A Study of a Competitive Reinforcement Learning Approach for Joint Spatial Division and Multiplexing in Massive MIMO

Tanyaluk Deeka[†], **Boriboon Deeka**, and **Surajate On-rit**, Non-members

ABSTRACT

Massive Multiple-Input Multiple-Output (MIMO) is widely considered a pivotal communication technology for future generations of wireless networks. Massive MIMO uses a large number of antennas at the base station, which offers better effectiveness in spectral and energy use. However, a Frequency Division Duplex (FDD) system is challenging in reciprocity since it is difficult to estimate channels and requires feeding back channel state information. Joint Spatial Division and Multiplexing (JSDM) is a simplified FDD technique to provide massive MIMO gains. The main idea of JSDM is related to grouping users with approximately similar channel covariance. Many machine learning algorithms have been applied to conduct user grouping. In this paper, to improve the user grouping, we employ Reinforcement Guided Competitive Learning (RGCL) to the user grouping and then compare it with clustering techniques, including K-means, and sequential K-means to achieve the appropriate user grouping. The experimental results show that the RGCL technique represents better performance in computational time and system throughput than the other two above mentioned techniques, since RGCL can avoid being trapping in local minima.

Keywords: Long Term Evolution, Joint Spatial Division and Multiplexing, Reinforcement Guided Competitive Learning, Multiple-Input Multiple-Output

1. INTRODUCTION

The use of mobile broadband over the last decades has significantly increased as exemplified by the total mobile traffic in petabytes per month in the networks throughout the world [1]. It leads to a challenge in increasing network capacity. According to the statement of the Shannon-Hartley theorem [2], the capacity of a mobile system can be improved by increasing bandwidth or signal to interference plus the signal to noise ratio (SINR). However, available frequencies for mobile cellular systems are limited. In addition, only some portions of the radio frequency spectrum are deployed for all cellular phone networks. Therefore, increasing SINR is another possible way to increase capacity which can be achieved by reducing the cell size [3].

As demands for data consumption grow, the existing networks should be improved. First, WiMax was introduced as the next technology. However, it needs a big budget and is more time consuming to deploy than existing carriers utilizing CDMA or GSM. In addition, it was a network built from the ground up. Then, Long Term Evolution (LTE) was suggested to be the next option [4]. It is cheaper, uses current technologies, and can be implemented rather quickly. Moreover, it has been considered effective in enhancing peak data rates, decreasing latency and improving system capacity and coverage, especially for multiple antenna technologies. Furthermore, LTE uses OFDMA technology for the downlink, in which all available bandwidth is divided into small frequency channels called chunks [5]. Each chunk has to be assigned to the cells. The chunk assignment is a step of frequency planning, which is a process in the network implementation.

Currently, the frequency planning is done by a fixed scheme called the frequency reuse factor (FRF) concept [6]. The concept divides the whole bandwidth into chunks that are allocated to the base station statically. However, these fixed frequency assignment algorithms are inappropriate for next-generation mobile networks, because the behavior of network traffic is dynamic, spatial, and temporal. Hence, the dynamic spectrum assignment (DSA) concept has been discussed which can manage the spectrum bandwidth more efficiently than the fixed scheme [7]. In addition, Reinforcement Learning (RL) has been studied in the field of dynamic spectrum assignments, and it has been shown in many pieces of research to have high efficiency in exploiting the available spectrum in OFDMA based networks [8–11].

The multi-antenna technique is one of the technologies exploited in the LTE system to improve

Manuscript received on October 25, 2019 ; revised on June 16, 2020 ; accepted on October 23, 2020. This paper was recommended by Associate Editor Chanon Warisarn.

The authors are with the Department of Computer Network Engineering, Faculty of Industrial Technology, Ubon Ratchathani Rajabhat University, Ubon Ratchathani, Thailand.

[†]Corresponding author: tanyaluk.d@ubru.ac.th

©2021 Author(s). This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 4.0 License. To view a copy of this license visit: https://creativecommons.org/licenses/by-nc-nd/4.0/.

Digital Object Identifier 10.37936/ecti-eec.2021191.226832

system capacity. It is a basic technique for multiple-input multiple-output (MIMO) technology [12]. In addition, the MIMO has been incorporated in LTE Release 10, also known as LTE-Advanced (LTE-A), since it also provides large spatial multiplexing throughput gains. However, the broadband demands are surprisingly exponentially increasing but the throughput only increases almost linearly. The difference between the demand and the supply is continuously expanding.

To solve this issue, [13] proposed a concept of using a large number of antennas at the base station called massive MIMO [14, 15]. Normally, when many transmit antennas are installed at base station, this yields more degrees of freedom for serving a large number of users. In addition, it provides greater reliability and throughput. Although attributable to achieving channel state information (CSI) at the base station, it is a challenge to support many users in massive MIMO at the same time [16, 17]. Frequency division duplex (FDD) mode in massive MIMO is also facing a design challenge. Furthermore, JSDM which is a downlink scheme for multi-user MIMO (MU-MIMO), was introduced by [18] to reduce pilot based channel estimation and CSI. Because these two signals consume a lot of spectrum and power, JSDM needs to reduce the dimension of both downlink training and CSI with two main stages: pre-beamforming and MU-MIMO pre-coding.

The key concept of JSDM lies in dividing users into similar channel covariance groups. The grouping information is provided to the two sequential stages. Therefore, the user grouping process plays a crucial role in the JSDM approach. Recently, some new user grouping techniques for JSDM have been proposed. They also consider similarity and grouping algorithms. The similarity, which is based on the chordal distance, was applied to two different clustering algorithms, K-means and fixed quantization [18, 19]. In addition, a reinforcement method, namely the Reinforcement Guided Competitive Learning (RGCL) clustering algorithm, was found that can avoid trapping in local minimum [20], and the RGCL was proved that it is more flexible than the original competitive leaning procedures. Competitive learning is a version of K-means. The RGCL algorithm was tested on the simple well-know problem, the IRIS dataset [21]. This problem is a clustering problem trying to minimize the average distance between the instances and their clusters' centroids. The IRIS dataset has 150 data points with 4 attributes and 3 clusters. Two clusters are distinctly separate, but another cluster is not clearly different from the two clusters.

In the first part of this paper, we develop the user grouping process by modifying the input entities of the original RGCL algorithm. Secondly, we introduce a new user location pattern based on a Gaussian Mixture Model to represent a real-world user locations. We also intensively explore the number of groups used as a very important parameter in the grouping process. Finally, we compare the computational time and data rate produced by different user grouping algorithms.

The rest of the paper is structured as follows. Section 2 introduces the concept of JSDM. RGCL for User Grouping of the JSDM Approach is described in Section 3. Then the simulation model is represented in Section 4, and Section 5 relates our experimental results. Finally, conclusions are drawn and ideas for future work are given in Section 6.

2. JOINT SPATIAL DIVISION AND MULTIPLEXING

JSDM was employed to reduce CSI and pilot based channel estimation as a consequence of seeking smaller spectral and energy consumption [22]. JSDM can reduce the dimension of both downlink training and CSI at the transmitter, which has an effect on Massive MIMO at the base station.

Furthermore, JSDM has two main stages: pre-beamforming and MU-MIMO precoding. The first stage is the pre-beamforming matrix ($\mathbf{B}$), which is related to the inter-group interference, and the multi-user matrix ($\mathbf{P}$). It is produced to minimize the interference inside each group (the intra-group interference).

The system considered here is a downlink system with M antennas quipped at the base station and each user terminal (UT) using a single antenna (total number of UTs is K). The transmit antennas are placed along one axis to form a uniform linear array (ULA). As mentioned in [22], the received signals of all UTs $\mathbf{y}$ are defined as

$$\mathbf{y} = \mathbf{H^H V d} + \mathbf{z} \tag{1}$$

where $(\cdot)^H$ is the Hermitian of a matrix. $\mathbf{H}$ is the actual channel between the base station and the UTs. $\mathbf{V}$ is the precoding matrix, and $\mathbf{d}$ is the data vector. $\mathbf{z}$ is the zero-mean circularly symmetric complex Gaussian noise vector.

The second stage of JSDM is pre-coding. The pre-coding matrix $\mathbf{V}$ is the product of two pre-coding matrices; $\mathbf{V} = \mathbf{BP}$. $\mathbf{B}$ is a $M \times b$ matrix, which is designed based on the transmit spatial correlation. The same $\mathbf{B}$ is applied to the UTs with the similar or the same transmit correlation, which forms a group of UTs. $\mathbf{B}$ is designed to reduce the inter-group interference. After the pre-beamforming, we have matrix $\mathbf{B}$ and the effective transmit size b, which are determined by dominant eigenmodes of the average transmit correlation of user groups. The part which is designed to reduce the interference within each group is controlled by matrix $\mathbf{P}$.

Now we can define the effective channel after pre-beamforming, $\tilde{\mathbf{H}}$ as Eq. (2) [22].

$$\tilde{\mathbf{H}} = \mathbf{B}^{\mathbf{H}}\mathbf{H} \tag{2}$$

Given the user grouping process, matrix $\mathbf{B_g}$ can be found, where g is the group index. [23] proposed approximating the average signal to interference plus noise ratio (SINR) with the assumption that there are no intra-group co-scheduled users. The SINR can be defined as

$$\bar{\gamma}_{g_k} = \frac{\frac{P}{\sum_g b_g}|\mathbf{tr}(\mathbf{B_g^H R_{g_k} B_g})|}{1 + \frac{P}{\sum_g b_g}\sum_{g' \neq g}|\mathbf{tr}(\mathbf{B_{g'}^H R_{g_k} B_{g'}})|}. \tag{3}$$

When the average SINR is computed for a user, it is assumed that in an instantaneous time slot, only the selected UT is scheduled. Then the rate for the scheduled user g_k is given by

$$\eta_{g_k} = \log_2(1 + \gamma_{g_k}). \tag{4}$$

System throughput r_{ws} is achieved as $r_{ws} = \sum_{g=1}^{G}\sum_{k \in N_g}\eta_{g_k}$. N_g is the scheduled user set in the gth group.

According to [22], the key concept of the JSDM lies in dividing the users within the cell into groups with similar channel covariance and feed grouping information into the two sequential stages. Therefore, the user grouping process plays a crucial role in the JSDM approach. Machine Learning (ML) algorithms can perform classification or grouping tasks. A Reinforcement Learning (RL) algorithm like Reinforcement guided competitive learning (RGCL) proposed by [20] is suitable for the complex and noisy environment of an MIMO system using the JSDM method.

3. RGCL FOR USER GROUPING

Reinforcement Learning (RL), which is a class of learning problems in machine learning, has become one of the most active research areas in artificial intelligence, neural networks, and machine learning [24]. RL is generally used for solving complex problems when the controller is difficult to design. An agent which is controlled by the RL algorithm, must learn how to map situations to actions in order to maximize a numerical reward signal from its environment. The agent learns from its interaction within an environment, and not from a set of training data or supervised learning.

In the RL framework, the dynamic environment is observed by an agent to select action as a result of maximizing its reward [24]. The environment interacts continuously with the agent to represent new situations to the agent which are related to those actions. The connection between the current state and the selected action is roughly called policy. In some cases, the policy is a simple function, such as a lookup table that involves extensive computation and a search process. The problem can be divided into a learning system and its environment. The long-term interaction between the dynamic environment and the agent is mathematically modeled as a Markov Decision Process.

Many RL algorithms have been developed using the concept of immediate RL. The RGCL described in [25] uses the family of REINFORCE algorithms where the w_{ij} is updated by Eq. (5).

$$\Delta w_{ij} = \alpha(r - b_{ij})(y_i - p_i)\frac{\partial s_i}{\partial w_{ij}} \tag{5}$$

where $\alpha > 0$ is the learning rate. r is the reward, b_{ij} is reinforcement baseline. y_i is the output of the Bernoulli unit i. p_i is a probability computed as $p_i = f(\sum_{j=1}^{P} w_{ij}x_j)$. P is number of properties, and s_i is the distance between x and w_i.

RGCL is the method of RL deployed in this paper, since the RGCL is based on the clustering procedure of the competitive learning techniques. Let $X = [x_1, ..., x_N]$ be unlabeled input data. N is total number of inputs, $x_i = [x_{i1}, ..., x_{ip}]^{\mathrm{T}}$. p is the total number of data properties. $w_j = [w_{j1}, ...w_{jp}]$ is a cluster. $j = 1, .., L$, and L is the total number of clusters. $d(x_i, w_j)$ is the distance between user i and the center of cluster j. For RGCL, which learns clustering policy from the reward signal, the immediate reward [25] is generated by Eq. (6).

$$r_i = \begin{cases} 1 & \text{if } i = i^* \text{ and } y_i = 1 \\ -1 & \text{if } i = i^* \text{ and } y_i = 0 \\ 0 & \text{if } i \neq i^* \end{cases} \tag{6}$$

where y_i is the output of unit (or cluster) i. $y_i = 1$ means the unit is active, but $y_i = 0$ means the unit is inactive. y_i is the output of a Bernoulli random variable, so $y_i \in Y = \{0, 1\}$. [25] showed that the generating function is of the form:

$$g_i(y_i, p_i) = \begin{cases} 1 - p_i & \text{if } y_i = 0 \\ p_i & \text{if } y_i = 1 \end{cases} \tag{7}$$

where $p_i = f(s_i) = \dfrac{1}{1 + e^{-s_i}}$, s_i is $d^2(x, w_i)$.

Following the specification of the reward function and setting $b_{ij} = 0$ for every i and j, Eq. (5) simplifies to be Eq. (8) [25].

$$\Delta w_{ij} = \alpha r_i(y_i - p_i)\frac{\partial s_i}{\partial w_{ij}} \tag{8}$$

In the case where $s_i = d^2(x, w_i) = \sum\limits_{j=1}^{p} i^2$, Eq. (8) becomes the update equation of RGCL as shown in

Eq. (9) [25].

$$\Delta w_{ij} = \alpha r_i(y_i - p_i)(x_j - w_{ij}) \tag{9}$$

Then [25] applies the reward, r_i from Eqs. (6–9), so the update function is

$$\Delta w_{ij} = \begin{cases} \alpha|(y_i - p_i)|(x_j - w_{ij}) & \text{if } i = i^* \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

RGCL with normalized distance has an enhancement to improve the user grouping performance. Hence, RGCL and RGCL with normalized distance on user grouping in JSDM for massive MIMO were described in this section. In order to take advantage of the high potential of the JSDM technique, the users have to be divided into groups with similar covariance eigenspaces.

To summarize the concept of RGCL, it is a learning system trying to maximize the expected reward at the forthcoming step. It has an indirectly stochastic way to follow the clustering strategy. High rewards are obtained when the system follows the clustering strategy, but low values of r are received when the system is unsuccessful in the reward function.

In this section, we are interested in RGCL with the normalized distance which will be applied to the user grouping. This is because user grouping is a key task in the JSDM as a part of improving the performance of the massive MIMO system. Furthermore, RGCL has a stochastic property which avoids getting trapped in local minima. Hence, RGCL user grouping is exploited for finding proper groups of users.

4. SIMULATION MODEL

In the simulation, we focus on the user grouping when a Uniform Linear Array (ULA) is installed at the base station. In this scenario, the well-known one-ring model is considered and the base station is located at the center of the coverage-circle area.

The antennas are installed along one axis with 0.5λ spacing (λ stands for the wavelength of the operating frequency) to form a ULA. Then the N single-antenna users are served within one of three sectors.

The location of each UT is randomly chosen in (r, θ), a pair of the base station-to-UT distance, r and the angle of arrival (AoA), θ. r is in the range $[20(\mathrm{m}), 100(\mathrm{m})]$, and θ is in the range $[-60°, 60°]$. We assume that there is no significant scattering around the base station antennas. Then the (m, p)-th entry of channel covariance matrix for the user. Located at (r, θ), according to [22], is given by Eq. (11).

$$[R]_{m,p} = \frac{1}{2\Delta}\int_{-\Delta+\theta}^{\Delta+\theta} e^{-j2\pi D(m-p)\sin(\alpha)}\, d\alpha \tag{11}$$

The channel covariance matrix is comprised of the (m, p)-th entries defined by Eq. (11). This matrix will be used to calculate eigenvectors for Eq. (12) to group the users with similar channel covariance.

Table 1: The parameters of configuration.

Parameter	Value	Parameter	Value
D	0.5	G	8
$\theta_{\min}$	$-60°$	$\theta_{\max}$	$60°$
$\Delta_{\min}$	$5°$	$\Delta_{\max}$	$15°$
$s_{\min}$	20 (m)	$s_{\max}$	100 (m)

It is assumed that the base station knows the information of the user channel covariance, which can be tracked and learned, since it does not change over time. We fix the number of groups G to be 8. The parameters of configuration are given in Table 1.

We provide three different input datasets for observing the performance of the user grouping algorithms. Using the information about the positions of the users given in Table 1, the datasets have two different Gaussian Mixture Models of user distributions (GMM01 and GMM02). Another dataset is the uniform distribution (UNIF). The three types of user distribution for the experiment are shown in Fig. 1.

The user grouping requires using the eigenvector spaces ($\mathbf{U}$) as the input dataset. $\mathbf{U_k}$ denotes the matrix of the eigenvectors of $\mathbf{R_k}$. The similarity between two matrices of the eigenvectors will be calculated by the chordal distance as

$$d_c(\mathbf{U_k}, \mathbf{V_g}) = ||\mathbf{U_k}\mathbf{U_k^H} - \mathbf{V_g}\mathbf{V_g^H}||^{\mathbf{2}}_{\mathbf{F}} \tag{12}$$

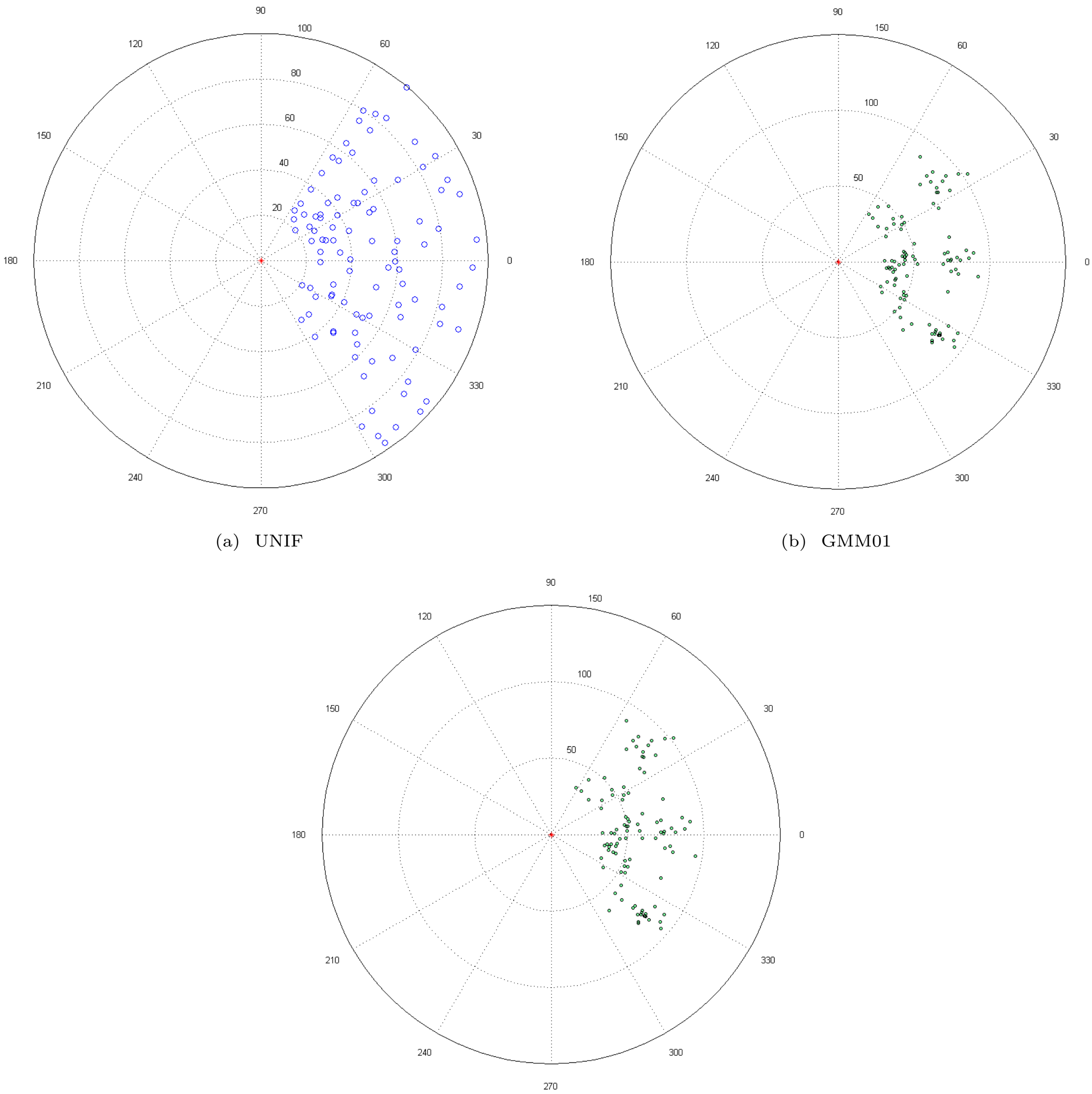
where $\mathbf{V_g}$ refers to as the matrix of the eigenvectors of the group center $\mathbf{R_g}$. This equation will be used in all of the user grouping algorithms to find the distance between the covariance eigenspaces of a user and its centroid.

4.1 Clustering Algorithms

In this experiment, three clustering algorithms are applied: K-means, Sequential K-means, and RGCL. K-means is chosen for the benchmark of the user grouping on JSDM because it has been done in previous work. The sequential K-means and the RGCL algorithms are similar in terms of the input instances processing and the centroid updating. They are considered sequential algorithms. These algorithms are compared to provide a good assessment.

4.2 Number of Groups for User Grouping

In Fig. 2, a summary of the data rates shows different numbers of users and groups. The summary of the data rate when the numbers of groups are 6 and 8 are highlighted (green and red). These groups were selected by [18]. The result shows that



(a) UNIF

(b) GMM01

(c) GMM02

Fig.1: *Uniform distribution and two different Gaussian Mixture Models of user distribution and for experiment testing.*

the number of groups for user grouping affects the summary of the data rate. It also shows that the optimum number of groups is not a fixed number (neither 6 nor 8) when the number of users in the system is changing. Based on JSDM, there are two main types of interference: Self-group Interference (SGI) and Inter-group Interference (IGI). SGI will increase if the number of users in a considered group increases. IGI will increase if the number of groups is increased. As a result, the number of groups should be flexible to adapt to a different number of users.

5. EXPERIMENTAL RESULTS

The simulation was run 20 times to evaluate the proposed strategy. We used K-means as the benchmark algorithm, and for a fair comparison, the sequential K-means algorithm, which uses a procedure similar to the RGCL algorithm, was deployed. The results from the simulation are given next.

5.1 Computational Time

Computational complexity of an algorithm is an important factor to be considered before applying the algorithm to a particular problem.The computational times are measured by capturing the CPU-time that the algorithms use to find the minimum Euclidean distance between the cluster centers and their members. The condition set to terminate the algorithm is $q > 4$, where q is an integer changed with the

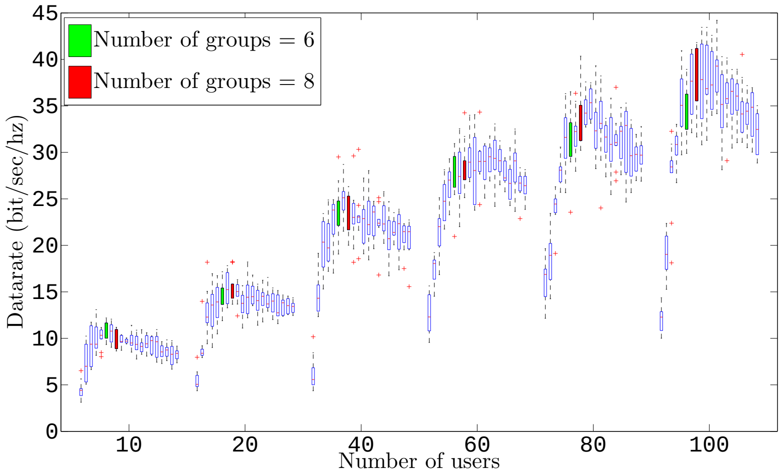


Fig.2: *Sum data rate of different number of users and groups.*

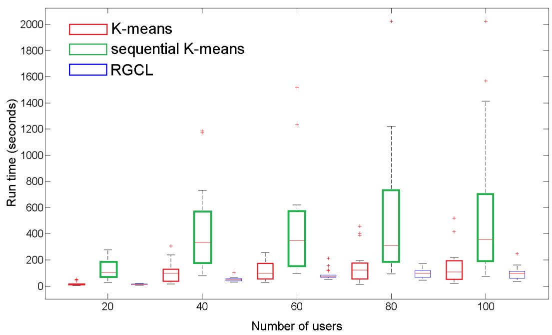


Fig.3: *Computational time for the UNIF distribution.*

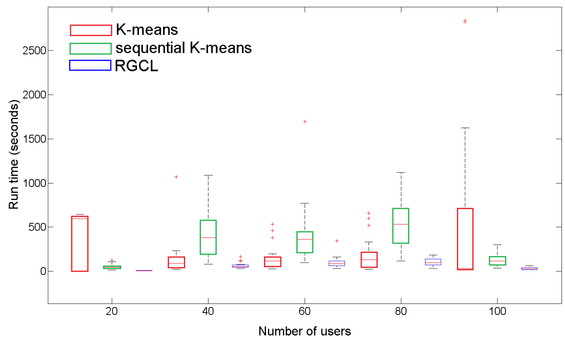


Fig.4: *Computational time for the GMM01 distribution.*

following conditions:

if $|J(n) - J(n-1)| < \epsilon J(n-1)$ **then**
    $q = q + 1$
else
    $q = 0$
end if

where $J(n)$ is the objective function and ϵ is a small constant value. In this experiment. ϵ is set to 0.001.

Figs. 3–5 present comparisons of the computational time among three algorithms. It can be clearly seen that the RGCL has the best performance because its run time is less than that of the K-means and sequential K-means methods.

5.2 Similarity Distance

The similarity distance is one of the key values showing the potential of JSDM approach since JSDM relies on the user grouping that partitions the users into groups with similar (approximately identical) eigenspaces. This parameter has a relationship on the ability of the JSDM technique to determine how well

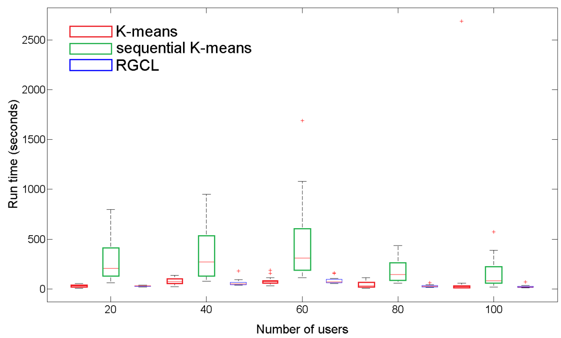


Fig.5: *Computational time for the GMM02 distribution.*

it can manage the interference. Figs. 6–8 show the performance of the RGCL user grouping. The RGCL user grouping has the lowest average distance, which means the RGCL user grouping divides the users into groups with more similar covariance eigenspaces than those of K-means and the sequential K-means user grouping. This is the result of the stochastic exploration mechanism inside the RGCL algorithm, which allows the algorithm to search for the best action (selecting a group for the user).

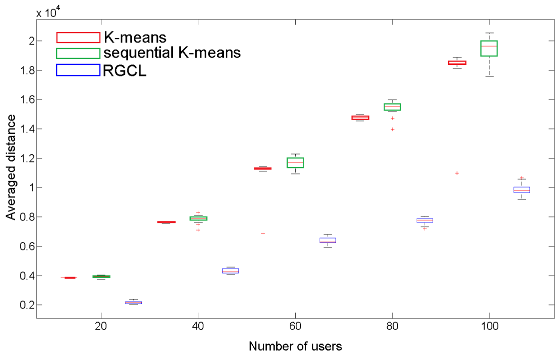


Fig.6: *Similarity for the UNIF distribution.*

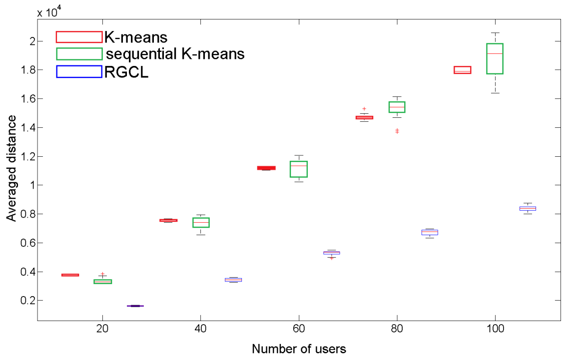


Fig.7: *Similarity for the GMM01 distribution.*

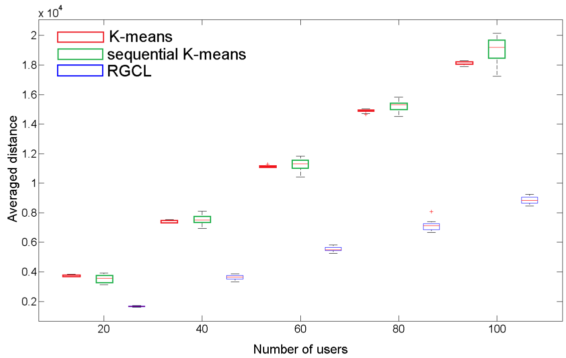


Fig.8: *Similarity for the GMM02 distribution.*

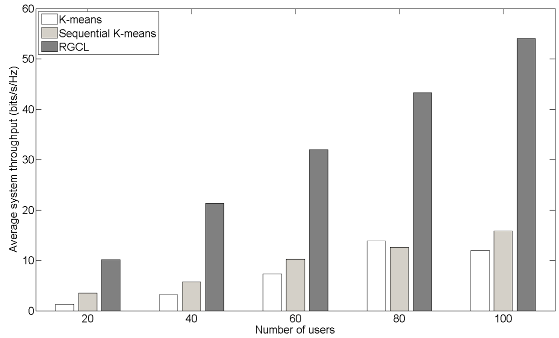


Fig.9: *System throughput for the uniform distribution users.*

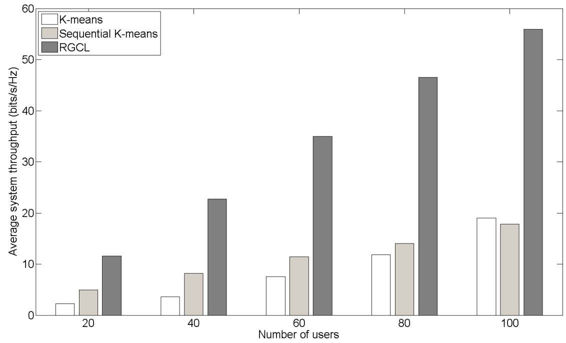


Fig.10: *System throughput for the GMM01 distribution users.*

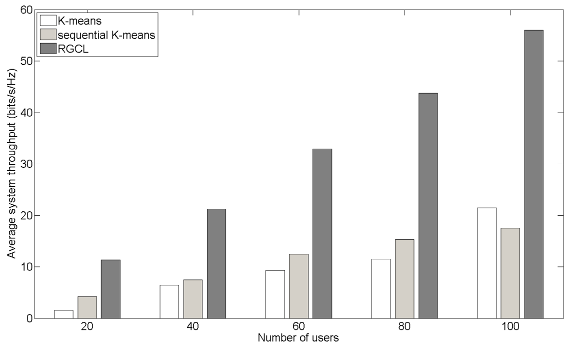


Fig.11: *System throughput for the GMM02 distribution users.*

5.3 System Throughput

To calculate the system throughput [see Section 2], we assume that all users are scheduled into the time-frequency resource block. However, the approximated SINR which is stored in matrix **P** is not involved. This means there is no intra-group interference for this experiment. The system throughput calculated from the equation contains computational error. Figs. 9–11 show the system throughput. The RGCL user grouping achieves the highest throughput. Although the calculated throughput contains some computational errors, it also indicates that RGCL user grouping would be a good technique.

In Fig. 12, comparison among K-means RGCL and RGCL's variance on sum data rate is shown. The original RGCL gives a lower sum data rate than the others. In the RGCL algorithm, the logistic function is used to calculate a probability for each pair of user and a centroid. The inputs (distances) of the function are supposed to be in the range that gives appropriate probabilities, allowing the algorithm to find an optimal solution. However, most of the distances between the matrices are close to 10 which gives probabilities close to 0. In this case, RGCL cannot explore enough and yields a bad solution when compared to one of the K-means. RGCL with normalized distance (RGCL norm) solves the problem which happens in the original RGCL. In RGCL norm, the maximum value of the input of the logistic function is designed to be 1 which gives a

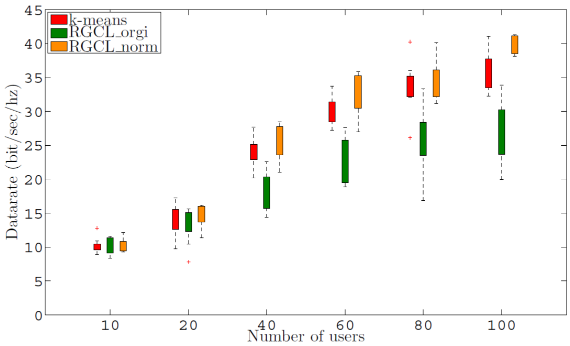


Fig.12: *Sum data rates, comparing between K-means and RGCL's variances in user grouping*

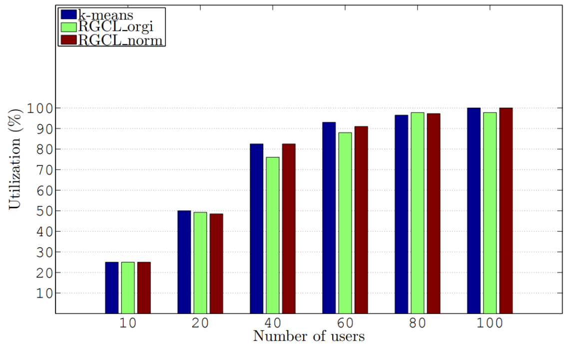


Fig.13: *Utilization of the system, number of beams over number of served users, at different number of users when applying different user grouping algorithms.*

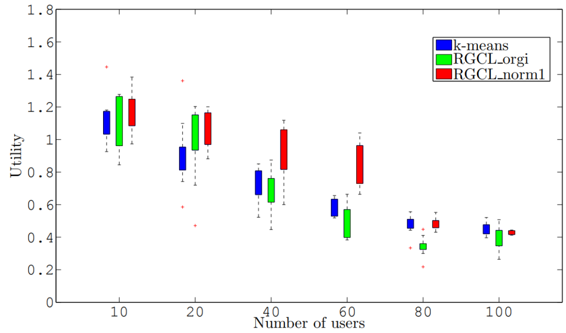


Fig.14: *Utility of the system at different number of users when applying different user grouping algorithms.*

probability of around 0.5. This allows the algorithm to give at least 0.5 probability to all centroids, so the algorithm has a higher chance of grouping users in different ways. As a result, it can find a good solution for user grouping. RGCL with normalized distance gives a good sum data rate that is slightly higher than the result from K-means. This leads to a suggestion for future work where the user grouping technique could be designed based on a dynamic number of groups to gain a higher sum data rate.

Comparison among K-means RGCL and RGCL's variance on sum data rate is shown in Fig. 12. It can be clearly seen that the original RGCL gives a lower sum data rate than the others. In the RGCL algorithm, a logistic function is used to calculate the probability for each pair of user and centroid. The inputs (distances) of the function are supposed to be in the range that gives appropriate probabilities allowing the algorithm to find an optimal solution. However, most of the distances between the matrices are close to 10, which gives probabilities close to 0. In this case, RGCL cannot explore enough and yields a bad solution when compared to one of K-means. RGCL with normalized distance (RGCL_norm) then solves the problem which happens in the original RGCL. In RGCL_norm, the maximum value of input of logistic function is designed to be 1 which gives a probability around 0.5. This allows the algorithm to give at least 0.5 probability to all centroids, so the algorithm has more chances to group users in different ways. As a result, it can find a good solution for user grouping. RGCL with normalized distance gives a good sum data rate and is slightly higher than the result from K-means.

Comparing Figs. 12 and 13, we can see that at 10 and 20 numbers of users, the system utilization (the number of beams over number of served users, at different number of users when applying different user grouping algorithms.) are not much different. They also give sum data rates without big differences. At small number of users in the service area, the probability that all of them are served is high, so sum data rates and utilization are not different. However, with 100 users, although the system utilization is close, the sum data rates are different. This is the result of user grouping techniques on the sum data rate. Different techniques are able to give different solutions.

These algorithms can group users to be served by nearly all ('all' for RGCL_norm) available beams, but they are using different methods and this causes different sum data rates.

Fig. 14 demonstrates utility gathered from the applied algorithms then measures achievable data rate over the number of group members. This result shows the importance of the number of group members in terms of proper data rates (all users are being served with reasonably different data rates). The difference in utility between groups can be decreased if the numbers of group members are closer to each other. This can lead to the situation that all users are being served with reasonably different data rates. With these data rates, it means that the users are transferring data with proper data rates.

Fig. 16 shows users' data rates in bits/s/Hz, and Mbps respectively. In the bottom figure, the waiting time is set to be 3 s, and the target data size to be downloaded is 2 Mbits.

It can be seen that there are some users being

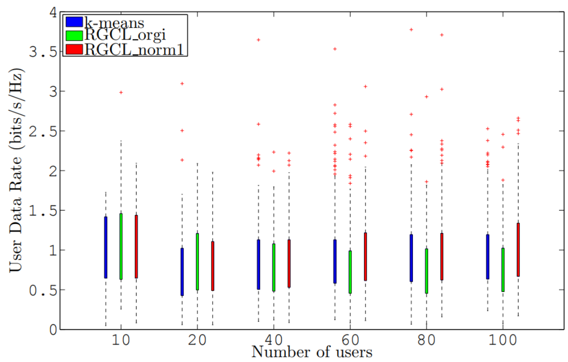


Fig. 15: *User data rates of different number of users (left to right: K-means, original RGCL and RGCL with normalized input data user grouping).*

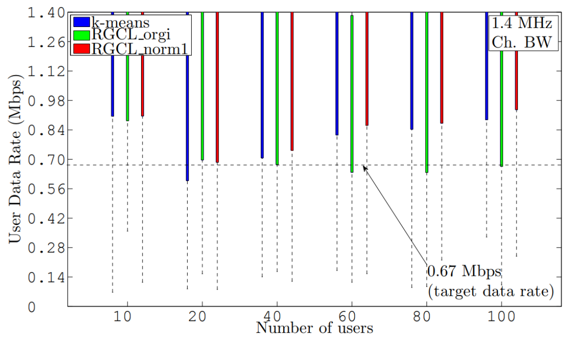


Fig. 16: *User grouping techniques (left to right: K-means, original RGCL and RGCL with normalized input data user grouping).*

served with data rates lower than the target data rate. The percentage of dissatisfaction can be calculated from the number of users transferring data with data rates lower than the target data rate.

5.4 Experimental Results and Discussions

RGCL and RGCL with normalized distance have been applied to the user grouping for the JSDM approach and compared with K-means and the sequential K-means user grouping. The aim of the strategy, applying the RGCL to the user grouping, is to find better groups of users, with better similarity, in order to support the JSDM approach to allow higher system throughput. The number of groups used in the user grouping process has been observed. The highest system throughput can be archived from a variable number of groups used in the user grouping process. System throughput is measured at different numbers of users in the cell with different numbers of groups used for RGCL user grouping. Fixing the number of groups for RGCL user grouping in JSDM gives lower system throughput when the number of users in the service area is changing. The results show that using RGCL with normalized distance for user grouping in JSDM for massive-MIMO enhance system throughput (an increase of 9.56 % points on average). Moreover, the system utility and utilization have been observed. RGCL with normalized distance user grouping gives a 15.08 % utility improvement, but no significant improvement for system utilization. Considering user dissatisfaction, using RGCL user grouping decrease dissatisfaction by 8.97 %.

6. CONCLUSION AND FUTURE WORK

Joint Spatial Division and Multiplexing (JSDM) approach to multiuser MIMO (MU-MIMO) downlink for Frequency Division Duplexing (FDD) system [10]. A year later, they proposed using a K-means clustering algorithm for the user grouping process in JSDM. The user grouping plays a significant role in JSDM. It strongly affects the massive MIMO system throughput.

Many user grouping techniques for JSDM have been proposed. K-means has been employed as a benchmark technique for our experiments on the JSDM's user grouping. A version of K-means called competitive learning (CL) was modified by [22]. The modified CL is Reinforcement Guided Competitive Learning (RGCL) clustering algorithm using the reinforcement to guide competitive learning. It does not get easily trapped in a local minimum because of the stochastic exploration mechanism. Thus, the idea of applying the RGCL clustering algorithm to the problem was formed.

In this paper, we aim to improve the user grouping by applying the RGCL algorithm to user grouping. We first observed the potential of the RGCL algorithm by performing clustering and compared it to K-means and sequential K-means. The RGCL shows good clustering potential that provides the highest system throughput. Furthermore, the RGCL user grouping performs the user grouping well, giving approximately 3.65 and 2.02 times higher system throughput than K-means and the sequential K-means user grouping respectively. Moreover, the RGCL gives approximately 50 % and 51 % lower similarity distance and consumes approximately 43 % and 83 % lower computational time than those of K-means and the sequential K-means user grouping respectively.

In future work, we plan to improve the RGCL user grouping by modifying the update function to improve both the computational time and the solution.

References

[1] C. V. Forecast, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013–2018," *Cisco Public Information*, vol. 9, Feb. 2014.

[2] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, pp. 379–428 and 623–656, 1948.

[3] C. Cox, *An Introduction to LTE: LTE, LTE-Advanced, SAE and 4G Mobile Communications.* Wiley, 2012.

[4] E. Dahlman, S. Parkvall, J. Skold, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*, 2nd ed. Oxford, UK: Academic Press, 2008.

[5] S. Sesia, I. Toufik, and M. Baker, Eds., *LTE - The UMTS Long Term Evolution*: From Theory to Practice. Wiley, 2009.

[6] Z. Wang and R. Stirling-Gallacher, "Frequency reuse scheme for cellular OFDM systems," *Electronics Letters*, vol. 38, no. 8, pp. 387–388, 2002.

[7] D. López-Pérez, A. Juttner, and J. Zhang, "Dynamic frequency planning versus frequency reuse schemes in OFDMA networks," in *IEEE 69th Vehicular Technology Conference (VTC Spring 2009)*, pp. 1–5, 2009.

[8] S. Singh and D. Bertsekas, "Reinforcement learning for dynamic channel allocation in cellular telephone systems," *Advances in Neural Information Processing Systems*, pp. 974–980, 1997.

[9] N. Lilith and K. Dogancay, "Distributed reduced-state SARSA algorithm for dynamic channel allocation in cellular networks featuring traffic mobility," in *2005 IEEE International Conference on Communications (ICC 2005)*, vol. 2, pp. 860–865, IEEE, 2005.

[10] F. Bernardo, R. Agustí, J. Pérez-Romero, and O. Sallent, "Dynamic spectrum assignment in multicell OFDMA networks enabling a secondary spectrum usage," *Wireless Communications and Mobile Computing*, vol. 9, no. 11, pp. 1502–1519, 2009.

[11] D. Kumar, N. Kanagaraj, and R. Srilakshmi, "Harmonized Q-learning for radio resource management in LTE based networks," in *2013 Proceedings of ITU Kaleidoscope: Building Sustainable Communities (K-2013)*, pp. 1–8, 2013.

[12] D.-S. Shiu, G. J. Foschini, M. J. Gans, and J. M. Kahn, "Fading Correlation and Its Effect on The Capacity of Multielement Antenna Systems," *IEEE Transactions on Communications*, vol. 48, no. 3, pp. 502–513, 2000.

[13] T. Marzetta, "Multi-Cellular Wireless with Base Stations Employing Unlimited Numbers of Antennas," in *Information Theory and Application (ITA) Workshop*, 2010.

[14] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO: How Many Antennas Do We Need?," in *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 545–550, 2011.

[15] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta, "Massive MIMO for Next Generation Wireless Systems," *IEEE Communications Magazine*, vol. 52, pp. 186–195, February 2014.

[16] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling Up MIMO: Opportunities and Challenges with Very Large Arrays," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 40–60, 2013.

[17] L. Lu, G. Li, A. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive mimo: Benefits and challenges," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 742–758, Oct 2014.

[18] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, "Joint Spatial Division and Multiplexing—The Large-Scale Array Regime," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6441–6463, 2013.

[19] J. Nam, A. Adhikary, J.-Y. Ahn, and G. Caire, "Joint Spatial Division and Multiplexing: Opportunistic Beamforming, User Grouping and Simplified Downlink Scheduling," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 876–890, Oct 2014.

[20] A. Likas, "A Reinforcement Learning Approach to Online Clustering," *Neural Computation*, vol. 11, no. 8, pp. 1915–1932, 1999.

[21] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[22] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, "Joint Spatial Division and Multiplexing," 2012, *arXiv:1209.1402*.

[23] Y. Xu, G. Yue, and S. Mao, "User Grouping for Massive MIMO in FDD Systems: New Design Methods and Analysis," *IEEE Access*, vol. 2, pp. 947–959, 2014.

[24] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction.* Cambridge, MA, USA: MIT Press, 1998.

[25] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3–4, pp. 229–256, 1992.

Tanyaluk Deeka received B.Eng. and M.Eng. degrees in Telecommunication Engineering from Suranaree University of Technology (SUT), Thailand, in 2004 and 2006, respectively. Her research interests include next-generation networking, routing protocols and machine learning.

Boriboon Deeka received B.Eng. and M.Eng. degrees in Telecommunication Engineering from Suranaree University of Technology (SUT), in 2001 and 2008, respectively. His current research focuses on wireless communications, MIMO, network modeling, cyber security, and machine learning.

Surajate On-rit received M.Eng. degree in Electronic and Telecommunication Engineering from King Mongkut's University of Technology Thonburi. He is an Assistant Professor and Dean of Industrial Technology faculty, UBRU. His research focuses on cyber security, cryptography, digital watermarking, and digital hardware design.