This paper has been accepted for presentation at IEEE PIMRC 2021, 13–16 September 2021.

Q-Learning-Based SCMA for Efficient Random Access in mMTC Networks With Short Packets

Duc-Dung Tran[*], Shree Krishna Sharma[*], Symeon Chatzinotas[*], and Isaac Woungang[†]

[*]*Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Luxembourg*

[†]*Department of Computer Science, Ryerson University, Toronto, Canada*

Email: {duc.tran, shree.sharma, Symeon.Chatzinotas}@uni.lu, iwoungan@ryerson.ca

***Abstract*—In massive machine-type communications (mMTC) networks, the ever-growing number of MTC devices and the limited radio resources have caused a severe problem of random access channel (RACH) congestion. To mitigate this issue, several potential multiple access (MA) mechanisms including sparse code MA (SCMA) have been proposed. Besides, the short-packet transmission feature of MTC devices requires the design of new transmission and congestion avoidance techniques as the existing techniques based on the assumption of infinite data-packet length may not be suitable for mMTC networks. Therefore, it is important to find novel solutions to address RACH congestion in mMTC networks while considering SCMA and short-packet communications (SPC). In this paper, we propose an SCMA-based random access (RA) method, in which Q-learning is utilized to dynamically allocate the SCMA codebooks and time-slot groups to MTC devices with the aim of minimizing the RACH congestion in SPC-based mMTC networks. To clarify the benefits of our proposed method, we compare its performance with those of the conventional RA methods with/without Q-learning in terms of RA efficiency and evaluate its convergence. Our simulation results show that the proposed method outperforms the existing methods in overloaded systems, i.e., the number of devices is higher than the number of available RA slots. Moreover, we illustrate the sum rate comparison between SPC and long-packet communications (LPC) when applying the proposed method to achieve more insights on SPC.**

***Index Terms*—mMTC, SCMA, Q-Learning, short-packet communications.**

I. Introduction

Massive machine-type communications (mMTC) has recently been demonstrated as a potential paradigm for the fifth generation (5G) and beyond wireless networks (5GBNs) to support the connection of billions of devices [1]. According to IHS Markit forecast, there will be around 125 billions of devices by 2030 connected in a variety of applications such as industrial automation, remote surgery, intelligent transportation, smart city, and vehicle-to-everything communications [2]. However, the ever-increasing number of devices has led to critical challenges for mMTC networks, especially random access channel (RACH) congestion [1, 2]. Besides, short-packet communications (SPC) has been considered as a promising solution to achieve the stringent requirements of reliability and latency for novel applications in ultra-dense cellular networks such as mMTC systems [3]. Nevertheless, this requires the redesign of the communication protocols because the traditional transmission methods, designed based on Shannon theorem utilizing long data-packets, may not be suitable for the analysis of mMTC networks with SPC [4]. Therefore, investigating effective solutions to support SPC and address the RACH congestion in mMTC networks is a crucial problem to be tackled [5, 6].

In recent years, the application of machine learning (ML) for RACH congestion problem has received a great attention due to their capability of learning the system variations and accordingly adapting the system parameters [1]. Among ML techniques, Q-learning seems promising for MTC devices since it is model-free and can be realized in a distributed manner [7]. Given this context, some recent works on random access (RA) methods based on Q-learning have been conducted [5, 6, 8–11], which are briefly summarized in the following.

The authors in [8] examined Q-learning algorithm for intelligent time-slot allocation enabling the coexistence of human-type-communication (HTC) and MTC devices in a cellular network. The work in [9] investigated a method of selecting the best BS for MTC devices based on Q-learning. In addition, the authors in [10] proposed a collaborative distributed Q-learning algorithm for frame-based slotted-Aloha (SA) RA scheme to avoid RACH congestion and improve the throughput using the collision level per time-slot to build a reward function. In [11], non-orthogonal multiple access (NOMA) and Q-learning were considered to reduce the RACH congestion. However, the works [8–11] did not consider the transmission constraints due to SPC, which is considered an enabling paradigm to reduce latency in 5G and beyond applications. Taking SPC into account, Han *et al.* [5] proposed a power allocation method to maximize the energy efficiency and addressed the problem of subchannel allocation to the devices in NOMA-based mMTC networks by using Q-learning. In [6], an adaptive Q-learning scheme was proposed to mitigate RACH congestion in NOMA-based mMTC networks, where block error rate (BLER) of devices has been used as a global cost in the learning process. Nevertheless, the works [5, 6] did not investigate the combination of Q-learning and sparse code multiple access (SCMA) to minimize the congestion and improve the system performance in terms of RA efficiency (RAE) and sum rate. Note that SCMA is a promising technique for 5GBNs, especially mMTC networks, since it can enable multi-user communication with massive connectivity by exploiting an additional degree of freedom in the code domain [12, 13].

In this paper, we propose a novel RA method using Q-learning and frame-based SA, namely QL-SCMA-SA, to mitigate the RACH congestion and improve the system performance in SPC-based mMTC networks, in which SCMA is employed. In SCMA, the information bits are mapped into the multi-dimensional codewords from a predefined codebook and each device has a different codebook. The careful design of codebooks can result in a higher system performance [12, 13]. The main contributions of this paper are summarized as follows: i) we propose a QL-SCMA-SA method to optimize the joint allocation of SCMA codebooks and time-slots for the devices to address the RACH congestion in SCMA-based mMTC networks with SPC; ii) we evaluate and compare the RAE of our proposed method with some conventional techniques in the literature; iii) we perform the sum rate comparison between SPC and long-packet communications (LPC) when applying the proposed method to obtain more insights on SPC.

The remainder of the paper is organized as follows. Section II presents the system model in detail. Section III describes the Q-learning algorithm and the proposed QL-SCMA-SA method for effective RA in SCMA-based mMTC networks with SPC. Section IV presents the numerical results. Finally, Section V concludes this paper.

II. System Model

In this paper, we consider SPC in an uplink SCMA-based mMTC network, as depicted in Fig. 1. The network consists of one base station (BS) and M MTC devices deployed randomly around the BS within the radius $\mathcal{R}$. In this setting, the devices communicate with the BS using an SCMA-based SA (SCMA-SA) protocol [12]. In the considered SCMA scheme, multiple devices are multiplexed over K shared orthogonal resources (i.e., time-slots in this case) and each device uses one SCMA codebook [14]. The K-dimensional codewords of a codebook are sparse vectors with N ($N \leq K$) non-zero elements. This implies that each device uses N out of K available time-slots. Thus, there are $C_b = \frac{K!}{(K-N)!N!}$ different codebooks/devices sharing the same K-time-slot group. For example, with $K = 4$ and $N = 2$, we have $C_b = 6$, as shown in Fig. 1. Therefore, at most six devices can use the same 4-time-slot group in this case. When K increases, more devices can be served by properly selecting the value of N. However, the implementation complexity of SCMA also becomes higher when K and N increase.

With the SCMA-SA protocol, each device transmits its single packet to the BS within a frame by selecting randomly one codebook and one of $\mathcal{T}_g$ K-time-slot groups, where $\mathcal{T}_g = T/K$ and T denotes the number of available time-slots. After each frame, the BS sends a feedback bit to inform the devices if their transmissions are successful or not [10, 11]. The devices can use this control message for synchronization purpose [11]. Note that in the conventional SA method, each time-slot can be used by only one device, otherwise a collision is observed. Thus, with K-time-slot group, the SA method can serve at most K devices. In contrast, a K-time-slot group can be allocated to $C_b > K$ devices by using SCMA-SA method. Therefore, this method can reduce the collision in overloaded systems, i.e., the number of devices is larger than the number of available time-slots.

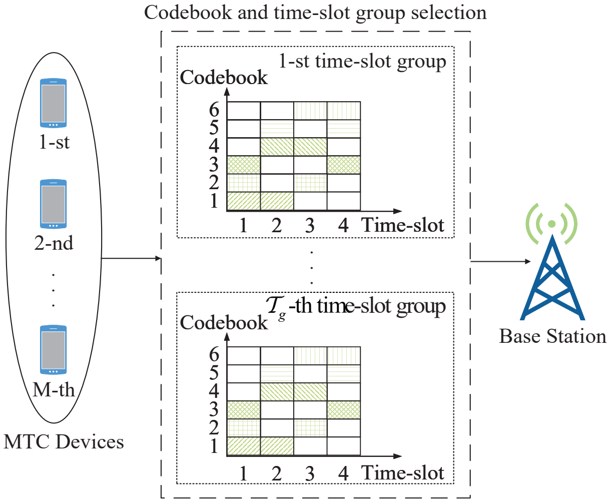

Fig. 1. An illustration of an uplink SCMA-based mMTC network under SPC, where 4 time-slots are used to form a 4-dimensional codeword of a codebook with two non-zero entries.

Let us assume that there are $\hat{M}$ ($\hat{M} \leq M$) devices using the same t-th ($1 \leq t \leq \mathcal{T}_g$) K-time-slot group to transmit their messages to the BS simultaneously by applying SCMA. Accordingly, the received signal at the BS in the t-th time-slot group of the i-th frame is given by

$$\mathbf{y}_t\left(i\right) = \sum_{m=1}^{\hat{M}} \sqrt{\frac{P_m}{N d_m^{\theta}}} \mathrm{diag}\left(\mathbf{f}_{m,t}(i)\right) \mathrm{diag}\left(\mathbf{h}_{m,t}(i)\right) \mathbf{x}_{m,t}\left(i\right) \\ + \mathbf{w}_t\left(i\right), \tag{1}$$

where P_m is the transmit power of the device m; $d_m < \mathcal{R}$ is the distance from the device m to the BS; θ is the path loss exponent; $\mathbf{x}_{m,t}\left(i\right) = \left[x_{m1,t}\left(i\right) \ \ldots \ x_{mK,t}\left(i\right)\right]^T$ is the codeword of the device m; $\mathbf{h}_{m,t}(i) = \left[h_{m1,t} \ \ldots \ h_{mK,t}\right]^T$ represents the channel coefficient vector of the device m, which is assumed to be constant during frame i under a quasi-static scenario [11]; $\mathbf{w}_t\left(i\right) \sim \mathcal{CN}\left(0, \mathbf{I}\sigma^2\right)$ is the additive white Gaussian noise (AWGN); and $\mathbf{f}_{m,t}(i) = \left[f_{m1,t}(i) \ \ldots \ f_{mK,t}(i)\right]^T$ denotes the binary indicator vector of the device m, where $f_{mk,t}(i)$ is the time-slot index with $f_{mk,t}(i) = 1$ if $x_{mk,t}(i) \neq 0$, and $f_{mk,t}(i) = 0$ if $x_{mk,t}(i) = 0$. From (1), the effective signal-to-noise ratio (SNR) of the device m is given by [13, 14]

$$\gamma_m = \prod_{k=1}^{K} \left(1 + \beta \hat{\gamma}_{m,k}\right) - 1, \tag{2}$$

where β indicates the multi-user interference cancellation capability, which is recommended to be from 0.82 to 0.97 [13, 14]; and $\hat{\gamma}_{m,k} = \frac{P_m f_{mk,t} |h_{mk,t}|^2}{N d_m^{\theta} \sigma^2}$ denotes the effective SNR of the device m on the resource block (i.e., time-slot) k under perfect interference cancellation scenario.

Taking SPC into account, the achievable rate of the device m can be approximated as [3]

$$R_m \approx log_2\left(1+\gamma_m\right) - \sqrt{\frac{v_m}{B_m}}Q^{-1}\left(\varepsilon_m\right), \tag{3}$$

where $Q^{-1}\left(\varepsilon_m\right)$ is the inverse of the Gaussian Q-function $Q\left(x\right) = \int\limits_{x}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$, $v_m = \left(\log_2 e\right)^2 \left[1 - \frac{1}{\left(1+\gamma_m\right)^2}\right]$ represents the channel dispersion, B_m denotes the blocklength, and ε_m is the decoding error probability. Note that we consider a scenario, where a successful transmission of a message from the device m occurs when R_m is larger than or equal to a minimum rate threshold, R_{th}, i.e.,

$$R_m \geq R_{th}. \tag{4}$$

III. Proposed Method for Effective Random Access

In this section, we present the proposed QL-SCMA-SA algorithm, which allows the MTC devices to autonomously find the best codebook and time-slot group for their short-packet transmissions to avoid the RACH congestion by leveraging the SCMA spectral efficiency and Q-learning algorithm.

In the proposed method, we use SCMA-SA protocol for RA in mMTC networks to improve the system performance and reduce the collision in overloaded scenario, as depicted in Section II. However, with the RA nature, the devices selecting the same K-time-slot group can choose the same SCMA codebook, leading to a collision. To overcome this issue, we utilize Q-learning algorithm for the RA based on SCMA-SA protocol to optimize the codebook and time-slot group allocation for the devices and reduce the collision.

Q-learning is one of the most popular reinforcement learning algorithms, which enables agents to learn from the previous experiment in order to achieve successful strategies by interacting with the environment [10]. In this method, an agent can move from the current state to the next state by carrying out an action through an interaction with the surrounding environment and then receives an associated reward [7]. We employ the Q-learning method in the investigated SCMA-based mMTC network to mitigate the RACH congestion by considering the following system set-up: the agents are the devices, the environment is the investigated network, and the state-action pair is the combination of the codebook and time-slot group. In particular, the state of a device m at time step u is $s_{u,m}(c,t) \in \mathbb{S}$ $(1 \leq c \leq C_b$ and $t \leq t \leq \mathcal{T}_g)$ if it occupies a (codebook, time-slot group) pair (c,t) for its transmission. An action $a_{u,m}(s_{u,m}, s_{u+1,m}) \in \mathbb{A}$ at time step u is a transition from the current state $s_{u,m}$ to the next state $s_{u+1,m}$, which is a request of the device to move from one (codebook, time-slot group) pair to another (codebook, time-slot group) pair.

By using Q-learning, device m creates its own Q-table, which indicates the relationship between the agent and the environment. To do this, it can utilize a greedy policy to select the actions. Specifically, at time step u and state $s_{u,m}$, device m performs an action $a_{u,m}$ that has the largest Q-value. After taking action $a_{u,m}$, it receives a reward $r_{u+1,m}$ and moves to the next state $s_{u+1,m}$. Accordingly, the Q-value of the state-action pair $(s_{u,m}, a_{u,m})$ can be updated by means of an iterative procedure as follows [7, 10]:

Algorithm 1: Proposed QL-SCMA-SA algorithm for minimizing the RACH congestion in SCMA-based MTC Networks.

```
Data  : $M$, $T$, $P_m$, $B_m$, $K$, $N$, $\varepsilon_m$, $\mathcal{R}$, $R_{th}$, number of
        iterations/frames for learning process $J$.
Result: Q-Table for $M$ devices.
Initialize $C_b \times \mathcal{T}_g$ zero Q-table for all devices, $j \leftarrow 1$;
while $j \leq J$ do
    Device $m$ $(1 \leq m \leq M)$ selects an action $a_m$, i.e.,
      selecting a (codebook, time-slot group) pair for its
      transmission, with highest Q-value;
    if $size(a_m) > 1$ then
        Choose from $a_m$ an action randomly;
    end
    Take action $a_m$, get the reward according to (6);
    Update the Q-value according to (5);
    $j \leftarrow j + 1$;
end
```

$$\begin{aligned} &Q_{u+1,m}\left(s_{u,m}, a_{u,m}\right) \\ &\quad = \left(1-\alpha\right) Q_{u,m}\left(s_{u,m}, a_{u,m}\right) \\ &\quad\quad + \alpha\left[r_{u+1,m} + \gamma \max_{a\in\mathbb{A}} Q_{u,m}\left(s_{u+1,m}, a\right)\right], \end{aligned} \tag{5}$$

where $0 \leq \alpha \leq 1$ denotes the learning rate; $0 \leq \gamma \leq 1$ is the discount factor; and $r_{u+1,m}$ represents the reward function, which is defined based on the achievable rate and transmission outcome of the device m as follows:

$$r_{u+1,m} = R_m - p_{f,m}, \tag{6}$$

where $p_{f,m} = 1$ if $R_m < R_{th}$, otherwise $p_{f,m} = 0$.

Algorithm 1 presents the detail of the proposed QL-SCMA-SA method to minimize the RACH congestion in SPC-based mMTC networks, where each device uses the greedy policy to build its own Q-table. Specifically, the Q-value for each possible (codebook, time-slot group) pair is first initialized to zero. The devices then choose randomly a (codebook, time-slot group) pair and transmit their messages to the BS by using the selected (codebook, time-slot group) pair. At the end of the frame, the devices update their Q-table utilizing Equations (5) and (6) based on their transmission outcomes. Each device then selects a (codebook, time-slot group) pair with the highest Q-value to transmit in the next frame. This process ends when the convergence is eventually observed, where each device finds a unique (codebook, time-slot group) pair for its transmission.

To evaluate the performance of the proposed QL-SCMA-SA method, we use two performance metrics, namely RAE and sum rate (i.e., aggregate system throughput). Specifically, RAE is defined as the number of successful transmissions over the number of available time-slots within a frame. And, the sum

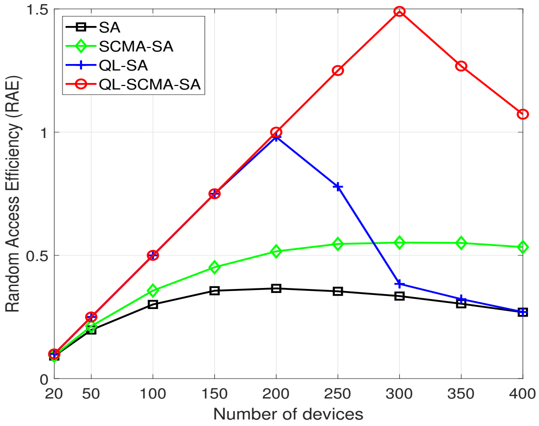

Fig. 2. RAE versus number of devices with different RA methods.

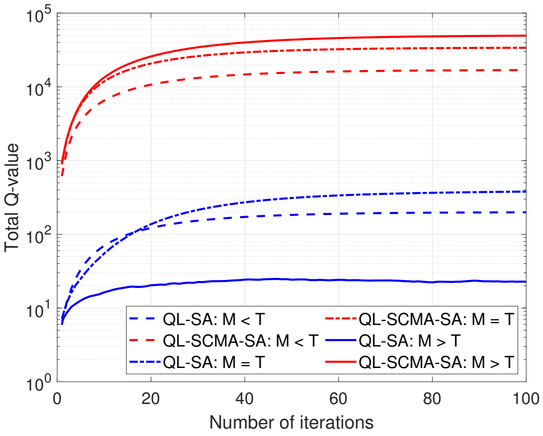

Fig. 3. Total Q-value of the considered RA methods based on Q-learning.

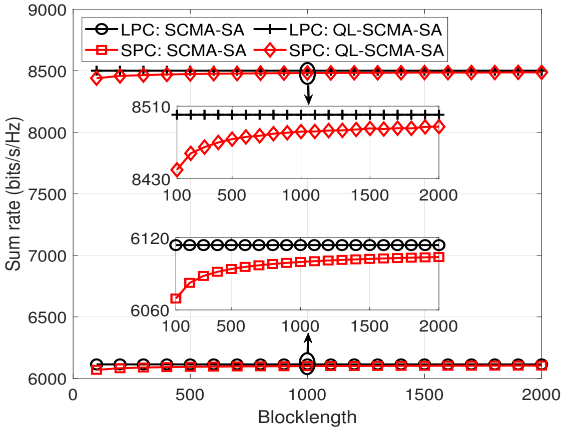

Fig. 4. Sum rate comparison between SPC and LPC, where $M = 100$.

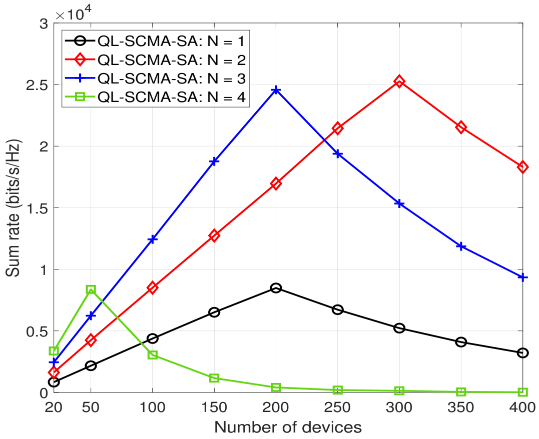

Fig. 5. Sum rate versus number of devices with the different values of N.

rate is calculated as the summation of achievable rates of all MTC devices.

IV. Numerical Results

In this section, we provide numerical results for the performance analysis of the proposed QL-SCMA-SA method in terms of RAE and the sum rate. For the proposed Q-learning algorithm, we set the parameters as follows [10, 11]: the learning rate $\alpha = 0.1$, the discount factor $\gamma = 0.5$, and the reward function is defined in (6). In addition, the predetermined simulation parameters are set as follows unless otherwise stated [5, 11, 13]: the time-slots $T = 200$; the transmit power for all devices $P = 10$ dBm; the cell radius $\mathcal{R} = 120$ m; the path loss exponent $\theta = 3$; the noise variance $\sigma^2 = -174$ dBm; the minimum rate threshold $R_{th} = 2$ bits/s/Hz; the blocklength for all devices $B = 300$; the decoding error probability for all devices $\epsilon = 10^{-5}$; the number of time-slots used for a codeword $K = 4$; and the number of non-zero entries of a codeword $N = 2$.

Fig. 2 depicts the system performance in terms of RAE versus the number of devices (M) with different RA methods. Specifically, we compare our proposed QL-SCMA-SA method against the SA, SCMA-SA, and QL-SA schemes. In SA, each device selects one time-slot for its transmission randomly and a collision occurs when there are more than one devices choosing the same time-slot. In SCMA-SA, SCMA is applied to mitigate the collision. In QL-SA, Q-learning is used for the SA method to help the devices select the time-slots effectively without considering SCMA. Note that this method uses the binary reward for the learning process, where $r_m = -1$ if the collision occurs (i.e., there are more than one devices selecting the same time-slot) and $r_m = 1$ otherwise as in [10, 11].

We can observe from Fig. 2 that SCMA brings a better performance for the SA scheme. Furthermore, the methods using Q-learning (i.e., QL-SA and QL-SCMA-SA) significantly improve the system performance as compared to the cases without Q-learning (i.e., SA and SCMA-SA). It is also noted that among the considered methods, our proposed QL-SCMA-SA scheme yields the better RAE based on the combination of Q-learning and SCMA. With QL-SA, the best RAE is achieved when $M = T$ because each time-slot can serve only one single device and the collision can be observed when $M > T$, leading to the significant performance reduction. In contrast, our proposed QL-SCMA-SA method can further improve the RAE when $M > T$. This is because more than one devices can be served on each time-slot in SCMA, which improves the achievable rates of devices and hence increasing the RAE. Under the investigated conditions, our proposed method achieves the best RAE when $M = 1.5T$.

To illustrate the convergence of the considered Q-learning methods (i.e., QL-SA and QL-SCMA-QL), Fig. 3 plots the total Q-value versus the number of iterations in the following three cases: $M < T$, $M = T$, and $M > T$, where $M \in \{100, 200, 300\}$ and $T = 200$. This figure indicates that the total Q-value of each presented case gradually converges to a certain value, hence, the convergence can be observed. Accordingly, the devices can find unique time-slots and (codebook, time-slot group) pairs for their transmissions when using the QL-SA and QL-SCMA-SA methods, respectively. Moreover, when $M > T$, the achieved total Q-value of the QL-SA scheme is lower than the case $M \leq T$, resulting in the performance reduction. In contrast, our proposed QL-SCMA-SA method achieves the higher total Q-value when $M > T$ as compared to the case $M \leq T$, making the performance better. This confirms the observations from Fig. 2, where our proposed method outperforms the QL-SA when $M > T$.

It should be noted that the data size transmitted by the MTC devices may be small [2]. Therefore, long-packet communications (LPC) based on Shannon theorem may no longer be suitable for mMTC networks with SPC. In Fig. 4, we perform the sum rate comparison between SPC and LPC when applying our proposed QL-SCMA-SA method and the SCMA-SA scheme to achieve more insights on SPC. This figure shows that the sum rate when using LPC ($\hat{R}_{LPC}$) is unchanged with the increase in the blocklength. This is because $\hat{R}_{LPC}$ is calculated based on the Shannon theorem and hence it does not depend on the blocklength. In contrast, based on (3), the sum rate when using SPC ($\hat{R}_{SPC}$) increases as the blocklength increases and $\hat{R}_{SPC} < \hat{R}_{LPC}$. Thus, SPC can help mMTC networks to fulfill the stringent reliability and latency requirements of new applications, but also leads to a rate performance degradation.

Fig. 5 shows the effects of the number of non-zero entries of a codeword (N), on the sum rate of the proposed QL-SCMA-SA method. It is observed that for small M, the larger sum rate can be achieved with the increase in the value of N since the effective SNR increases as per Equation (2). When M gets larger, the sum rate increases for all cases of N and achieves the peak value at $M = 50$ for $N = 4$, $M = 200$ for $N = 1$ and $N = 3$, and $M = 300$ for $N = 2$. This is because the maximum number of devices sharing the same K-time-slot group $C_b = 6$ when $N = 2$, $C_b = 4$ when $N = 1$ and $N = 3$, and $C_b = 1$ when $N = 4$. Thus, more devices can be served by one time-slot group when $N = 2$, leading to the higher sum rate in high M area as compared to the remaining cases of N, i.e., $N \in \{1; 3; 4\}$. Meanwhile, the worst case occurs when $N = 4$ due the fact that one time-slot group can only serve the maximum of one device (i.e., $C_b = 1$), which in turn results in a higher possibility of collision when M grows up.

V. Conclusion

In this paper, we have proposed a QL-SCMA-SA method for SPC-based mMTC networks to minimize the RACH congestion based on Q-learning and SCMA. The proposed method enables the MTC devices to find the best SCMA codebook and time-slot group for their transmissions dynamically in order to avoid the RACH congestion. The achieved results have indicated that the system performance can be significantly improved by properly selecting the number of non-zero elements of a K-dimensional SCMA codeword (N). Furthermore, it has been shown that our proposed SA method achieves a better system performance than conventional SA methods such as SA, SCMA-SA, and QL-SA, when the number of devices is larger than the number of available time-slots.

Acknowledgment

This work was supported in part by the FNR-funded project CORE 5G-Sky under Grant C19/IS/13713801, and ERC-funded project Agnostic under Grant 742648.

References

[1] S. K. Sharma and X. Wang, "Toward massive machine type communications in ultra-dense cellular IoT networks: Current issues and machine learning-assisted solutions," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 426–471, Firstquarter 2020.

[2] C. Bockelmann *et al.*, "Massive machine-type communications in 5G: physical and MAC-layer solutions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 59–65, Sep. 2016.

[3] G. Durisi, T. Koch, and P. Popovski, "Toward massive, ultra-reliable, and low-latency wireless communication with short packets," *Proc. IEEE*, vol. 104, no. 9, pp. 1711–1726, Sep. 2016.

[4] D. D. Tran, S. K. Sharma, S. Chatzinotas, I. Woungang, and B. Ottersten, "Short-packet communications for MIMO NOMA systems over Nakagami-m fading: BLER and minimum blocklength analysis," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3583–3598, Apr. 2021.

[5] S. Han *et al.*, "Energy-efficient short packet communications for uplink NOMA-based massive MTC networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 12 066–12 078, Dec. 2019.

[6] D. D. Tran, S. K. Sharma, and S. Chatzinotas, "BLER-based adaptive Q-learning for efficient random access in NOMA-based mMTC networks," in *IEEE Veh. Technol. Conf. (VTC-Spring)*, Helsinki, Finland, Apr. 2021, to appear.

[7] R. Li *et al.*, "Intelligent 5G: When cellular networks meet artificial intelligence," *IEEE Wireless Commun.*, vol. 24, no. 5, pp. 175–183, Oct. 2017.

[8] L. M. Bello, P. Mitchell, and D. Grace, "Application of Q-learning for RACH access to support M2M traffic over a cellular network," in *Eur. Wireless Conf.*, Barcelona, Spain, May 2014.

[9] A. H. Mohammed, A. S. Khwaja, A. Anpalagan, and I. Woungang, "Base station selection in M2M communication using Q-Learning algorithm in LTE-A networks," in *Int. Conf. Adv. Inf. Netw. Appl.*, Gwangiu, South Korea, Mar. 2015.

[10] S. K. Sharma and X. Wang, "Collaborative distributed Q-learning for RACH congestion minimization in cellular IoT networks," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 600–603, Apr. 2019.

[11] M. V. da Silva, R. D. Souza, H. Alves, and T. Abrão, "A NOMA-based Q-learning random access method for machine type communications," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1720–1724, Oct. 2020.

[12] Y. Yang, Y. Zhao, and D. Li, "SCMA uplink decoding with codebook collision," in *IEEE Veh. Technol. Conf. (VTC-Fall)*, Toronto, ON, Canada, Sep. 2017.

[13] J. Zeng *et al.*, "Achieving ultrareliable and low-latency communications in IoT by FD-SCMA," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 363–378, Jan. 2020.

[14] M. Cheng, Y. Wu, Y. Li, Y. Chen, and L. Zhang, "PHY abstraction and system evaluation for SCMA with UL grant-free transmission," in *IEEE Veh. Technol. Conf. (VTC-Spring)*, Sydney, NSW, Australia, Jun. 2017.