pla, J. Martinez-Bauset, and V. Casares-Giner. Performance analysis of access class barring for handling massive m2m traffic in lte-a networks. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6. **2016**.

[5] M. Tavana, A. Rahmati, and V. Mansouri. Congestion control with adaptive access class barring for lte m2m overload using kalman filters. *Computer Networks*, 141:222 – 233, **2018**. ISSN 1389-1286. doi:https://doi.org/10.1016/j.comnet.2018.01.044.

[6] L. Tello-Oquendo, J. Vidal, V. Pla, and L. Guijarro. Dynamic access class barring parameter tuning in lte-a networks with massive m2m traffic. In *2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pages 1–8. **2018**.

[7] L. Tello-Oquendo, D. Pacheco-Paramo, V. Pla, and J. Martinez-Bauset. Reinforcement learning-based acb in lte-a networks for handling massive m2m and h2h communications. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–7. **2018**.

[8] D. Pacheco-Paramo, L. Tello-Oquendo, V. Pla, and J. Martinez-Bauset. Deep reinforcement learning mechanism for dynamic access control in wireless networks handling mmtc. *Ad Hoc Networks*, 94:101939, **2019**. ISSN 1570-8705. doi:https://doi.org/10.1016/j.adhoc.2019.101939.

[9] Nawel Zangar, Sami Gharbi, and Marwen Abdennebi. Service differentiation strategy based on macb factor for m2m communications in lte-a networks. In *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pages 693–698. **2016**. doi:10.1109/CCNC.2016.7444864.

[10] M. Koseoglu. Pricing-based load control of m2m traffic for the lte-a random access channel. *IEEE Transactions on Communications*, 65:1353–1365, **2017**.

[11] Ziqi Chen and David B. Smith. Heterogeneous machine-type communications in cellular networks: Random access optimization by deep reinforcement learning. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6. **2018**. doi:10.1109/ICC.2018.8422775.

[12] 3GPP. Study on ran improvements for machine-type communications. In *Technical Specification*, volume TR 37.868, 11.0.0. **2011**.

[13] 3GPP. Medium access control (mac) protocol specification. In *Technical Specification*, volume TS 36.321. **2017**.

[14] Andres Laya, Luis Alonso, and Jesus Alonso-Zarate. Is the random access channel of lte and lte-a suitable for m2m communications? a survey of alternatives. *IEEE Communications Surveys Tutorials*, 16(1):4–16, **2014**. doi:10.1109/SURV.2013.111313.00244.

[15] Huda Althumali and Mohamed Othman. A survey of random access control techniques for machine-to-machine communications in lte/lte-a networks. *IEEE Access*, 6:74961–74983, **2018**. doi:10.1109/ACCESS.2018.2883440.

[16] Morteza Tavana, Vahid Shah-Mansouri, and Vincent W.S. Wong. Congestion control for bursty m2m traffic in lte networks. In *2015 IEEE International Conference on Communications (ICC)*, pages 5815–5820. **2015**. doi:10.1109/ICC.2015.7249249.

[17] Luis Tello-Oquendo, Vicent Pla, Israel Leyva-Mayorga, Jorge Martinez-Bauset, Vicente Casares-Giner, and Luis Guijarro. Efficient random access channel evaluation and load estimation in lte-a with massive mtc. *IEEE Transactions on Vehicular Technology*, 68(2):1998–2002, **2019**. doi:10.1109/TVT.2018.2885333.

[18] Lu Guan, Bo Yan, Zhiyong Guo, and Yejiang Gong. Priority-based random access control mechanism for m2m communications. In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pages 2313–2317. **2016**. doi:10.1109/CompComm.2016.7925112.

[19] John S. Vardakas, Nizar Zorba, Charalabos Skianis, and Christos V. Verikoukis. Performance analysis of m2m communication networks for qos-differentiated smart grid applications. In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6. **2015**. doi:10.1109/GLOCOMW.2015.7414116.

[20] Anh-Tuan H. Bui, Chuyen T. Nguyen, Truong Cong Thang, and Anh T. Pham. A comprehensive distributed queue-based random access framework for mmtc in lte/lte-a networks with mixed-type traffic. *IEEE Transactions on Vehicular Technology*, 68(12):12107–12120, **2019**. doi:10.1109/TVT.2019.2949024.

[21] Qi Pan, Xiangming Wen, Zhaoming Lu, Wenpeng Jing, and Haijun Zhang. Autonomous and adaptive congestion control for machine-type communication in cellular network. *International Journal of Distributed Sensor*

Networks, 15(4):1550147719841869, **2019**. ISSN 1550-1477. doi:10.1177/1550147719841869.

[22] M. R. Raza, C. Natalino, P. Öhlen, L. Wosinska, and P. Monti. A slice admission policy based on reinforcement learning for a 5g flexible ran. In *2018 European Conference on Optical Communication (ECOC)*, pages 1–3. **2018**. doi:10.1109/ECOC.2018.8535483.

[23] Helin Yang, Zehui Xiong, Jun Zhao, Dusit Niyato, Chau Yuen, and Ruilong Deng. Deep Reinforcement Learning Based Massive Access Management for Ultra-Reliable Low-Latency Communications. *IEEE Transactions on Wireless Communications*, 20(5):2977–2990, **2021**. ISSN 1536-1276, 1558-2248. doi:10.1109/TWC.2020.3046262.

[24] Yiding Yu, Taotao Wang, and Soung Chang Liew. Deep-reinforcement learning multiple access for heterogeneous wireless networks. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–7. **2018**. doi:10.1109/ICC.2018.8422168.

[25] Almuthanna Nassar and Yasin Yilmaz. Deep Reinforcement Learning for Adaptive Network Slicing in 5G for Intelligent Vehicular Systems and Smart Cities. *IEEE Internet of Things Journal*, pages 1–1, **2021**. ISSN 2327-4662, 2372-2541. doi:10.1109/JIOT.2021.3091674.

[26] Muhammad Rehan Raza, Carlos Natalino, Peter Öhlen, Lena Wosinska, and Paolo Monti. Reinforcement learning for slicing in a 5g flexible ran. *Journal of Lightwave Technology*, 37(20):5161–5169, **2019**. doi:10.1109/JLT.2019.2924345.

[27] Amaal S. A. El-Hameed and Khaled M. F. Elsayed. A Q-learning approach for machine-type communication random access in LTE-Advanced. *Telecommunication Systems*, 71(3):397–413, **2019**. ISSN 1018-4864, 1572-9451. doi:10.1007/s11235-018-0509-2.

[28] Nan Jiang, Yansha Deng, Arumugam Nallanathan, and Jonathon A. Chambers. Reinforcement Learning for Real-Time Optimization in NB-IoT Networks. *IEEE Journal on Selected Areas in Communications*, 37(6):1424–1440, **2019**. ISSN 0733-8716, 1558-0008. doi:10.1109/JSAC.2019.2904366.

[29] Jihun Moon and Yujin Lim. A Reinforcement Learning Approach to Access Management in Wireless Cellular Networks. *Wireless Communications and Mobile Computing*, 2017:1–7, **2017**. ISSN 1530-8669, 1530-8677. doi:10.1155/2017/6474768.

[30] Nan Jiang, Yansha Deng, and Arumugam Nallanathan. Deep Reinforcement Learning for Discrete and Continuous Massive Access Control optimization. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, Dublin, Ireland, **2020**. ISBN 9781728150895. doi:10.1109/ICC40277.2020.9149055.

[31] Anh-Tuan H. Bui and Anh T. Pham. Deep reinforcement learning-based access class barring for energy-efficient mmtc random access in lte networks. *IEEE Access*, 8:227657–227666, **2020**. doi:10.1109/ACCESS.2020.3045811.

[32] Bin Han and Hans D. Schotten. Grouping-based random access collision control for massive machine-type communication. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–7. **2017**. doi:10.1109/GLOCOM.2017.8254619.

[33] Serdar Vural, Ning Wang, Paul Bucknell, Gerard Foster, Rahim Tafazolli, and Julien Muller. Dynamic preamble subset allocation for ran slicing in 5g networks. *IEEE Access*, 6:13015–13032, **2018**. doi:10.1109/ACCESS.2018.2800661.

[34] Mykhailo Klymash, Halyna Beshley, Marian Seliuchenko, and Mykola Beshley. Algorithm for clusterization, aggregation and prioritization of m2m devices in heterogeneous 4g/5g network. In *2017 4th International Scientific-Practical*

Conference Problems of Infocommunications. Science and Technology (PIC S T), pages 182–186. **2017**. doi:10.1109/INFOCOMMST.2017.8246376.

[35] Ogechi Akudo Nwogu, Gladys Diaz, and Marwen Abdennebi. A combined static/dynamic partitioned resource usage approach for random access in 5g cellular networks. In *2019 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 1–6. **2019**. doi:10.23919/SOFTCOM.2019.8903770.

[36] Umesh Phuyal, Ali T Koc, Mo-Han Fong, and Rath Vannithamby. Controlling access overload and signaling congestion in m2m networks. In *2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 591–595. **2012**. doi:10.1109/ACSSC.2012.6489075.

[37] Shao-Yu Lien, Tzu-Huan Liau, Ching-Yueh Kao, and Kwang-Cheng Chen. Cooperative access class barring for machine-to-machine communications. *IEEE Transactions on Wireless Communications*, 11(1):27–32, **2012**. doi:10.1109/TWC.2011.111611.110350.

[38] Sreekanth Dama, Thomas Valerrian Pasca, Vanlin Sathya, and Kiran Kuchi. A novel rach mechanism for dense cellular-iot deployments. In *2016 IEEE Wireless Communications and Networking Conference*, pages 1–6. **2016**. doi:10.1109/WCNC.2016.7564664.

[39] Israel Leyva-Mayorga, Miguel A. Rodriguez-Hernandez, Vicent Pla, Jorge Martinez-Bauset, and Luis Tello-Oquendo. Adaptive access class barring for efficient mmtc. *Computer Networks*, 149:252–264, **2019**. ISSN 1389-1286. doi:https://doi.org/10.1016/j.comnet.2018.12.003.

[40] Shao-Yu Lien, Kwang-Cheng Chen, and Yonghua Lin. Toward ubiquitous massive accesses in 3gpp machine-to-machine communications. *IEEE*

Communications Magazine, 49(4):66–74, **2011**. doi:10.1109/MCOM.2011.5741148.

[41] Shao-Yu Lien and Kwang-Cheng Chen. Massive access management for qos guarantees in 3gpp machine-to-machine communications. *IEEE Communications Letters*, 15(3):311–313, **2011**. doi:10.1109/LCOMM.2011.011811.101798.

[42] Ming-Yuan Cheng, Guan-Yu Lin, Hung-Yu Wei, and Alex Chia-Chun Hsu. Overload control for machine-type-communications in lte-advanced system. *IEEE Communications Magazine*, 50(6):38–45, **2012**. doi:10.1109/MCOM.2012.6211484.

[43] Tarik Taleb and Andreas Kunz. Machine type communications in 3gpp networks: potential, challenges, and solutions. *IEEE Communications Magazine*, 50(3):178–184, **2012**. doi:10.1109/MCOM.2012.6163599.

[44] Chie Ming Chou, Ching Yao Huang, and Chun-Yuan Chiu. Loading prediction and barring controls for machine type communication. In *2013 IEEE International Conference on Communications (ICC)*, pages 5168–5172. **2013**. doi:10.1109/ICC.2013.6655404.

[45] Suyang Duan, Vahid Shah-Mansouri, and Vincent W. S. Wong. Dynamic access class barring for m2m communications in lte networks. In *2013 IEEE Global Communications Conference (GLOBECOM)*, pages 4747–4752. **2013**. doi:10.1109/GLOCOMW.2013.6855701.

[46] Sen-Hung Wang, Hsuan-Jung Su, Hung-Yun Hsieh, Shu-ping Yeh, and Minnie Ho. Random access design for clustered wireless machine to machine networks. In *2013 First International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pages 107–111. **2013**. doi:10.1109/BlackSeaCom.2013.6623391.

[47] Huasen Wu, Chenxi Zhu, Richard J. La, Xin Liu, and Youguang Zhang. Fasa: Accelerated s-aloha using access history for event-driven m2m communications.

IEEE/ACM Transactions on Networking, 21(6):1904–1917, **2013**. doi:10.1109/TNET.2013.2241076.

[48] German Corrales Madueño, Čedomir Stefanović, and Petar Popovski. Efficient lte access with collision resolution for massive m2m communications. In *2014 IEEE Globecom Workshops (GC Wkshps)*, pages 1433–1438. **2014**. doi:10.1109/GLOCOMW.2014.7063635.

[49] Luca Rose, E. Veronica Belmega, Walid Saad, and Mérouane Debbah. Pricing in heterogeneous wireless networks: Hierarchical games and dynamics. *IEEE Transactions on Wireless Communications*, 13(9):4985–5001, **2014**. doi:10.1109/TWC.2014.2326678.

[50] Tiago P. C. de Andrade, Carlos A. Astudillo, and Nelson L. S. Da Fonseca. Random access mechanism for ran overload control in lte/lte-a networks. In *2015 IEEE International Conference on Communications (ICC)*, pages 5979–5984. **2015**. doi:10.1109/ICC.2015.7249275.

[51] Lawal Mohammed Bello, Paul Mitchell, David Grace, and Tautvydas Mickus. Q-learning based random access with collision free rach interactions for cellular m2m. In *2015 9th International Conference on Next Generation Mobile Applications, Services and Technologies*, pages 78–83. **2015**. doi:10.1109/NGMAST.2015.22.

[52] Hongliang He, Qinghe Du, Houbing Song, Wanyu Li, Yichen Wang, and Pinyi Ren. Traffic-aware acb scheme for massive access in machine-to-machine networks. In *2015 IEEE International Conference on Communications (ICC)*, pages 617–622. **2015**. doi:10.1109/ICC.2015.7248390.

[53] Hyun-Yong Hwang, Sung-Min Oh, Changhee Lee, Jae Heung Kim, and Jaesheung Shin. Dynamic rach preamble allocation scheme. In *2015 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 770–772. **2015**. doi:10.1109/ICTC.2015.7354660.

[54] Hung-Wei Kao, You-Huei Ju, and Meng-Hsun Tsai. Two-stage radio access for group-based machine type communication in lte-a. In *2015 IEEE International Conference on Communications (ICC)*, pages 3825–3830. **2015**. doi:10.1109/ICC.2015.7248920.

[55] Guan-Yu Lin and Hung-Yu Wei. Flexible 5g m2m network access with cognitive ran: Survey and design principles. In *2015 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 36–41. **2015**. doi:10.1109/CSCN.2015.7390417.

[56] Germán Corrales Madueño, Nuno K. Pratas, Čedomir Stefanović, and Petar Popovski. Massive m2m access with reliability guarantees in lte systems. In *2015 IEEE International Conference on Communications (ICC)*, pages 2997–3002. **2015**. doi:10.1109/ICC.2015.7248783.

[57] Zehua Wang and Vincent W. S. Wong. Optimal access class barring for stationary machine type communication devices with timing advance information. *IEEE Transactions on Wireless Communications*, 14(10):5374–5387, **2015**. doi:10.1109/TWC.2015.2437872.

[58] Faezeh Morvari and Abdorasoul Ghasemi. Learning automaton based adaptive access barring for m2m communications in lte network. In *2016 24th Iranian Conference on Electrical Engineering (ICEE)*, pages 1466–1470. **2016**. doi:10.1109/IranianCEE.2016.7585752.

[59] Tiago P.C. De Andrade, Carlos A. Astudillo, Luiz R. Sekijima, and Nelson L.S. Da Fonseca. The random access procedure in long term evolution networks for the internet of things. *IEEE Communications Magazine*, 55(3):124–131, **2017**. doi:10.1109/MCOM.2017.1600555CM.

[60] Zaher Dawy, Walid Saad, Arunabha Ghosh, Jeffrey G. Andrews, and Elias Yaacoub. Toward massive machine type cellular communications. *IEEE Wireless Communications*, 24(1):120–128, **2017**. doi:10.1109/MWC.2016.1500284WC.

[61] Harpreet S. Dhillon, Howard Huang, and Harish Viswanathan. Wide-area wireless communication challenges for the internet of things. *IEEE Communications Magazine*, 55(2):168–174, **2017**. doi:10.1109/MCOM.2017.1500269CM.

[62] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 1889–1897. **2015**.

[63] Jen-Po Cheng, Chia-han Lee, and Tzu-Ming Lin. Prioritized random access with dynamic access barring for ran overload in 3gpp lte-a networks. In *2011 IEEE GLOBECOM Workshops (GC Wkshps)*, pages 368–372. **2011**. doi:10.1109/GLOCOMW.2011.6162473.

[64] Tzu-Ming Lin, Chia-Han Lee, Jen-Po Cheng, and Wen-Tsuen Chen. Prada: Prioritized random access with dynamic access barring for mtc in 3gpp lte-a networks. *IEEE Transactions on Vehicular Technology*, 63(5):2467–2472, **2014**. doi:10.1109/TVT.2013.2290128.

[65] S. Duan, V. Shah-Mansouri, Z. Wang, and V. W. S. Wong. D-acb: Adaptive congestion control algorithm for bursty m2m traffic in lte networks. *IEEE Transactions on Vehicular Technology*, 65(12):9847–9861, **2016**.