RESOURCE MANAGEMENT FOR OFDMA SYSTEMS USING ANT COLONY-BASED OPTIMIZATION TECHNIQUE

BY

MOHAMMAD SHOKRI KHALAF

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER NETWORKS

January, 2012

بسم الله الرحمن الرحيم

In the name of Allah, the Most Gracious and the Most Merciful

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by MOHAMMAD KHALAF under the direction of his thesis advisor and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE in COMPUTER NETWORKS.

Thesis Committee

Dr. Ashraf S. Hasan Mahmoud (Advisor)

9/1/2012

Dr. Samir Al-Ghadban (Co-Advisor)

9.1.2012

Dr. Shokri Selim (Member)

Sadiq Sait M 14 Jan 2012

Dr. Sadiq M. Sait (Member)

Dr. Tarek R. Sheltami (Member)

Dr. Basem Al-Madani
(Department Chairman)

Dr. Salam A. Zummo
(Dean of Graduate Studies)

عمادة الدراسات العليا
DEANSHIP OF GRADUATE STUDIES

18/2/12

Date

Dedicated

to

My Beloved Parents and Brothers

ACKNOWLEDGMENTS

All praise and thanks are due to Almighty Allah, Most Gracious and Most Merciful, for his immense beneficence and blessings. He bestowed upon me health, knowledge and patience to complete this work. May peace and blessings be upon prophet Muhammad (PBUH), his family and his companions.

Thereafter, acknowledgement is due to the support and facilities provided by the Computer Engineering Department of King Fahd University of Petroleum & Minerals for the completion of this work.

I acknowledge, with deep gratitude and appreciation, the inspiration, encouragement, valuable time and continuous guidance given to me by my thesis advisor, **Dr. Ashraf S. Hasan Mahmoud**. I am also grateful to my Committee members, **Dr. Samir Al-Ghadban**, **Dr. Shokri Selim**, **Dr. Sadiq M. Sait**, and **Dr. Tarek R. Sheltami** for their constructive guidance and support.

My heartfelt thanks are due to my parents and brothers for their prayers, guidance, and moral support throughout my academic life. My parents' advice, to strive for excellence has made all this work possible.

Last, but not least, thanks to all my colleagues and friends who encourage me a lot in my way to the achievement of this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THESIS ABSTRACT (ENGLISH)

NAME:	**MOHAMMAD KHALAF**
TITLE:	**RESOURCE MANAGEMENT FOR OFDMA SYSTEMS USING ANT COLONY-BASED OPTIMIZATION TECHNIQUE**
MAJOR FIELD:	**COMPUTER NETWORKS**
DATE OF DEGREE:	**SAFAR 1433 (H) (JAN 2012 G)**

The problem of sub-channel assignment and power allocation for a multiuser Orthogonal Frequency Division Multiplexing (OFDM) system while maximizing the total system throughput and satisfying the proportional rates constraint can be modeled as a mixed binary integer programming problem. The optimal solution for this problem is generally hard to find. In this thesis we develop and evaluate an Ant Colony-based optimization (ACO) algorithm to solve the problem and obtain solutions of acceptable qualities in terms of total system throughput and compliance with the proportional rates constraint, referred to by fairness. The developed algorithm performs joint sub-channel assignment and power allocation without making assumptions in regard to the initial power allocation. The algorithm performance is evaluated using simulations and is compared against several suboptimal deterministic algorithms from the related literature. Evaluation indicates that the ACO-based algorithm is able to obtain solutions that outperform the considered competing algorithms for most of the typical input parameters at the cost of prolonged execution time. In addition, the thesis also proposes a novel method to synthesize an optimization problem with a known answer and utilizes this method to test the quality of the obtained solution for the various considered algorithms against the optimal solution of the synthesized problem. The thesis includes numerical examples depicting the comparisons and highlighting the main features of the proposed algorithm.

MASTER OF SCIENCE DEGREE

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

Dhahran, Saudi Arabia

THESIS ABSTRACT (ARABIC)

ملخص الرسالة

الاسم: **محمد خلف**

عنوان الرسالة: **إداة الموارد لأنظمة التقسيم متعامدة التردد ومتعددة الوصول باستخدام تقنيات التحسين المستمدة من مستعمرات النمل**

التخصص: **شبكات الحاسوب**

تأريخ التخرج: **صفر 1433 هـ - (يناير 2012 م)**

إن مشكلة توزيع القنوات الفرعية والطاقة لنظام التقسيم المتعامد التردد والمتعدد الوصول لأكثر من مستخدم والتي تعظم إنتاجية النظام الكلي وتحقق القيود النموذجية من إجمالي الطاقة والإنصاف تعتبر مشكلة مختلطة ثنائية برمجية وعددية. عموما، من الصعب العثور على الحل الأمثل لهذه المشكلة. في هذه الرسالة نريد تطوير وتقييم خوارزمية تحسين مستعمرة النمل لحل هذه المشكلة والحصول على حلول مقبولة الصفات من حيث إنتاجية النظام الكلي والامتثال لمعدلات القيد النسبي المشار اليها بالانصاف. إن الخوارزمية المطورة تخصص كل من القنوات الفرعية والطاقة معا لكل مستخدم من دون وضع افتراضات فيما يتعلق بتوزيع الطاقة الاولية. لقد تم تقييم أداء الخوارزمية باستخدام المحاكاة والمقارنة مع عدة خوارزميات من المؤلفات ذات الصلة. إن تقييم خوارزمية تحسين مستعمرة النمل يشير الى أن الخوارزمية قادرة على الحصول على حلول أفضل من الخوارزميات المنافسة لمعظم معلمات الادخال النموذجية ولكن بحاجة لفترات طويلة من التنفيذ. بالإضافة الى ذلك، تقترح الرسالة طريقة مبتكرة لتوليف وتركيب الحل الافضل واستخدامة لاختبار نوعية الحلول المستخرجة من الخوارزميات المختلفة. وكذلك فان الرسالة تتضمن الأمثلة العددية التي تصور المقارنات وتسلط الضوء على الملامح الرئيسية للخوارزمية المقترحة.

شهادة ماجستير علوم

جامعة الملك فهد للبترول والمعادن

الظهران ، المملكة العربية السعودية

Chapter 1

INTRODUCTION

Huge research is carried out in the area of Orthogonal Frequency Division Multiplexing (OFDM) due to its ability to support high bit rate communication over wireless channels [1]. OFDM introduces a solution for intersymbol interference (ISI) [2]. Basically, OFDM is a special case of multicarrier modulation where a wideband channel is divided into multiple narrowband sub-channels. All sub-channels may be assigned to one user at a time as the case for single user OFDM systems, while in Orthogonal Frequency Division Multiple Access (OFDMA), sub-channels and time slots are shared amongst users. OFDMA is the key technology used in Worldwide Interoperability for Microwave Access (WiMAX) and advanced mobile systems such as Long-Term Evolution (LTE) [3].

In OFDM, sub-channels are orthogonal to each other which allow simultaneous transmission without interference from each other. The existence of inverse Fast Fourier Transform (FFT) is the reason behind the wide usage of OFDM systems [4]. OFDM is an enhancement over Frequency Division Multiplexing (FDM) which has the advantage of flat power profile over time in Time Division Multiple Access (TDMA).

The orthogonality of OFDM solves the problem of the frequency-selective fading by transforming a wideband frequency-selective channel into a set of parallel flat fading narrowband channels [4]. Figure 1.1 shows the OFDM signal and shows how the sub-channels (sub-carriers) are transmitted at the same time without interfering with each other.

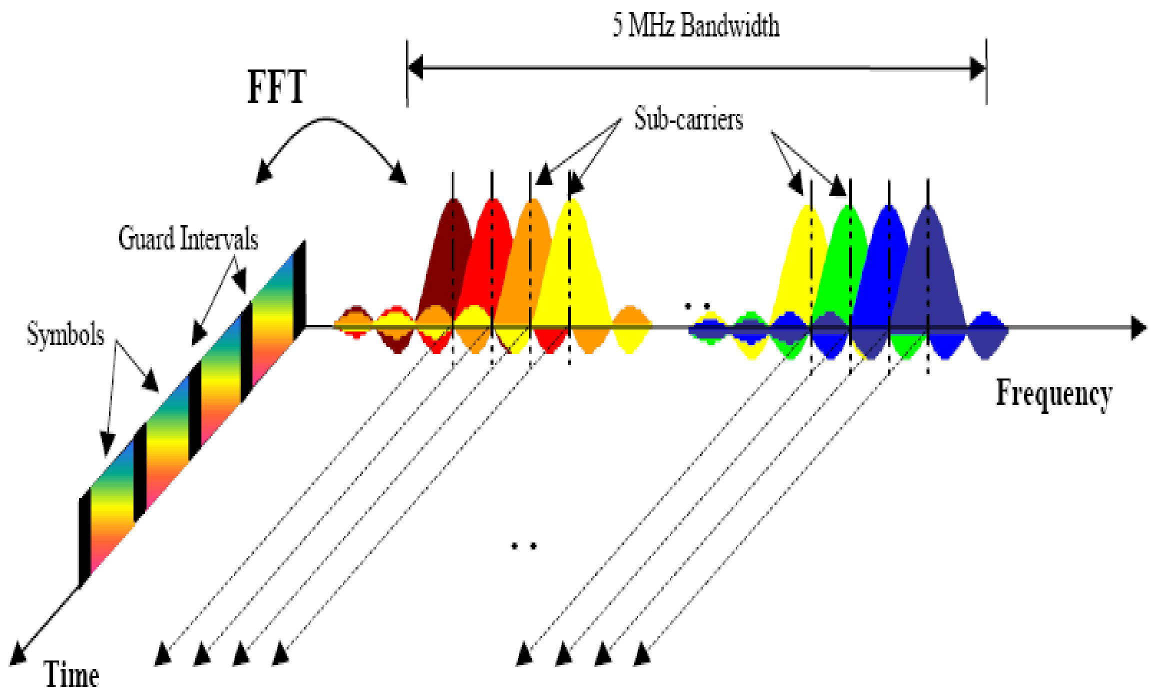

Figure 1.1: OFDM Signal [4]

At the same time, OFDMA allows freedom in scheduling. Therefore, recent research is directed to OFDMA resource allocation which is the process of assigning sub-channels, bits, and power to different OFDMA users. The sub-channels are the signals that are used to carry the bits, while power is the required energy to transmit the bits through the sub-channels [2]. OFDM resource allocation problem is divided into two schemes. First one minimizes total assigned power for wired systems with a constraint on user data rate; while second one maximizes total data rate for wireless systems with a constraint on total assigned power and fairness between users.

The problem of sub-channel and power allocation for a multiuser OFDMA system that maximizes the total system throughput while satisfying the constraints of total power and fairness can be modeled as a mixed binary integer programming problem [5]. The optimal solution is generally hard to find and considered as an NP-hard problem that is difficult to tackle. Therefore, solving this problem is the motivation behind this thesis work. This problem is described in details in the problem statement section. In addition, thesis objectives and thesis contribution are specified in this chapter.

1.1 PROBLEM STATMENT

Assume an OFDMA system with N sub-channels that is serving K users where the total system bandwidth B Hz is divided into the N narrowband flat fading sub-channels. The N sub-channels are distributed over the K users in order to maximize the overall network throughput. Therefore the objective function can be stated as follows

$$\max_{p_{k,n},\rho_{k,n}} \sum_{k=1}^{K}\sum_{n=1}^{N} \frac{\rho_{k,n} B}{N} \log_2\left(1 + p_{k,n} H_{k,n}\right) \tag{1.1}$$

where $H_{k,n}$ is the n^{th} sub-channel power gain relative to noise power as received by the k^{th} user, $n = 1, 2, \dots, N$, and $k = 1, 2, \dots, K$. $\rho_{k,n} \in \{0,1\}$ to indicate if the sub-channel n is allocated to the k^{th} user. If $\rho_{k,n} = 1$, then sub-channel n is allocated to the k^{th} user, while if $\rho_{k,n} = 0$, then sub-channel n is not allocated to the k^{th} user. Ω_k is the set of sub-channel indices where $\rho_{k,n} = 1$, i.e. $\Omega_k = \{n : \rho_{k,n} = 1\}$. The problem is subjected to three constraints.

First, the total power constraint is specified by

$$\sum_{k=1}^{K}\sum_{n\epsilon\Omega_k} p_{k,n} \leq P_{Total} \text{ and } p_{k,n} \geq 0 \tag{1.2}$$

where P_{Total} is the total power budget for the system, and $p_{k,n}$ is the power allocation for the $H_{k,n}$ sub-channel. Second, the sub-channel allocation Ω_k for different users is mutually exclusive. Finally, the proportional rates constraint is as follows

$$R_1/\gamma_1 = R_2/\gamma_2 = \cdots = R_K/\gamma_K \tag{1.3}$$

where R_k is the k^{th} user bit rate, after the allocation process is completed. The rate, R_k, is computed using

$$R_k = \sum_{n\epsilon\Omega_k} \frac{B}{N}\log_2(1 + p_{k,n}H_{k,n}) \tag{1.4}$$

and the constants γ_1, γ_2, …, and γ_K are the proportional rates constraint constants.

To assess the compliance of the obtained solution with the proportional rates constraint specified by equation (1.3), this thesis utilizes the Jian's fairness index formula [6] specified as follows

$$F = \left(\sum_{k=1}^{K}\left(\frac{R_k}{\gamma_k}\right)\right) \Big/ \left(K\sum_{k=1}^{K}\left(\frac{R_k}{\gamma_k}\right)^2\right) \tag{1.5}$$

After finding the required power allocations $p_{k,n}$'s and computing the user rates R_k's using (1.4), then one can evaluate the fairness achieved by substituting in (1.5). A value of one for the fairness index indicates that the proportional rates constraint is 100% satisfied while a value of zero indicates that the constraint is not met at all.

In this study, Ant Colony-based Optimization (ACO-based) algorithm is used to allocate sub-channels and power in a multiuser OFDMA system to obtain solutions of acceptable qualities in terms of total system throughput and compliance with the proportional rates constraint.

1.2 Thesis Objectives

The objectives of this thesis work are as follows:

- To implement an ACO-based algorithm to solve the optimization problem specified by (1.1) while meeting the relevant constraints with refinements.
- Analyze the performance of the ACO-based algorithm and study the effect of some of the algorithm input parameters on the quality of the obtained solution.
- Compare with other algorithms [1, 5, 7, 8] that solve the optimization problem specified by (1.1).

1.3 Thesis Contributions

The contributions of this thesis work are as follows:

- Designed and implemented an ACO-based algorithm for solving the resource allocation problem in multiuser OFDMA systems while meeting power and fairness constraints.
- Analyzed performance on the presented ACO-based algorithm.

- Analyzed performance of several competing methods [1, 5, 7, 8] and comparison in terms of overall throughput and meeting constraints.
- Publications: One conference paper.

Chapter 2

LITERATURE REVIEW

In this chapter, a survey is presented of the algorithms that solve the resource allocation problem of multiuser OFDMA systems. After that, a survey of ACO-based applications in the literature is presented.

2.1 OFDMA Resource Allocation Solutions

The optimal solution for the problem of sub-channel and power allocation for a multiuser OFDMA system that optimizes the overall throughput of the system while satisfying the typical constraints is generally hard to find. The problem of multiuser OFDMA system is considered as a non-convex optimization problem (NP-hard problem) that is difficult to solve. Therefore, many dynamic resource allocation schemes are developed for the OFDMA systems to find the solution of either minimizing the overall transmit power [2, 9-11] with constraints on the users' data rate, known as Margin Adaptive (MA) problem, or maximizing the sum of users' data rate [1, 5, 7, 8, 12-15] with a total power transmit constraint, known as Rate Adaptive (RA) problem [16].

In the literature, the algorithms that solve the resource allocation problem of OFDMA systems can be classified into deterministic algorithms [1, 2, 5, 7, 8, 17-19] and stochastic algorithms [9, 10, 12, 13, 20-26]. The deterministic algorithms are the algorithms that always have the same solution for the same input parameters, while the stochastic algorithms are the algorithms that have different solutions for the same input parameters. In addition, the algorithms are classified based on the resource allocation. Some algorithms allocate sub-channels only and make assumptions in regard to the power allocation as in [1] and in [7]. Other algorithms allocate power only for an assumed or a given sub-channels allocation as in [5] and in [8]. The other algorithms allocate both power and sub-channels as [2, 9, 11, 15, 22-25].

Lagrange relaxation [2] is the first deterministic algorithm that is used to solve this problem. Lagrange relaxation can be defined as a mathematical method for simplifying hard optimization problems by relaxing them. Relaxing the problem is done by removing the constraints which make the problem hard to solve and adding them to the objective functions. In [2], the author tries to minimize the total power consumption with limitation on data rate for users who require different data rates. The author relaxes the requirement of the number of bits required for the assignment of sub-channels to users, by allowing it to be a real number within the interval of zero and the maximum number of bits that can be transmitted by each sub-channel. Moreover, linear programming is used in [17], [18], and [19] to assign resources. Linear programming solves the resource allocation problem through linearizing the function of rate in term of power.

In [1], the authors study the problem of dynamic multiuser sub-channel allocation in the downlink of OFDM systems. They develop a multiuser convex optimization problem to find the sub-optimal allocation. They propose a suboptimal adaptive sub-channel allocation algorithm to solve the problem. The authors' algorithm assumes equal power for all sub-channels and assigns the sub-channels one by one to the users with the least capacity. They maximize the smallest capacity of all users.

The study in [7] assigns sub-channels to the users with the least normalized capacity over proportional rate constants. The allocation of power in [7] is different from the study in [1]. First, the study in [7] assumes equal power for all sub-channels as in [1]. Then, water-filling algorithm is used for each user to allocate the power. The allocated power for each user is directly proportional to the number of sub-channels allocated to that user.

In [8], the author focuses on the power allocation problem. The author does the allocation of sub-channels and power separately. Firstly, the allocation of sub-channels is done based on Rhee algorithm [1], assuming equal power distribution across all sub-channels. Secondly, the distribution of power over users is done based on solving a set of nonlinear equations. Then, water-filling is used for each user to distribute user power across user sub-channels to maximize the capacity.

In [5], the authors propose a systematic mathematical algorithm that computes the optimal power allocation for a given sub-channel allocation scheme. Their solution satisfy the proportional rate constraint in the strictest sense depending on the drop of weak sub-channels and therefore can provide absolute guarantees for the expected quality of service.

In addition to the deterministic algorithms, bio-inspired algorithms [9-13, 20-27], which are a subset of stochastic algorithms, are also used to solve the problem. Genetic algorithm (GA) is one type of the bio-inspired algorithms which is used in [13, 20, 23, 26, 28] in order to find solutions for the OFDMA resource allocation problem. Moreover, the study in [24] uses the GA to solve the joint sub-channel and power allocation problem for Multiple Input Multiple Output OFDMA (MIMO-OFDMA) system. The authors in [24] try to maximize the sum of user data rates subject to constraints on total power, bit error rate, and proportionality among user data rates. Particle Swarm Optimization (PSO) algorithm and Ant Colony-based Optimization (ACO-based) algorithm are newly introduced to solve the problem. The PSO is a population-based search algorithm which is employed in [9, 12, 14, 15, 25]. At the same time, the ACO-based method is employed in [10, 11, 21, 22, 27, 29, 30].

In [9], the authors apply the GA and the PSO for adaptive sub-channel and bit allocations to minimize the overall transmit power of a multiuser OFDM system. The GA is modified by using a fractional generation gap that helps to converge quickly by taking the good genes for the next generation. The algorithms guarantee at least one sub-channel to be assigned to each user.

In [15], the authors try to solve the problem of bit allocation to maximize data rate under the power and bit error rate constraints using the PSO. The authors propose Cloud Particle Swarm Optimization (CPSO) algorithm. The CPSO is described as a novel evolutionary model with property of cloud model to improve the diversity of population and overcome the shortcoming of running into local minimum in the PSO. Also in [25], the PSO is employed to allocate sub-channels for users followed by power allocation using water-filling algorithm. The study in [25] solves the joint sub-channel and power allocation problem. The study maximizes the sum of user's data rates subjects to constraints on total power, bit error rate, and proportionality among user's data rates.

In [11], the authors use the ACO-based algorithm to solve the bit and sub-channel resource allocation problem of single-cell OFDMA systems. The authors' goal is to allocate the sub-channels that minimize the total power consumption and guarantee the required minimum bit rate for all users. Results in [11] have shown that the ACO-based algorithm outperforms both the genetic algorithm and the modified genetic algorithm that uses water-filling algorithm. Moreover, the authors in [22] employ the ACO-based algorithm to allocate resources in an OFDMA mesh network to obtain an acceptable solution that maximizes throughput under power and Quality of Service (QoS) constraints. They propose Ant-Colony-based algorithm that is capable of satisfying different requirements and constraints.

In [10] and [21], the ACO-based algorithm is used to support the authors' goal of finding one suboptimal solution for OFDMA allocation in a short period of time. In [21], the authors show how the ACO-based algorithm can be used to dynamically allocates sub-channels that maximize total data rate under power constraints without considering the proportionality constraint. In [10], the authors' target is to find the solution with the minimum power consumption. The study does not consider any users' rate constraints or users' proportionality constraints. Therefore, it finds the solution in a short period of time. The results in [21] show that the number of users and sub-channels play a significant factor in the time required for finding solutions. Also, the study in [21] shows that the execution time increases when the number of ants increases for small fixed number of sub-channels and users. For example, the authors use nine sub-channels and three users.

Most importantly, none of the previous work in the literature solves the problem of maximizing the total system capacity while satisfying users proportional rate constraint using the ACO-based algorithm.

2.2 ACO Applications

In addition to the previous resources, the ACO-based method is used for many other applications different than the OFDMA [31-55]. The ACO-based method solves the travelling salesman problem (TSP) in [31-35]. Furthermore, the ACO-based method is applied to the static routing problems as the study in [36] and dynamic routing problems as the studies in [37, 38]. Moreover, the ACO-based approach solves the problem of virtual-wavelength-path and wavelength allocation in [39]. Additionally, the ACO-based algorithm can be applied to continuous domain problems beside to discrete domain problems as in [40]. The discrete domain problem is the problem that is defined for a set of integer numbers within a set of real numbers, while the continuous domain problem is the problem that is defined for all the real numbers in the interval of the set [56].

The studies in [41-45] describe the stages of development in the ant systems and describe the ACO-based algorithm in details. They mention and describe a lot of applications that have been solved by the ACO-based algorithm such as travelling salesman problem, routing in communications networks, and other meta-heuristic applications. Also in [46], the author combines the ACO-based algorithm with tree search algorithm, called beam search, for the application to open shop scheduling (OSS). The results in [46] show that the hybrid algorithm (beam-ACO) outperforms the original ACO-based algorithm. Moreover, data mining classification problem is another application that has been solved by the ACO-based algorithm in [47, 48]. Data mining is defined as the study of data patterns to apply discovery algorithms and make it automated process.

Additionally, the ACO-based method is used to solve the sequencing problems in [49, 50]. In [49], the authors study a car sequencing problem where the cars need to be arranged in assembly line to add features to them. The authors express greedy heuristics, local search, and the ACO-based algorithms in their work and apply the algorithms on the problem. They compare between the results of them and show that the ACO-based algorithm has the best outcome. Another sequencing problem is a just-in-time (JIT) sequencing problem [50]. The JIT sequencing problem is required for the production systems in the modern manufacturing firms. The author in [50] compares the ACO-based algorithm with simulated annealing, tabu search, genetic algorithms, and neural networks algorithms. The results confirm that the ACO-based algorithm is better in terms of performance and CPU requirements.

A bin packing problem (BPP) and a cut stock problem (CSP) are solved by a combination of the ACO-based algorithm and simple local search algorithm in [51]. The author adds the local search algorithm to improve the performance of the ACO-based algorithm. The BPP problem and the CSP problem are classified as NP-hard problems. In the BPP, the problem is to combine items into bins of a certain capacity in order to minimize the total number of bins. In the CSP, the problem is to cut items from stocks of a certain length to minimize the number of stocks. The results in [51] show that the hybrid ACO algorithm outperforms the evolutionary programming approach (EP), the hybrid grouping genetic algorithm (HGGA), and the Martello and Toth's reduction procedure (MTP).

In [52], the authors solve the resource-constrained project scheduling problem (RCPSP) using the ACO-based algorithm. The authors define the RCPSP as an "**optimization problem to schedule the activities of a project such that the makespan of the schedule is minimized while given precedence constraints between the activities are satisfied and resource requirements of the scheduled activities per time unit do not exceed given capacity constraints for the different types of resources**" [52]. The RCPSP paper [52] shows that the ACO-based algorithm has the best results on average against tabu search, simulated annealing, and genetic algorithms.

In [53], the ACO-based algorithm solves another scheduling problem called a single machine total weighted tardiness problem (SMTWTP). The SMTWTP is an NP-hard problem where a single machine processes number of jobs sequentially. The authors in [53] use the ACO-based algorithm to find the sequence of jobs that minimizes the sum of weighted tardiness. Also in [54], the ACO-based algorithm provides a solution to a scheduling problem in industrial in an aluminum casting center. The authors use the ACO-based algorithm to get an efficient representation of a continuous horizontal casting operation taking into account a number of objects that are important to scheduler [54].

In [55], the authors use the ACO-based algorithm to solve the vehicle routing problem (VRP). They modify the ACO-based algorithm to find multiple routes of the VRP. The VRP is described as a number of vehicles need to find the minimum cost of combined routes from source location to multiple destination locations. The results in

[55] show that the ACO-based algorithm work properly for limited list sizes but not for large ones.

Chapter 3

METHODOLOGY

In this chapter, the ACO-based algorithm is discussed. Then a plan for implementing the ACO-based algorithm is introduced. Also the basic implementation of ACO-based algorithm with OFDMA systems is presented. Then, different enhancements are added to the original implementation and described in the sections of this chapter.

3.1 STUDY OF ANT COLONY BASED OPTIMIZATION (ACO-based) ALGORITHM

Ant Colony-based Optimization (ACO-based) algorithm is a stochastic algorithm which is used to find an optimal solution for combinatorial optimization problems [57]. The experiments run by Goss et *al*. in [58] inspire ACO-based algorithm using a colony of real ants [57]. The idea behind the ACO-based algorithm comes from the way ants look for their food. The ants' goal is to find the shortest path to the food source. The ants mark their path by depositing a liquid, known as pheromone, on the ground. The pheromone concentration is affected by the number of ants as well as the time. As more ants pass from the same path, the pheromone concentration will be higher on that path. Simultaneously, the time has an inverse effect on the pheromone which is faded with time. Paths with higher pheromone will have higher probability to be selected by the ants in their next tours.

Figure 3.1 shows the basic flowchart of the ants' method. First, ants go individually to look for their food. The ants who find the food source will come back to their nest to tell the others. Then, the new ants will start to follow the effect of the pheromone on the ground to reach the food source. The new ants will deposit the pheromone on the ground through the back way to the nest. Therefore, the shortest path will have high pheromone after long time. Some ants may get lost and go astray through other paths. But the effect of pheromone on these paths will quickly degrade as these paths are longer than the selected one and the number of ants passing through them is too small. The ants' algorithm can be summarized as follows

- $\forall\ t = 1$ to number of cycles needed to move food,
  - $\forall\ a = 1$ to number of ants,
    - While food source not found,
      - Select path with probability (P) based on the pheromone and the ants visibility.

The implementation of ACO-based algorithm for any application or problem should have the main three loops. The first two loops are important to reach the final solution. But the third loop, while loop, is important to achieve the required conditions for the application or the problem. Probability (P) is calculated based on the application or the problem that need to be solved and it has general formula. The formula will be discussed in section 3.2 for the resource allocation problem of the OFDMA systems. The pheromone concentration and the visibility of the ants will affect the values of the probability.

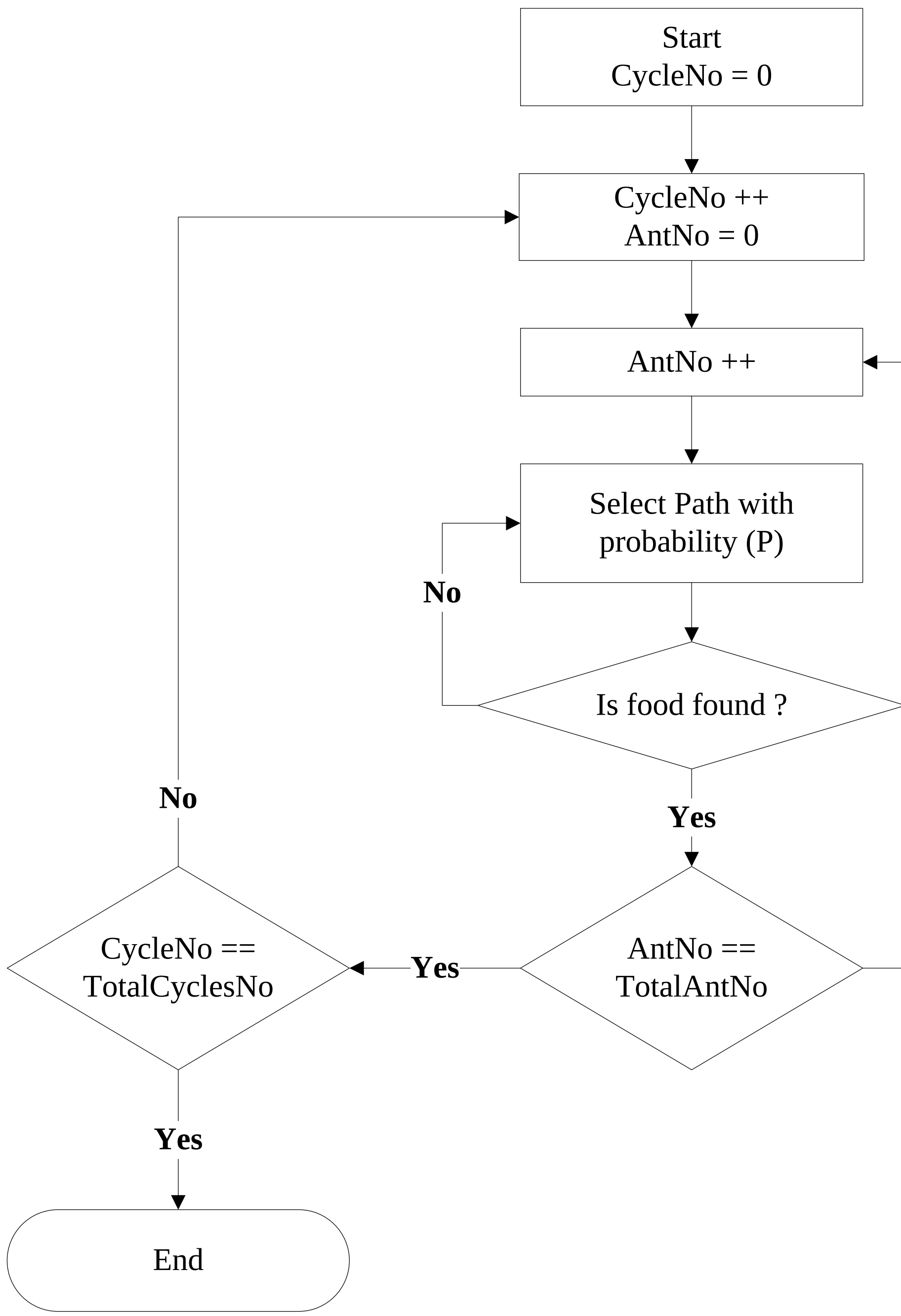

Figure 3.1: Basic Ants Algorithm

In this thesis work, the ACO-based algorithm for the resource allocation problem of a multiuser OFDMA system which maximizes the total system capacity and satisfies the proportional rate constraint is designed and implemented in three stages. Each stage adds enhancements to the previous one. The solution of stage 3 is the one that deemed to fulfill the solution requirements for the resource allocation problem in this thesis, in terms of satisfying the respective constraints. Figure 3.2 shows the stages in order with the main outline of each one. The stages are described individually in section 3.2, section 3.3, and section 3.4.

Stage 1: Algorithm Version 1
- Basic implementation of ACO-based algorithm for OFDMA systems.
- Assignments of sub-channels and users are random and not driven by any method.
- Assumes equal power for all sub-channels.
- Presented in section 3.2.

Stage 2: Algorithm Version 2
- Round robin assignments of sub-channels to users.
- Water-filling algorithm is used for power allocation.
- Presented in section 3.3.

Stage 3: Algorithm Version 3
- Final implementation of ACO-based algorithm for OFDMA systems.
- Sub-channels are assigned to the users with the least normalized data rate over proportional rate constants.
- Presented in section 3.4.

Figure 3.2: ACO-based Algorithm Main Stages

3.2 STAGE 1: BASIC IMPLEMENTATION OF ACO-BASED ALGORITHM FOR OFDMA PROBLEM

In this study, the ACO-based algorithm is implemented to find the maximum total system capacity for an OFDMA system. The ACO-based algorithm allocates sub-channels in a way that each sub-channel can be assigned to only one user to satisfy the problem constraints, since sub-channel allocation sets are assumed to be mutually exclusive. Each sub-channel is allocated a fixed power equal to P_{Total}/N.

The assignment of sub-channel n to the k^{th} user depends on the probability of assigning them to each other. The assignment probability is calculated based on the density of using the assignment and the desirability of the assignment itself. The density of selecting the assignment of sub-channel n to user k is called trail intensity and is denoted by $T_n^k(t)$. The desirability of selecting the assignment of sub-channel n to user k is based on the rate of this assignment, $R(k,n)$, where $R(k,n) = \frac{1}{N} * \log_2(1 + \frac{P_{Total}}{N} H(k,n))$. Assignments with higher rates are more desirable to be selected. Therefore, the probability of assigning sub-channel n to user k, P_n^k, is calculated as follows

$$P_n^k \;=\; \frac{(T_n^k(t))^{\alpha}(R(k,n))^{\beta}}{\sum_{k=1}^{K}\sum_{n\in A}\left(T_n^k(t)\right)^{\alpha}(R(k,n))^{\beta}} \tag{3.1}$$

A is the set of available sub-channels. α and β are constants to control the influence of the trail intensity and the desirability respectively, where $\alpha \geq 0$ and $\beta \geq 1$ [58]. The trail intensity, $T_n^k(t)$, is calculated as follows

$$T_n^k(t) = \rho T_n^k(t-1) + \Delta T_n^k \tag{3.2}$$

where $T_n^k(t)$ is the new value of the trail intensity and $T_n^k(t-1)$ is the old value of the trail intensity. ρ is the evaporation coefficient that is used to reduce the values of the trail intensity over time. This will avoid any possibilities of getting stuck in local optima. ΔT_n^k is the amount of increase in the trail intensity of the assignment of sub-channel n to user k. ΔT_n^k is updated as follows

$$\Delta T_n^k = \sum_{\forall a \in antsNumber} \Delta T_{an}^k \tag{3.3}$$

where ΔT_{an}^k is the amount of trail intensity updated for the ants that succeed in the assignment. ΔT_{an}^k is calculated as follows

$$\Delta T_{an}^k = \begin{cases} (Q * R_a) & a^{\text{th}}\text{ant select user } k \text{ to sub} - \text{channel } n \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

where Q is a constant. R_a is the total assigned rate by the a^{th} ant which assigns sub-channel n to user k. R_a is calculated by finding the total system capacity of the a^{th} ant where $R_a = \sum_{k=1}^{K} \sum_{n \in \Omega_k} \frac{1}{N} * \log_2(1 + \frac{P_{Total}}{N} H(k,n))$.

In this implementation, the allocations are chosen based on the probability of the assignment of sub-channel n to the k^{th} user. The selection of sub-channel n and user k will be random and not driven by any method. Therefore, the assignments will be chosen from $K * N'$ matrix where $1 \leq N' \leq N$, and N' is the number of available sub-channels. The pseudo code for the basic implementation of ACO-based algorithm is presented in

Figure 3.3. The probability (P_j^i) and the trail intensity ($T_n^k(t)$) will be calculated using equations (3.1) and (3.2), respectively. For more clarification of how the basic implementation works, Figure 3.4 shows the flow chart of the implemented algorithm. Most important, version 1 of the algorithm is concerned with finding the solution of the resource allocation problem for the OFDMA systems that maximizes the total system throughput without satisfying any proportional rate constraint.

The $fairnessLevel$ parameter used in Figure 3.3 and Figure 3.4 is used to specify the lowest level of fairness to be achieved amongst users. The use of this variable will take effect mainly in version 1 of the algorithm because the assignments of sub-channels to users in this version are random and not driven by any method. Therefore, the fairness between users will not be affected. The fairness will be controlled at the end of the assignments; if the fairness condition satisfied, then the assignments are taken. If not, then they are rejected. The $fairnessLevel$ variable is mainly used to show the relation between the total system capacity and the fairness between users. Therefore, this variable will not affect the other versions of the algorithm. The $maxRate$ variable in both figures, i.e. Figure 3.3 and Figure 3.4, is used to save the best solution which is found by the algorithm. Ω_{Final} is used to save the sub-channels allocations based on the $maxRate$ variable.

ACO-based Algorithm Version 1:

0) Initialization
   a. Initialize N to number of sub-channels, K to number of users, P_{Total} to the system total power, $cyclesNumber$, $antsNumber$, and $fairnessLevel$
   b. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{ T_n^k(t) = 1, \Omega_{Final}(k,n) = 0,$ Generate $H(k,n)\}$ where $T_n^k(t)$ and $\Omega_{Final}(k,n)$ are the trail intensity and the final sub-channel allocation for each sub-channel n to the user k, respectively.
   c. For $k = 1$ to $K \rightarrow \{maxRate(k) = 0\}$
   d. $S = \{1, 2, \ldots, K\}$
1) For $t = 1$ to $cyclesNumber$
   a. For $a = 1$ to $antsNumber$
      i. Initialize $A = \{1,2, \ldots, N\}$
      ii. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{\Omega(k,n) = 0\}$ where $\Omega(k,n)$ is the sub-channel allocation for each assignment of sub-channel n to user k of the a^{th} ant.
      iii. While $A \neq \emptyset$,
         1. Initialize $R(k,n) = 0, \forall n \in A$ and $\forall k \in S$
         2. Calculate $R(k,n)$ $\forall n \in A$ and $\forall k \in S$ where $R(k,n) = \frac{1}{N} * \log_2(1 + \frac{P_{Total}}{N} H(k,n))$
         3. Assign sub-channel j to user i where j and i are randomly generated using P_j^i specified by (3.1), where $i \in S$ and $j \in A$.
         4. $A := A - \{j\}$
         5. $\Omega(i,j) = 1$
      iv. Calculate $R(k)$ $\forall k \in S$ where $R(k) = \sum_{n \in \Omega_k} \frac{1}{N} * \log_2(1 + \frac{P_{Total}}{N} H(k,n))$
      v. Calculate fairness (F) such that $F = (\sum_{k=1}^{K} \frac{R_k}{\gamma_k})^2 / (\left(\sum_{k=1}^{K} (\frac{R_k}{\gamma_k})^2\right) * K)$
      vi. If ($F \geq fairnessLevel$)
         1. If ($\sum_{\forall k \in S} R(k) > \sum_{\forall k \in S} maxRate(k)$)
            a. $maxRate(k) = R(k), \forall k \in S$
            b. $\Omega_{Final}(k,n) = \Omega(k,n) \forall k \in S$ and $n = \{1,2, \ldots, N\}$
   b. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{$update $T_n^k(t)$ based on (3.2)$\}$

Figure 3.3: ACO-based Algorithm Version 1

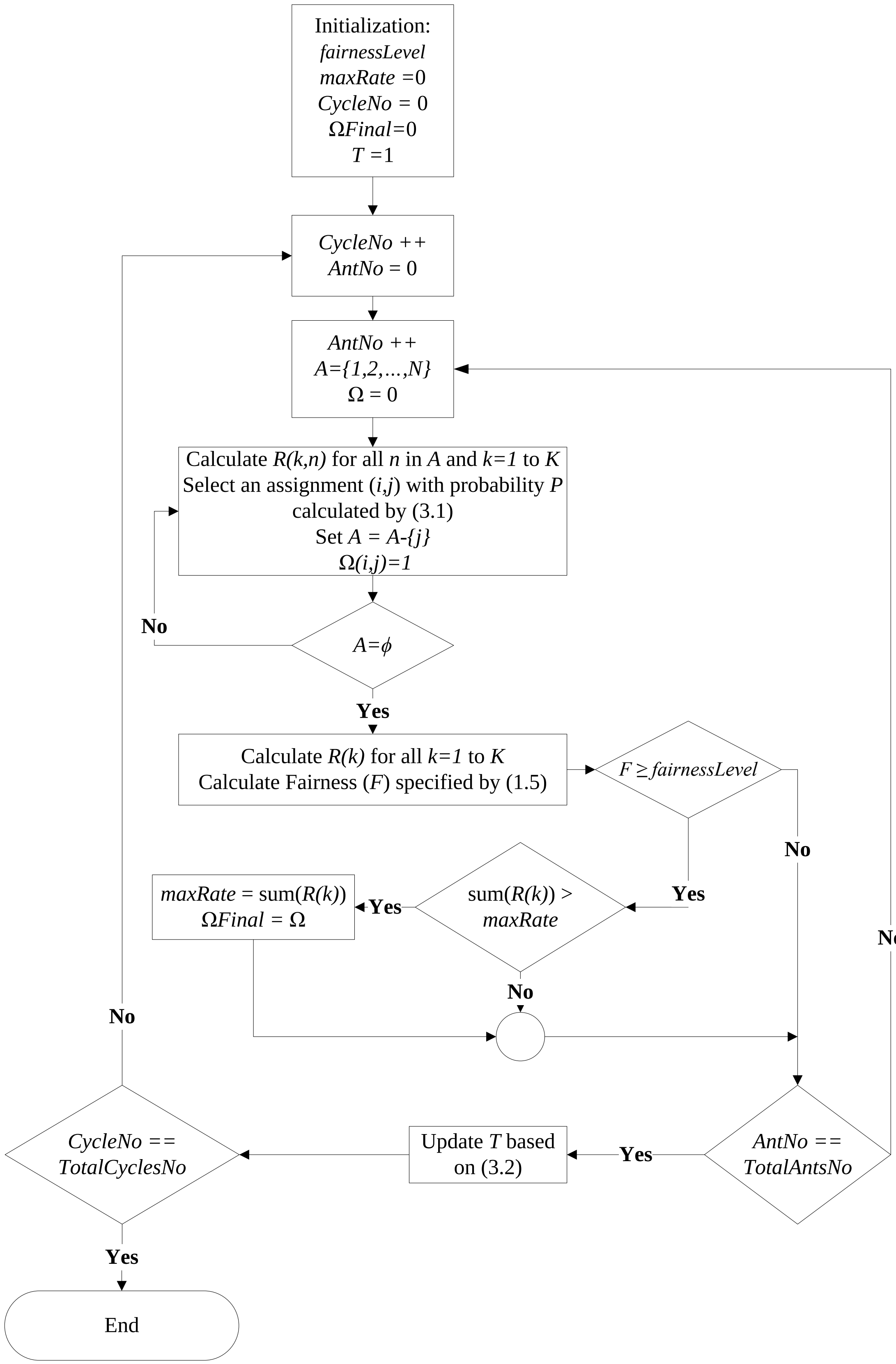

Figure 3.4: Flow Chart of ACO-based Algorithm Implementation

3.3 STAGE 2: ROUND ROBIN ASSIGNMENT WITH WATER-FILLING ALGORITHM

Fairness amongst users, in terms of satisfying the proportionality rate constraint, is one of the major objectives that need to be achieved in this thesis. In the ACO-based algorithm version 1, users' assignments are random and not driven by any method. Therefore, the fairness between users will not be affected. The fairness will be evaluated at the end of the assignments; if the minimum fairness level is achieved, then the assignments are considered, otherwise they are rejected.

Therefore, some enhancements should be added to improve the fairness between users and guarantee the maximum possible users' data rate (total system capacity). To increase the rate, the ACO-based algorithm version 2 whose pseudo code is shown in Figure 3.5 uses water-filling algorithm for power allocation to increase the total system capacity as the studies in [5, 7, 8]. Water-filling algorithm assigns more power to the sub-channels with the higher gain. Simultaneously, the fairness is enhanced in the ACO-based algorithm version 2 by enhancing the assignments of sub-channels to users. In the ACO-based algorithm version 2, the sub-channels are assigned to the users in a round robin fashion which enhances the fairness between users and the effect of $fairnessLevel$ parameter is reduced.

ACO-based Algorithm Version 2:

0) Initialization
   a. Initialize N to number of sub-channels, K to number of users, P_{Total} to the system total power, $cyclesNumber$, $antsNumber$, and $fairnessLevel$
   b. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{T_n^k(t) = 1, \Omega_{Final}(k,n) = 0, p_{Final}(k,n) = 0$, Generate $H(k,n)\}$ where $T_n^k(t)$, $\Omega_{Final}(k,n)$, and $p_{Final}(k,n) = 0$ are the trail intensity, the final sub-channel allocation, and the final power allocation for each sub-channel n to the user k, respectively.
   c. For $k = 1$ to $K \rightarrow \{maxRate(k) = 0\}$
   d. $S = \{1, 2, \dots, K\}$
1) For $t = 1$ to $cyclesNumber$
   a. For $a = 1$ to $antsNumber$
      i. Initialize $A = \{1,2,\dots,N\}, consumedPower = 0, userNumber = 1$
      ii. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{\Omega(k,n) = 0, p(k,n) = 0\}$ where $\Omega(k,n)$ and $p(k,n)$ are the sub-channel allocation and power allocation for each assignment of sub-channel n to user k of the a^{th} ant.
      iii. While $A \neq \emptyset$,
         1. Calculate $p_{Temp}(userNumber, n)\ \forall n \in A$ using water-filling algorithm by distributing ($P_{Total} - consumedPower$) on the available sub-channels.
         2. Calculate $R(userNumber, n)\ \forall n \in A$ where $R(userNumber, n) = \frac{1}{N} * \log_2(1 + p_{Temp}(userNumber, n) H(userNumber, n))$
         3. Assign sub-channel j to user ($userNumber$) where j is randomly generated using $P_j^{userNumber}$ specified by (3.1), where $j \in A$.
         4. $A \coloneqq A - \{j\}$
         5. $consumedPower \coloneqq consumedPower + p_{Temp}(userNumber, j)$
         6. $\Omega(userNumber, j) = 1$
         7. $p(userNumber, j) = p_{Temp}(userNumber, j)$
         8. If $userNumber == K$ then $userNumber = 1$, else $userNumber{+}{+}$
      iv. Calculate $R(k)\ \forall k \in S$ where $R(k) = \sum_{n \in \Omega_k} \frac{1}{N} * \log_2(1 + p(k,n)H(k,n))$
      v. Calculate fairness (F) such that $F = (\sum_{k=1}^{K} \frac{R_k}{\gamma_k})^2 / (\left(\sum_{k=1}^{K} (\frac{R_k}{\gamma_k})^2\right) * K)$
      vi. If ($F \geq fairnessLevel$)
         1. If ($\sum_{\forall k \in S} R(k) > \sum_{\forall k \in S} maxRate(k)$)
            a. $maxRate(k) = R(k), \forall k \in S$
            b. $\Omega_{Final}(k,n) = \Omega(k,n) \forall k \in S$ and $n = \{1, 2, \dots, N\}$
   b. For $k = 1$ to K, For $n = 1$ to $N \rightarrow$ {update $T_n^k(t)$ based on (3.2)}

Figure 3.5: ACO-based Algorithm Version 2

3.4 STAGE 3: FINAL IMPLEMENTATION

The main problem in this thesis work is to find a solution for the resource allocation problem of OFDMA systems that maximizes the total system capacity while satisfying the proportional rate constraint using the ACO-based algorithm. Version 1 and version 2 of the ACO-based algorithm did not satisfy the proportional rate constraint. Therefore, the assignments of the sub-channels to the users should be driven to satisfy the maximum fairness based on the proportional rate constraint. In order to achieve this objective, the ACO-based algorithm version 3 in Figure 3.6 assigns the sub-channels to the users with the least normalized data rate over proportional rate constants. This enhanced the fairness amongst users.

The ACO-based algorithm version 3 uses water-filling algorithm as in version 2 of the algorithm. Additionally, the ACO-based algorithm version 3 uses water-filling algorithm after each assignment of sub-channel n to user k to redistribute the total power of user k on k user's sub-channels as the study in [7].

This version (version 3) represents the final implementation of the ACO-based algorithm that finds a solution for the resource allocation problem in the OFDMA systems that maximizes the total system capacity while satisfying the proportional rate constraint. Therefore, most of the parametric studies and the comparisons will be done for the ACO-based algorithm version 3.

ACO-based Algorithm Version 3:

0) Initialization
   a. Initialize N to number of sub-channels, K to number of users, P_{Total} to the system total power, $cyclesNumber, antsNumber$, and $fairnessLevel$
   b. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{\, T_n^k(t) = 1, \Omega_{Final}(k,n) = 0, p_{Final}(k,n) = 0$, Generate $H(k,n)\}$ where $T_n^k(t)$, $\Omega_{Final}(k,n)$, and $p_{Final}(k,n) = 0$ are the trail intensity, the final sub-channel allocation, and the final power allocation for each sub-channel n to the user k, respectively.
   c. For $k = 1$ to $K \rightarrow \{maxRate(k) = 0\}$
   d. $S = \{1, 2, \dots, K\}$
1) For $t = 1$ to $cyclesNumber$
   a. For $a = 1$ to $antsNumber$
      i. Initialize $A = \{1,2, \dots, N\}, consumedPower = 0$
      i. For $k = 1$ to K, For $n = 1$ to $N \rightarrow \{\Omega(k,n) = 0, p(k,n) = 0\}$ where $\Omega(k,n)$ and $p(k,n)$ are the sub-channel allocation and power for each assignment of sub-channel n to the user k of the a^{th} ant.
      ii. While $A \neq \emptyset$,
         1. Find user i such that $R_i/\gamma_i \leq R_k/\gamma_k \; \forall k \in S$, where $i \in S$
         2. Calculate $p_{Temp}(i,n) \; \forall n \in A$ using water-filling algorithm to distribute ($P_{Total} - consumedPower$) on the available sub-channels.
         3. Calculate $R(i,n) \; \forall n \in A$ where $R(i,n) = \frac{1}{N} * \log_2(1 + p_{Temp}(i,n)H(i,n))$
         4. Assign sub-channel j to user i where j is randomly generated using P_j^i specified by (3.1), where $j \in A$.
         5. $A \coloneqq A - \{j\}$
         6. $consumedPower \coloneqq consumedPower + p_{Temp}(i,j)$
         7. $\Omega(i,j) = 1$
         8. $p(i,j) = p_{Temp}(i,j)$
         9. Calculate the new $p(i,n)$ for all the n sub-channels that are assigned to user i by redistributing the total power of user i on the sub-channels of user i using water-filling algorithm.
      iii. Calculate $R(k) \; \forall k \in S$ where $R(k) = \sum_{n \in \Omega_k} \frac{1}{N} * \log_2(1 + p(k,n)H(k,n))$
      iv. Calculate fairness (F) where $F = (\sum_{k=1}^{K} \frac{R_k}{\gamma_k})^2 / (\left(\sum_{k=1}^{K} (\frac{R_k}{\gamma_k})^2\right) * K)$
      v. If ($F \geq fairnessLevel$)
         1. If ($\sum_{\forall k \in S} R(k) > \sum_{\forall k \in S} maxRate(k)$)
            a. $maxRate(k) = R(k), \forall k \in S$
            b. $\Omega_{Final}(k,n) = \Omega(k,n) \forall k \in S$ and $n = \{1, 2, \dots, N\}$
   b. For $k = 1$ to K, For $n = 1$ to $N \rightarrow$ {update $T_n^k(t)$ based on (3.2)}

Figure 3.6: ACO-based Algorithm Version 3

Chapter 4

RESULTS AND DISCUSSIONS

In this chapter, we will discuss the results for some simulations that have been performed. First, we will describe the simulation parameters and we will state the justification for such choice. Then, a detailed discussion of the findings is given for the implemented algorithms in Chapter 3.

4.1 SIMULATION PARAMETERS

In this study, simulations are performed for the same OFDMA system in [5]. The channel model is consisting of 6 independent Rayleigh multipaths. The study assumes a maximum delay spread of 5 μs and maximum doppler of 30 Hz. The total system bandwidth is assumed as 1 MHz and the total sub-channels are assumed as 64. We assume a total of 1 Watt for total system power. The power spectral density for noise is equal to -65 dBW per Hz.

The ACO-based algorithm parameters are selected based on some parametric studies that match the usage of the parameters in the literature [11]. In the simulations, constants α and β in (3.1) are assumed to be 1. This will make the effect of the trail intensity similar to the effect of the desirability. The constant Q in (3.4) is assumed to be 100.

4.2 SIMULATION RESULTS

First, some parametric studies are performed on the ACO-based algorithm to study the effect of the ACO-based algorithm parameters on the total system capacity and to select the suitable parameters for the simulations. First parametric studies are done on version 1 of the ACO-based algorithm to study the effect of both the fairness level and the evaporation coefficient against the total system capacity. We select version 1 of the algorithm because the allocations of the sub-channels to the users are totally random where the users are not selected before the sub-channels as in the other two versions. Moreover, version 1 aims to allocate the sub-channels and the power that maximize the total system capacity without ensuring any type of fairness between users. Therefore, we did not make the study for version 2 and version 3 of the ACO-based algorithm because the allocations in these two versions are driven to satisfy some fairness between users as discussed in section 3.3 and section 3.4.

Figure 4.1 shows the result of the study of total system capacity in (bit/s/Hz) against the fairness level for the OFDMA system in section 4.1 for $K = 16$ users and $N = 64$ sub-channels. The proportional rate constants (PRCs) are equal, i.e., $\gamma_{\mathrm{i}} = 1, \forall\ i = 1, 2, \dots, K$. The number of cycles and the number of ants are equal to 100, while ρ is equal to 0.8. The simulation is performed for one channel realization (only one gain matrix) because we need to study the effect of the ACO-based algorithm parameters on the results for the same gain matrix. The other ACO-based algorithm parameters are the same as in section 4.1. Figure 4.1 shows that there is a trade-off between the total system capacity and the fairness. The result shows that there is no solution with fairness level higher than 0.8. This is because version 1 of the ACO-based algorithm assigns the sub-channels to the users in order to maximize the total system capacity without ensuring any fairness between users. Therefore, the best choice of $fairnessLevel$ that maximizes the total system capacity is from 0.1 to 0.3.

The effect of the evaporation coefficient (ρ) is shown in Figure 4.2 for the OFDMA system and the ACO-based algorithm parameters in section 4.1 for $K = 16$ users and $N = 64$ sub-channels. The PRCs are equal, i.e., $\gamma_{\mathrm{i}} = 1, \forall\, i = 1, 2, \dots, K$. The $fairnessLevel$ is equal to 0.2 to ensure the maximum total system capacity, while the number of cycles and the number of ants are equal to 100. Also, the simulation is performed for one channel realization. Figure 4.2 shows that as the value of the evaporation coefficient increases, the total system capacity increases since the trail intensity of good solutions will be high for high values of evaporation coefficients. At the same time, the trail intensity of good solutions will be low for low values of evaporation coefficients which allow the bad solutions to be selected. Based on the literature [11, 57], the best value of ρ is 0.8. This value matches good results in Figure 4.2. Therefore, we use ρ to be 0.8 for the other simulations.

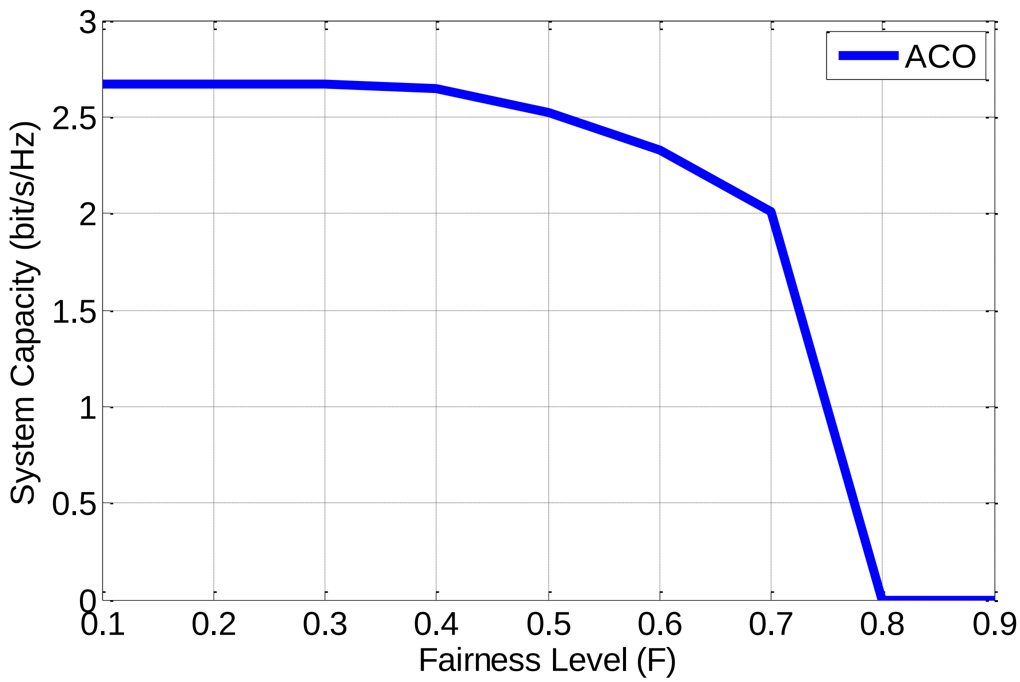

Figure 4.1: System Capacity versus Fairness Level

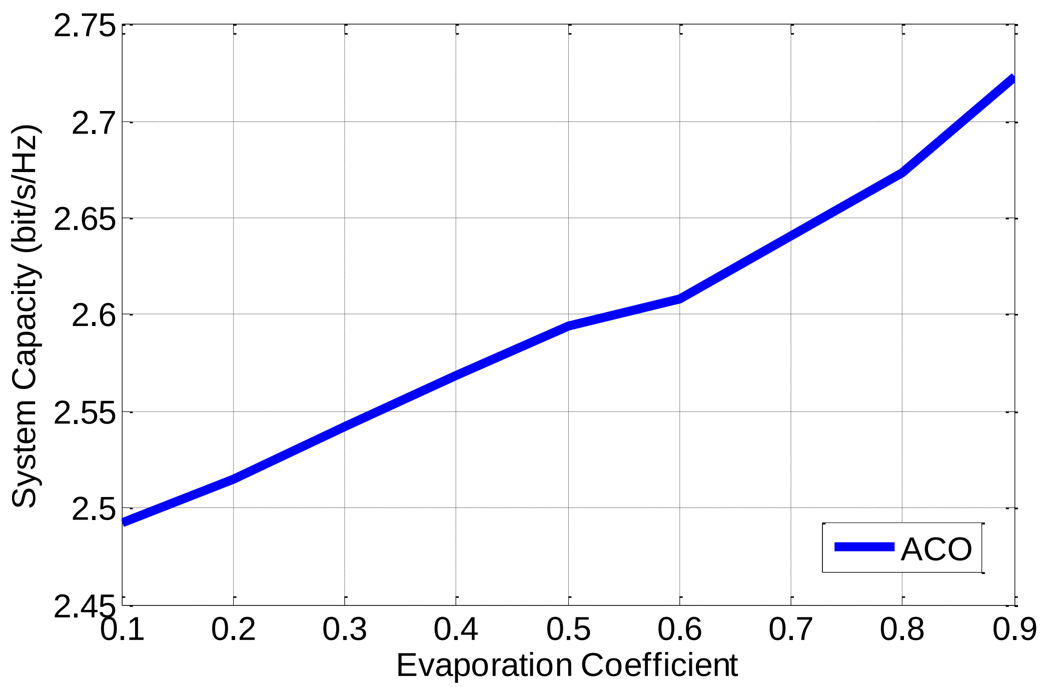

Figure 4.2: System Capacity versus Evaporation Coefficient (ρ)

Other parametric studies are performed for the same OFDMA system in section 4.1 for $K = 10$ users and $N = 64$ sub-channels. The PRCs are equal, i.e., $\gamma_{\mathrm{i}} = 1, \forall\, i = 1, 2, \dots, K$. The channel realization is one where the study is performed 100 times for the same channel gain matrix and then the average of the results is taken. Based on the previous parametric studies, $fairnessLevel = 0.2$ and $\rho = 0.8$. The parametric study is done to show the effect of the number of ants and the number of cycles on the total system capacity. This study is done for ACO-based algorithm version 3. Figure 4.3 shows the effect of the number of ants on the total system capacity for a given number of cycles, while Figure 4.4 shows the effect of the number of cycles on the total system capacity for a given number of ants. Both figures (Figure 4.3 and Figure 4.4) show that as the number of ants or the number of cycles increase, the total system capacity saturates. In Figure 4.3, the total system capacity has the highest value for 90 cycles and for ants higher than 70. At the same time, the total system capacity has the highest value for 90 ants and cycles higher than 80. Therefore, the best configurations for the number of cycles and the number of ants are 100 cycles and 100 ants to make sure that we have the maximum total system capacity. This configurations match the configurations used in the literature [11].

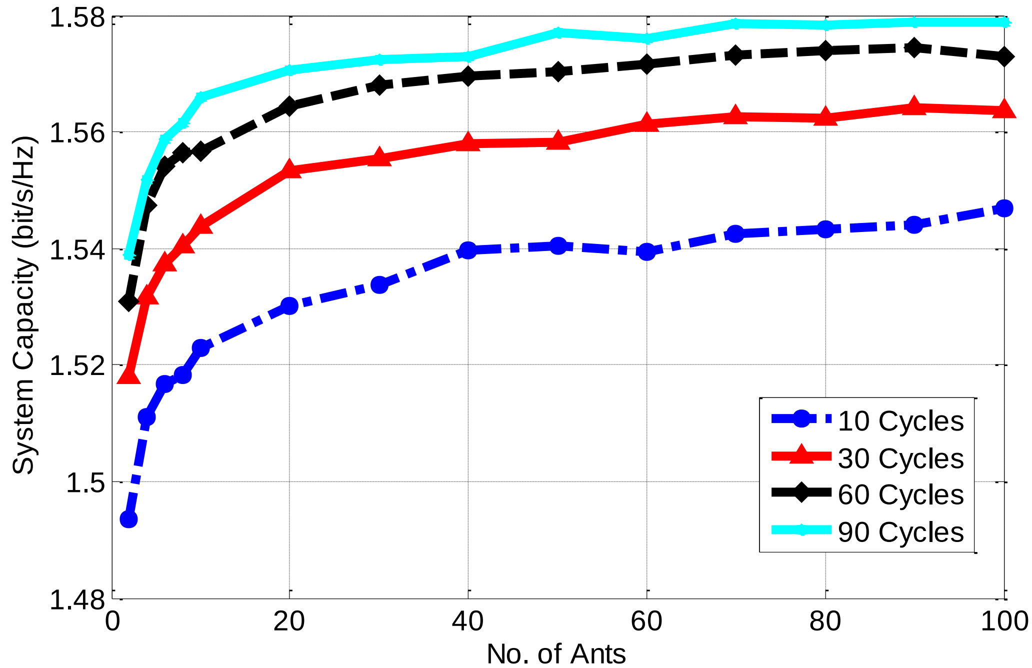

Figure 4.3: System Capacity versus Number of Ants

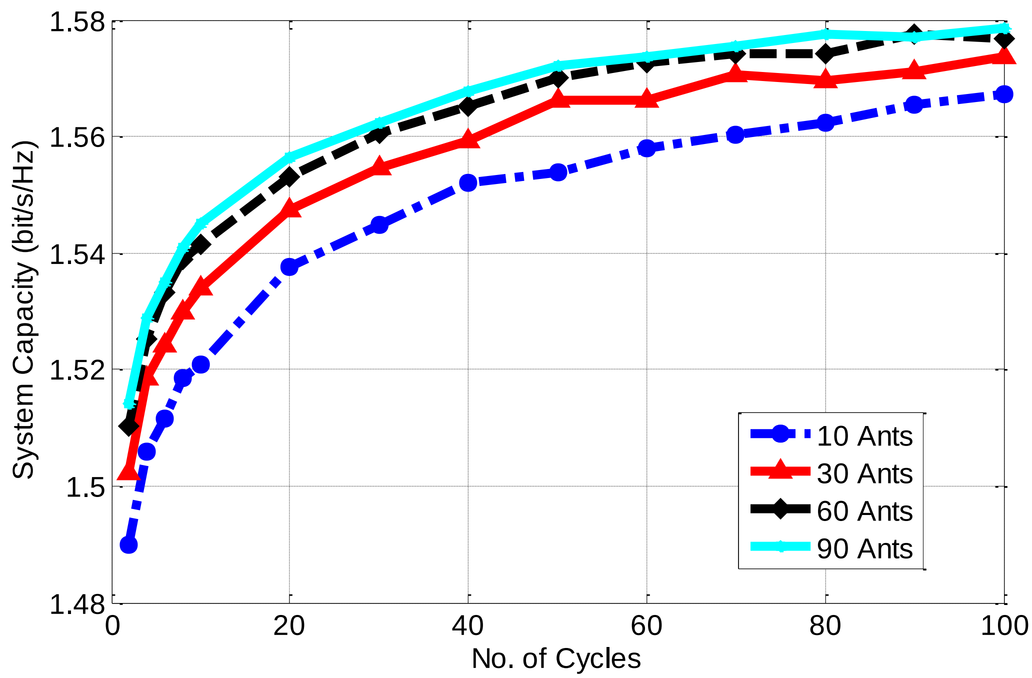

Figure 4.4: System Capacity versus Number of Cycles

A number of simulations are performed for the three versions of the ACO-based algorithm for different PRCs. In the simulations, the results of each version are compared with the results of the algorithms in [1, 5, 7, 8].

First, the simulations are performed for version 1 of the ACO-based algorithm for the same OFDMA system in section 4.1 for $N = 64$ sub-channels and users vary from 2 to 16. The simulations are performed for 1000 channel realizations where the algorithms are applied on 1000 different channel gain matrix, and then the average results are taken. Based on the previous parametric studies, $fairnessLevel = 0.2$ and $\rho = 0.8$. Also, the number of cycles and the number of ants are 100. The other simulation parameters are the same as the parameters in section 4.1. The simulation results of the ACO-based algorithm version 1 are shown in Figure 4.5 to Figure 4.10. The PRCs for Figure 4.5 and Figure 4.6 are equal, i.e., $\gamma_{\mathrm{i}} = 1, \forall\, i = 1, 2, \dots, K$. The PRCs for Figure 4.7 and Figure 4.8 are not equal, i.e., $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 8\ \forall\ (i\%2) = 0$ for $i = 1, 2, \dots, K$. Finally, $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 16\ \forall\ (i\%2) = 0$ for $i = 1, 2, \dots, K$ are the PRCs for Figure 4.9 and Figure 4.10.

The simulations evaluate the total system capacity in (bit/s/Hz) and the fairness index against the number of users. The simulations are preformed for three different PRCs parameters to study the effect of the PRCs on the fairness index between users. First simulation is performed for equal PRCs where $\gamma_{\mathrm{i}} = 1, \forall\, i = 1, 2, \dots, K$ to maximize the overall rate while trying to achieve equal rate for all users [7]. The other two simulations are performed for different PRCs between users to study the ability of the ACO-based algorithm to attain the fairness between users based on different PRCs and to study the effect of high values of PRCs on it.

The results in Figure 4.5, Figure 4.7, and Figure 4.9 show that as the number of users increases, the total system capacity increases; because the opportunity of having better reordering of assignments increases. The results in Figure 4.6, Figure 4.8, and Figure 4.10 show that as the number of users' increases, the fairness index decreases due to the difficulty of achieving better fairness between users.

From the results, we notice that the PRCs affect the fairness index. The fairness index in Figure 4.8 is worse than that in Figure 4.6 due to the difference in the PRCs. Also, the results show that as the difference in the PRCs increases, the fairness decreases since the fairness index in Figure 4.10 is worse than that in Figure 4.8. The difference in the PRCs between users will increase the gap between them and reduce the fairness index. Therefore, the fairness index will decrease when the difference in the PRCs increased.

Moreover, the results in Figure 4.5, Figure 4.7, and Figure 4.9 show that the ACO-based algorithm has the highest total system capacity when compared to Rhee [1], Mohanram [7], Shen [8], and Mahmoud [5] algorithms. Also, the results in Figure 4.6, Figure 4.8, and Figure 4.10 show that the ACO-based algorithm has the lowest fairness index. From the results, we notice the relation and the trade-off between the total system capacity and the fairness. The results in Figure 4.6 for Rhee [1], Mohanram [7], and Mahmoud [5] algorithms show high value of fairness index almost equal to 1, while the ACO-based algorithm has the lowest value of fairness index. At the same time, the results of Rhee [1], Mohanram [7], and Mahmoud [5] algorithms in Figure 4.5 show much lower total system capacity than that of the ACO-based algorithm.

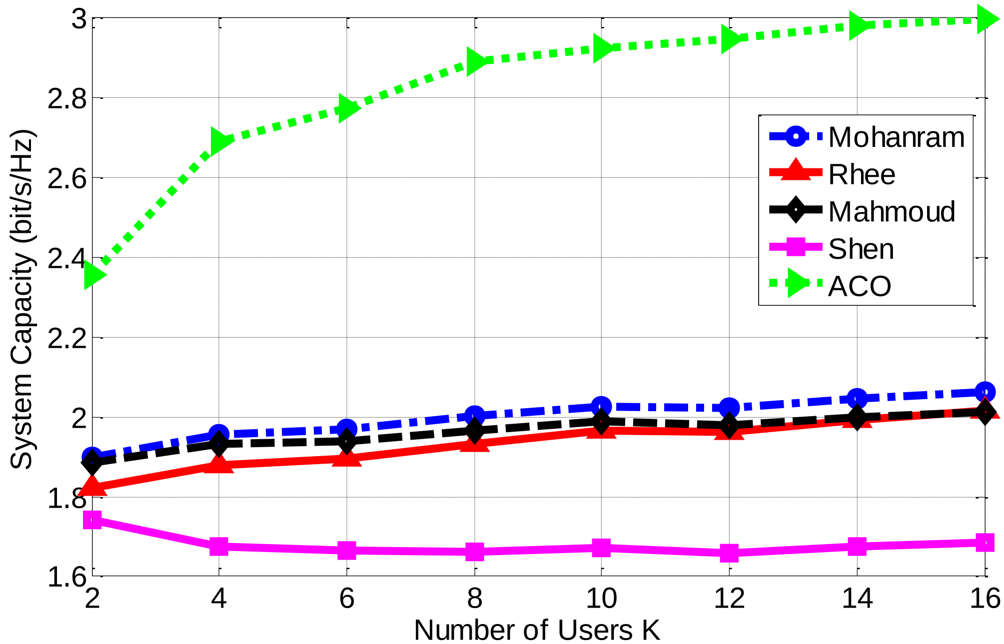

Figure 4.5: Comparison of total system capacity versus number of users for ACO-based algorithm version 1 and other algorithms for $\gamma_{\mathrm{i}} = 1, \forall\, i = 1, 2, \ldots, K$

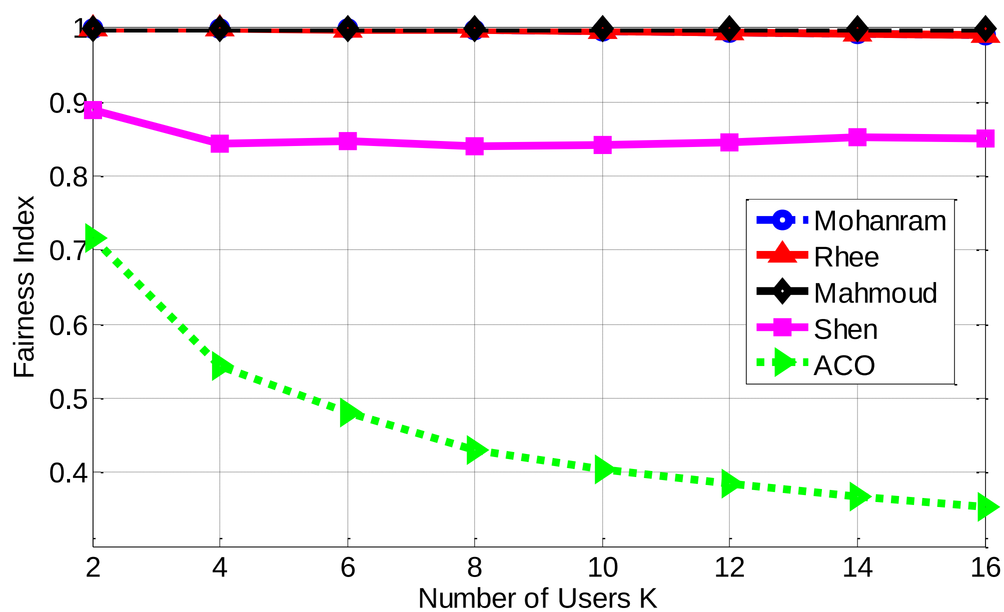

Figure 4.6: Comparison of fairness index versus number of users for ACO-based algorithm version 1 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \ldots, K$

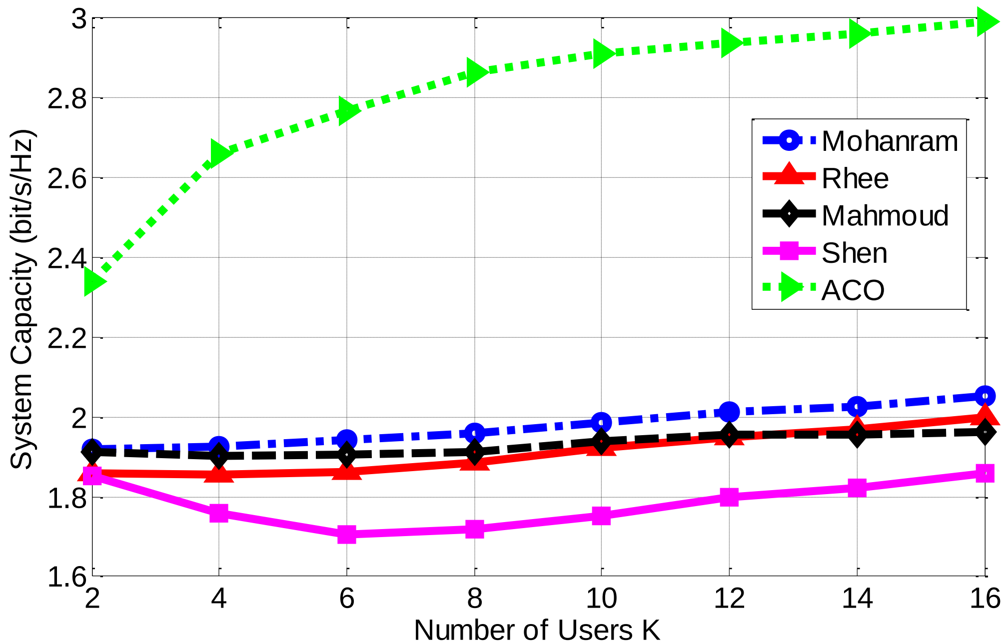

Figure 4.7: Comparison of total system capacity versus number of users for ACO-based algorithm version 1 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 8\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

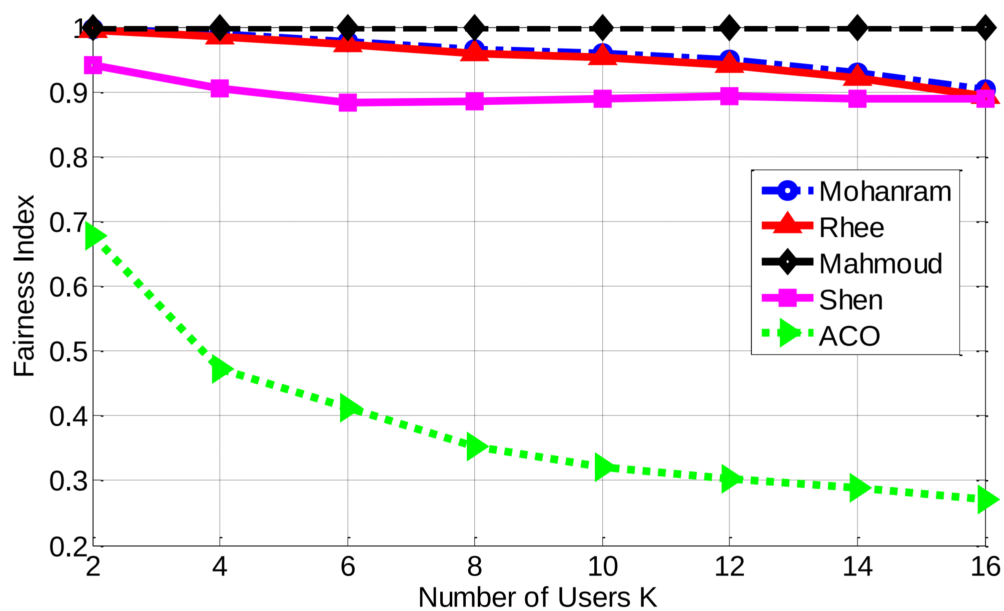

Figure 4.8: Comparison of fairness index versus number of users for ACO-based algorithm version 1 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 8\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

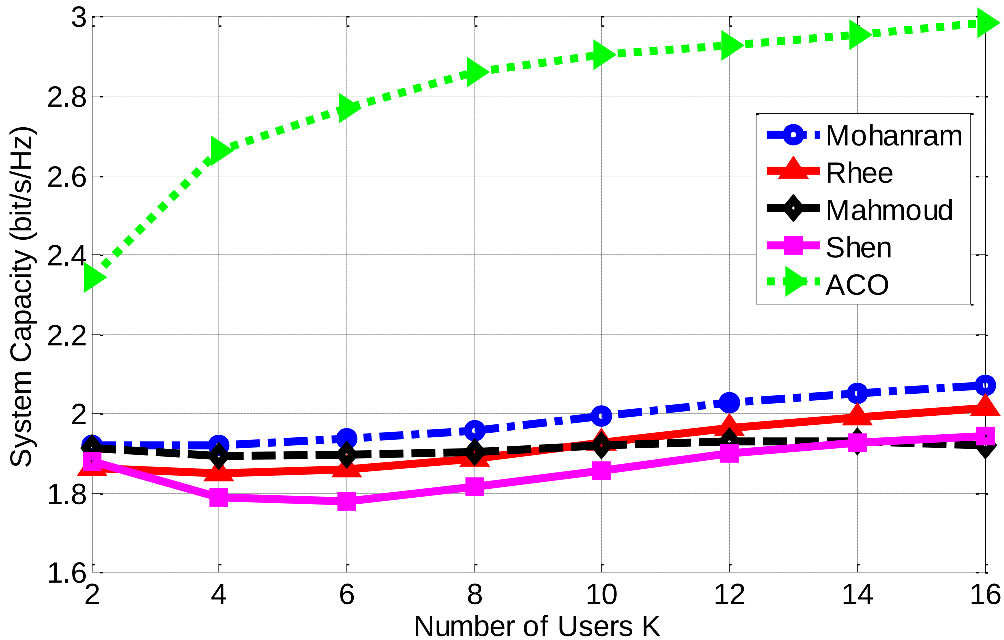

Figure 4.9: Comparison of total system capacity versus number of users for ACO-based algorithm version 1 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 16\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

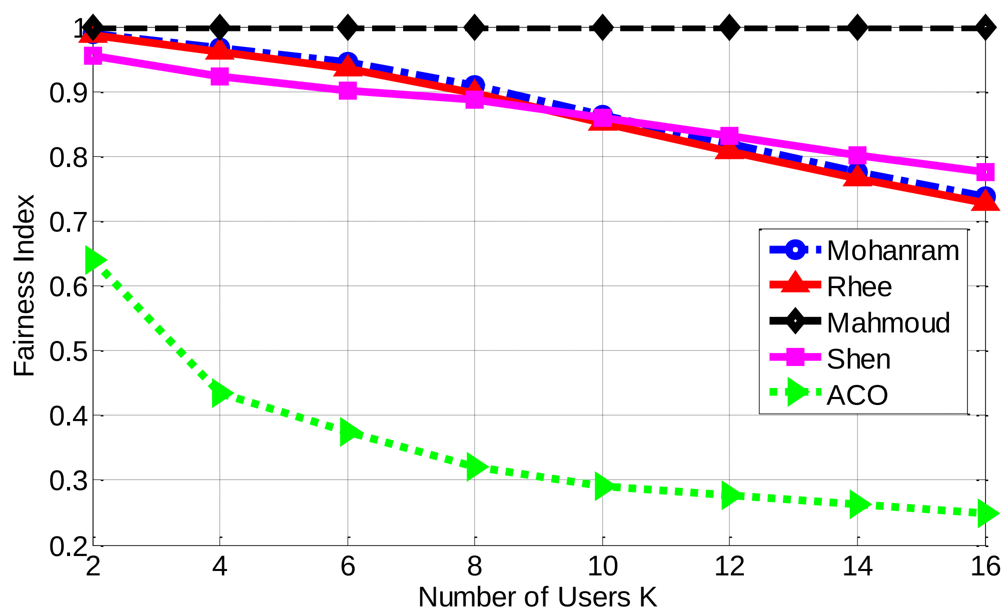

Figure 4.10: Comparison of fairness index versus number of users for ACO-based algorithm version 1 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 16\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

Second, the simulations are performed for version 2 of the ACO-based algorithm for the same OFDMA system in section 4.1 for $N = 64$ sub-channels and users vary from 2 to 16. The simulations are performed for 1000 channel realizations. Based on the previous parametric studies, $fairnessLevel = 0.2$ and $\rho = 0.8$. Also, the number of cycles and the number of ants are 100. The other simulation parameters are the same as the parameters in section 4.1. The simulation results of the ACO-based algorithm version 2 are shown in Figure 4.11 to Figure 4.16. The PRCs for Figure 4.11 and Figure 4.12 are equal, i.e., $\gamma_\mathrm{i} = 1, \forall\, i = 1, 2, \dots, K$. The PRCs for Figure 4.13 and Figure 4.14 are not equal, i.e., $\gamma_i = 1 \;\forall\; (i\%2) = 1$ and $\gamma_i = 8 \;\forall\; (i\%2) = 0$ for $i = 1, 2, \dots, K$. Finally, $\gamma_i = 1 \;\forall\; (i\%2) = 1$ and $\gamma_i = 16 \;\forall\; (i\%2) = 0$ for $i = 1, 2, \dots, K$ are the PRCs for Figure 4.15 and Figure 4.16.

The results show that the total system capacity of version 2 of the ACO-based algorithm is becoming 10% less than that of version 1 for $K = 2$ and as low as 24% for $K = 16$ and equal PRCs. At the same time, the fairness index is 35% higher for $K = 2$ and equal PRCs. But it still low for different PRCs in Figure 4.14 and Figure 4.16. The fairness becomes higher in version 2 due to the process of leading the assignments of sub-channels to users in a round robin fashion. The enhancement on the fairness for different PRCs is still not much since version 2 does not include the PRCs in its assignments. Moreover, Figure 4.12 shows that the fairness index of version 2 of the ACO-based algorithm is 8% higher than that of Shen [8] algorithm for $K = 8$.

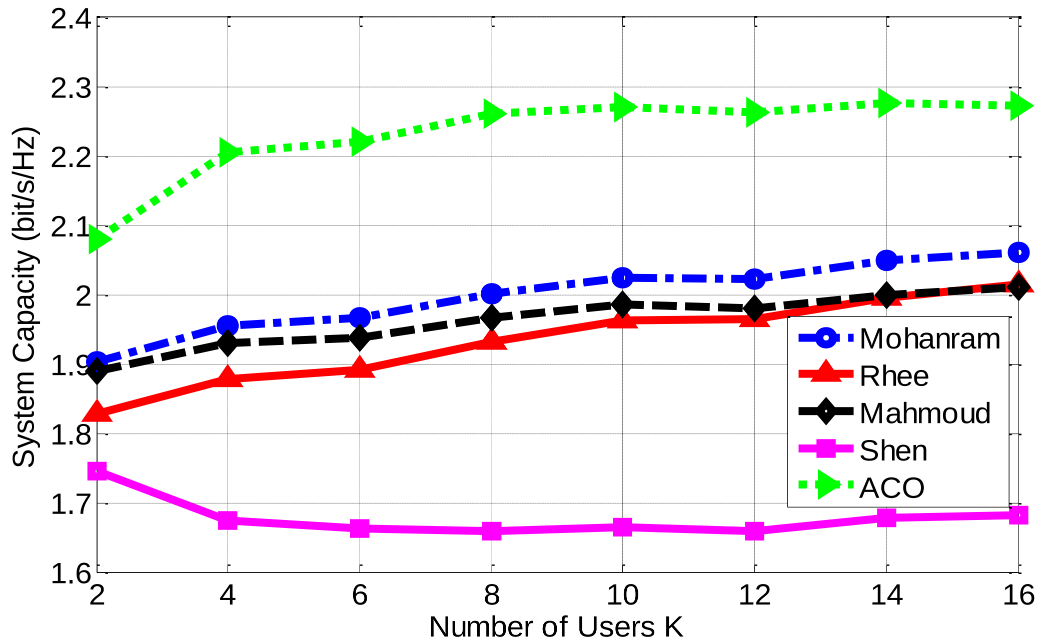

Figure 4.11: Comparison of total system capacity versus number of users for ACO-based algorithm version 2 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \ldots, K$

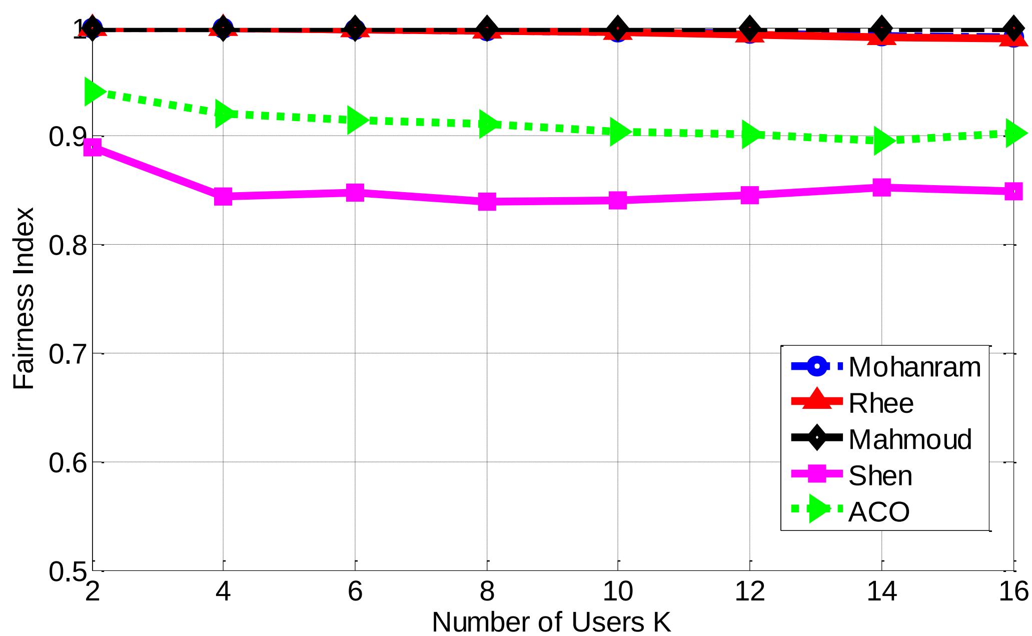

Figure 4.12: Comparison of fairness index versus number of users for ACO-based algorithm version 2 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \ldots, K$

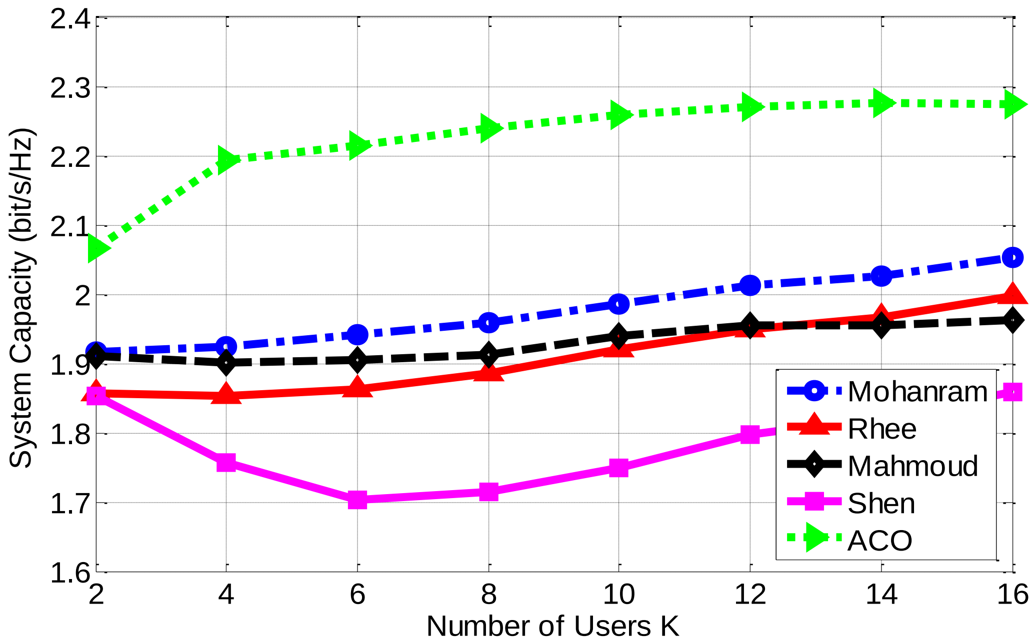

Figure 4.13: Comparison of total system capacity versus number of users for ACO-based algorithm version 2 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 8\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

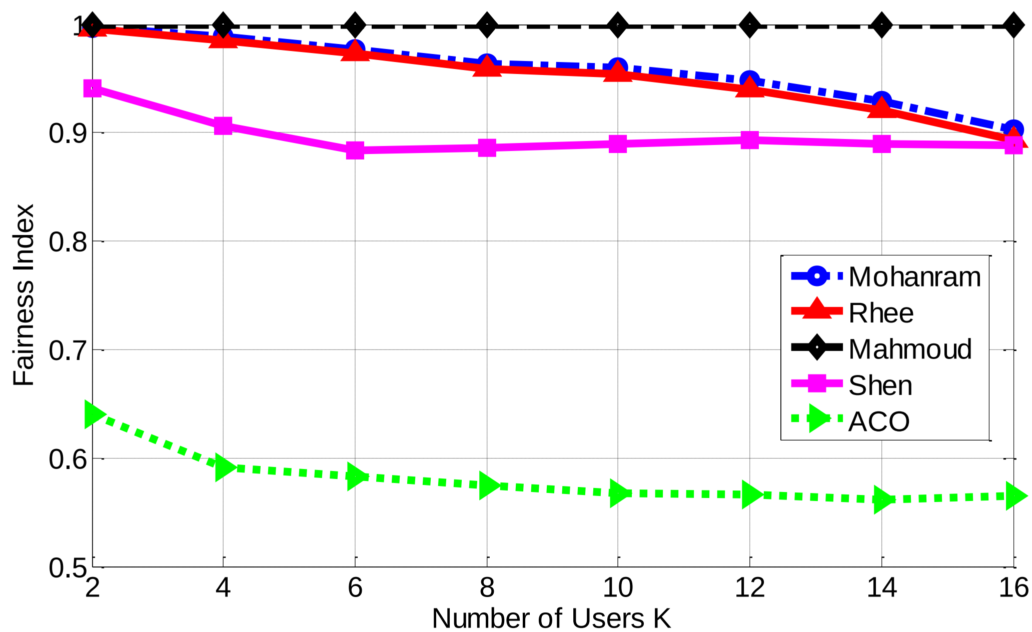

Figure 4.14: Comparison of fairness index versus number of users for ACO-based algorithm version 2 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 8\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

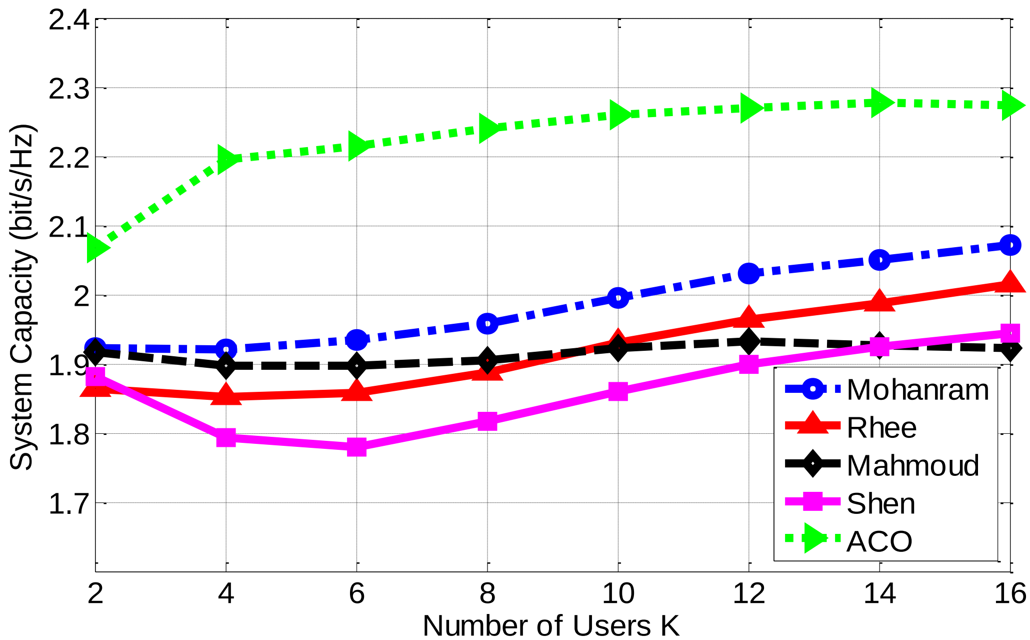

Figure 4.15: Comparison of total system capacity versus number of users for ACO-based algorithm version 2 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 16\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

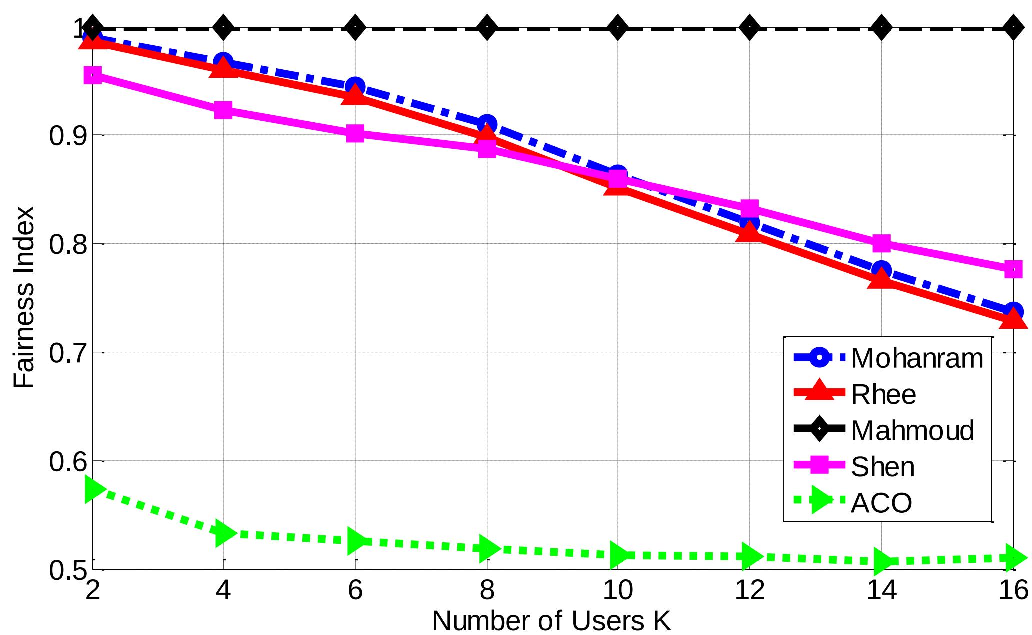

Figure 4.16: Comparison of fairness index versus number of users for ACO-based algorithm version 2 and other algorithms for $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 16\ \forall\ (i\%2) = 0$ for $i = 1, 2, \ldots, K$

Most importantly, the simulations of the final implemented ACO-based algorithm (version 3) are performed for the same OFDMA system in section 4.1 for $N = 64$ sub-channels and users vary from 2 to 16. The simulations are performed for 1000 channel realizations. Based on the previous parametric studies, $fairnessLevel = 0.2$ and $\rho = 0.8$. Also, the number of cycles and the number of ants are 100. The other simulation parameters are the same as the parameters in section 4.1. The simulation results of the ACO-based algorithm version 3 are shown in Figure 4.17 to Figure 4.22. The PRCs for Figure 4.17 and Figure 4.18 are equal, i.e., $\gamma_{\mathrm{i}} = 1, \forall\, i = 1, 2, \dots, K$. The PRCs for Figure 4.19 and Figure 4.20 are not equal, i.e., $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 8\ \forall\ (i\%2) = 0$ for $i = 1, 2, \dots, K$. Finally, $\gamma_i = 1\ \forall\ (i\%2) = 1$ and $\gamma_i = 16\ \forall\ (i\%2) = 0$ for $i = 1, 2, \dots, K$ are the PRCs for Figure 4.21 and Figure 4.22.

The results show that version 3 of the ACO-based algorithm enhances the fairness at the expense of the total system capacity. The total system capacity of version 3 becomes 3% less for $K = 2$ and as less as 9% for $K = 16$ than that of version 2 for equal PRCs. The fairness of version 3 for equal PRCs is almost one as shown in Figure 4.18. The fairness index for different PRCs in Figure 4.20 is good as it is higher than 0.9. Moreover, the fairness index in Figure 4.22 is high for users less than 10.

The results in Figure 4.17, Figure 4.19, and Figure 4.21 show that version 3 of the ACO-based algorithm has the highest total system capacity when compared with the other algorithms in [1, 5, 7, 8]. Additionally, the ACO-based algorithm version 3 has fairness index equal to that of the other algorithms in [1, 5, 7, 8] and better than them for users higher than 10 excluding Mahmoud [5] algorithm as shown in Figure 4.18. Mahmoud [5] algorithm is designed to have 100% of fairness between users. But the total system capacity of Mahmoud [5] algorithm is 4% less than that of the ACO-based algorithm.

The results in Figure 4.20 show that the ACO-based algorithm has the highest fairness when compared with Rhee [1], Mohanram [7], and Shen [8] algorithms. The results in Figure 4.22 show that the fairness index of Mohanram [7] algorithm is 1% better than that of the ACO-based algorithm for users less than 6. At the same time, the fairness index of Shen [8] algorithm is 3% better than the fairness of the ACO-based algorithm for $K = 12$ users.

From the results, it is clear that the ACO-based algorithm finds a solution to the resource allocation problem of OFDMA systems that outperforms the considered competing algorithms [1, 5, 7, 8] for most of the typical input parameters at the cost of prolonged execution time.

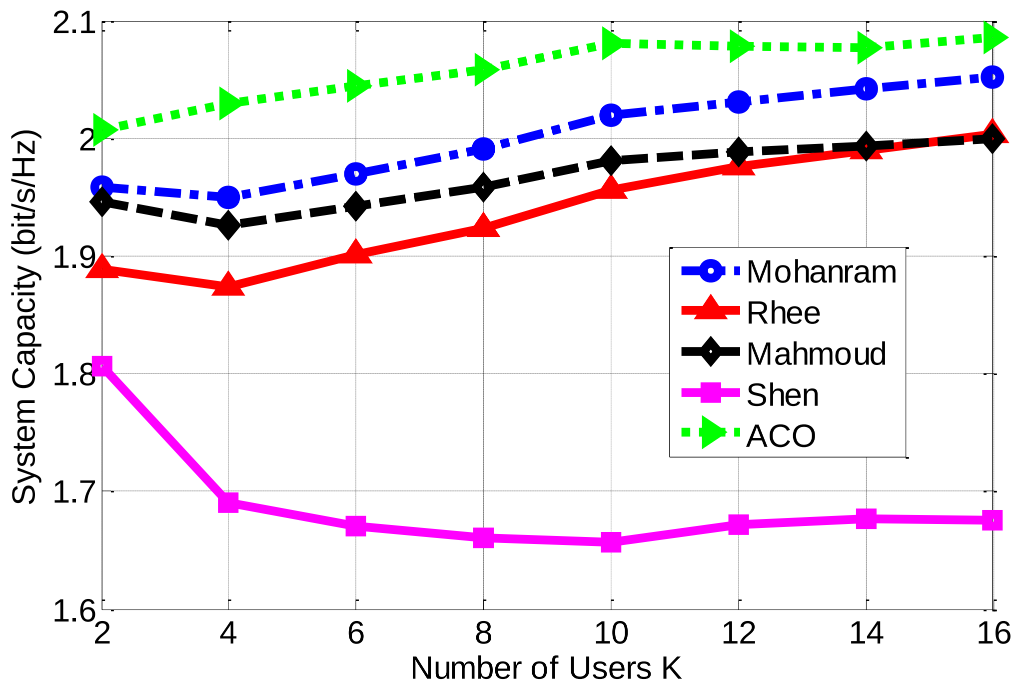

Figure 4.17: Comparison of total system capacity versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \dots, K$

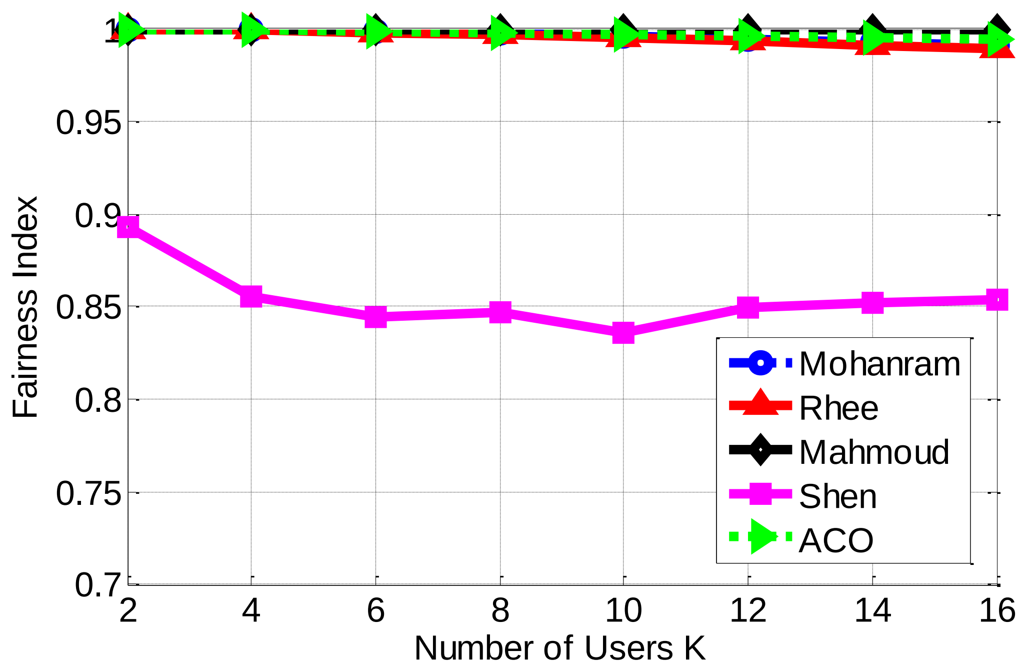

Figure 4.18: Comparison of fairness index versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \dots, K$

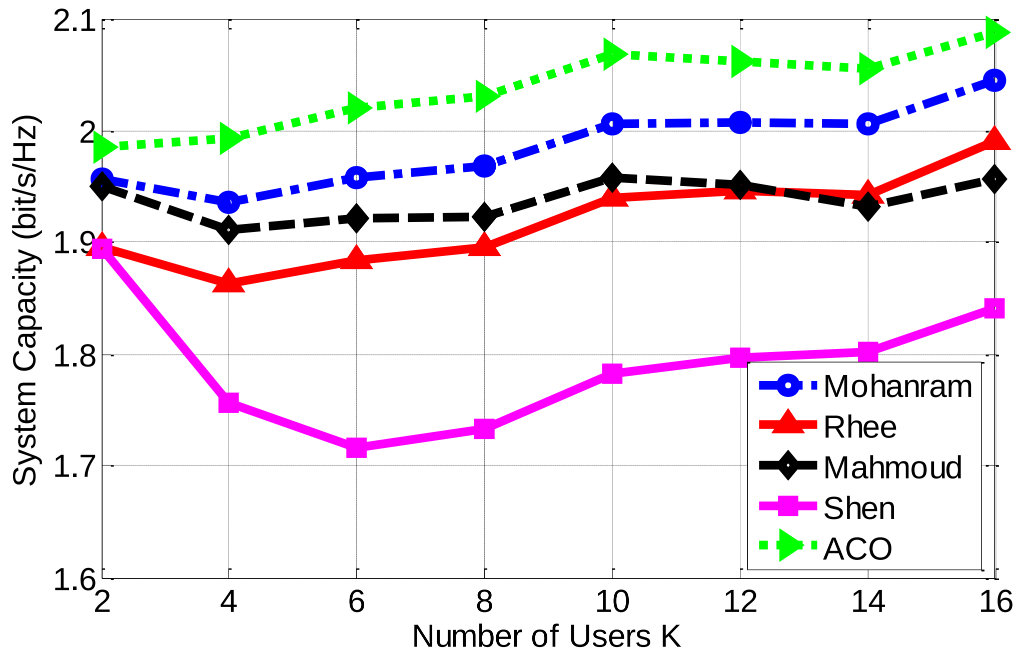

Figure 4.19: Comparison of total system capacity versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1 \; \forall \; (i\%2) = 1$ and $\gamma_i = 8 \; \forall \; (i\%2) = 0$ for $i = 1, 2, \ldots, K$

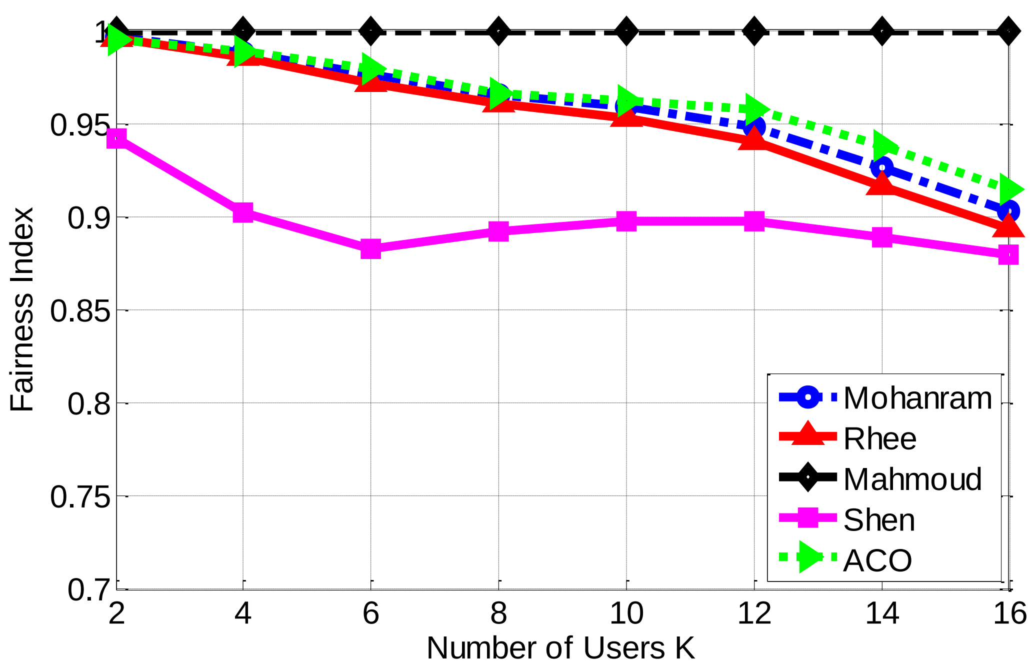

Figure 4.20: Comparison of fairness index versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1 \; \forall \; (i\%2) = 1$ and $\gamma_i = 8 \; \forall \; (i\%2) = 0$ for $i = 1, 2, \ldots, K$

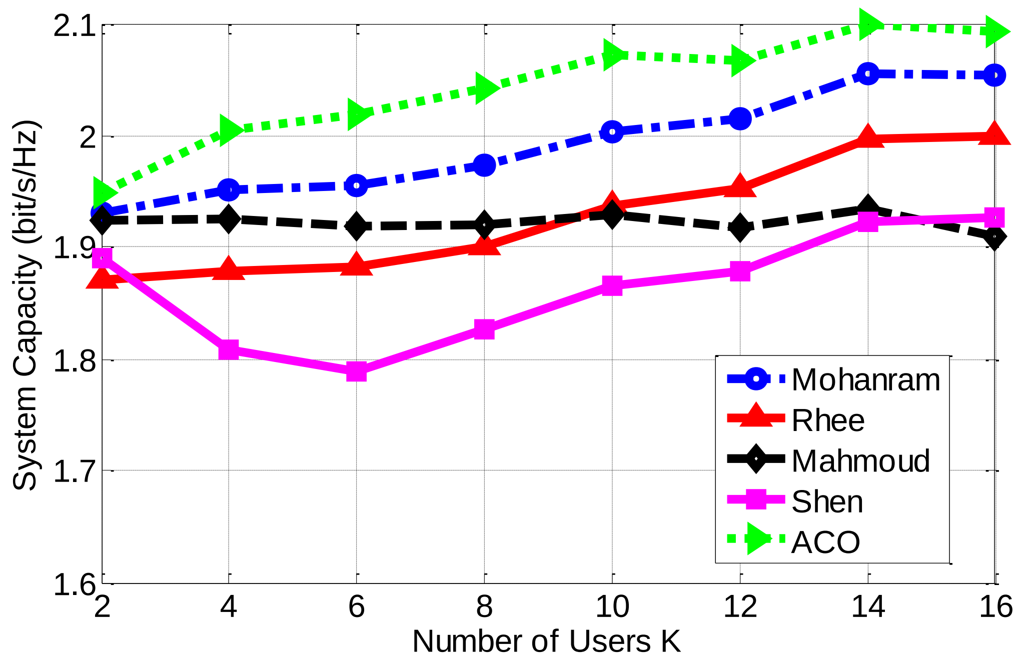

Figure 4.21: Comparison of total system capacity versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1 \; \forall \; (i\%2) = 1$ and $\gamma_i = 16 \; \forall \; (i\%2) = 0$ for $i = 1, 2, \ldots, K$

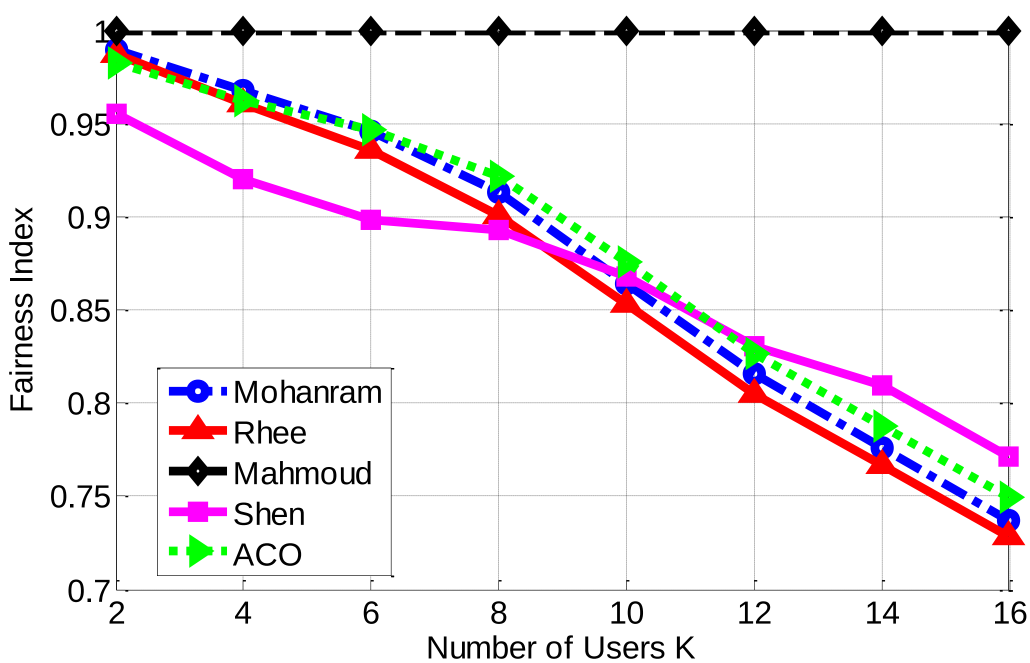

Figure 4.22: Comparison of fairness index versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1 \; \forall \; (i\%2) = 1$ and $\gamma_i = 16 \; \forall \; (i\%2) = 0$ for $i = 1, 2, \ldots, K$

Moreover, the same studies in Figure 4.17 and Figure 4.18 are repeated for 30 and 10 ants to study the effect of small number of ants on the results of the comparison with the other algorithms [1, 5, 7, 8]. Figure 4.23 and Figure 4.24 show the results for 30 ants. The results in Figure 4.23 show that the difference between the total system capacity of the ACO-based algorithm and Mohanram [7] algorithm is less than the difference in Figure 4.17. The results of the ACO-based algorithm in Figure 4.23 is 0.5% better than the results of Mohanram [7] algorithm for 16 users, while the results of the ACO-based algorithm in Figure 4.17 is 2% better than the results of Mohanram [7] algorithm for the same number of users. The fairness results in Figure 4.24 are similar to that in Figure 4.18. The results of the study for 10 ants are shown in Figure 4.25 and Figure 4.26. The fairness results in Figure 4.26 are the same as the other results in Figure 4.18 and Figure 4.24. The difference between the results of the ACO-based algorithm and Mohanram [7] algorithm in Figure 4.25 is less than that in Figure 4.23 where the results of the ACO-based algorithm is almost the same as Mohanram [7] algorithm for 14 and 16 users.

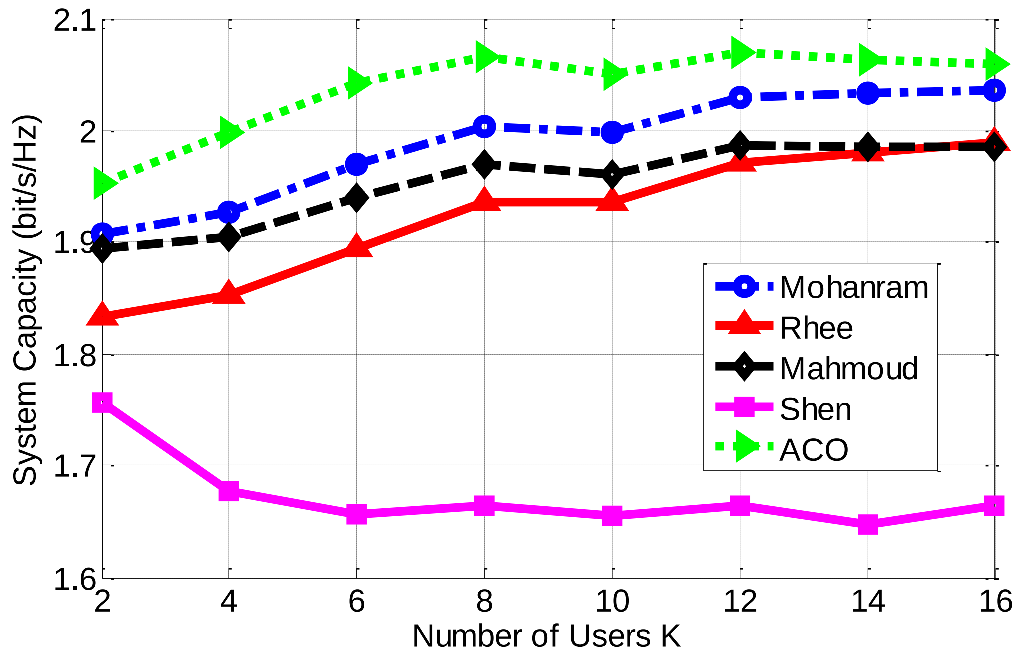

Figure 4.23: Comparison of total system capacity versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \ldots, K$ and 30 ants

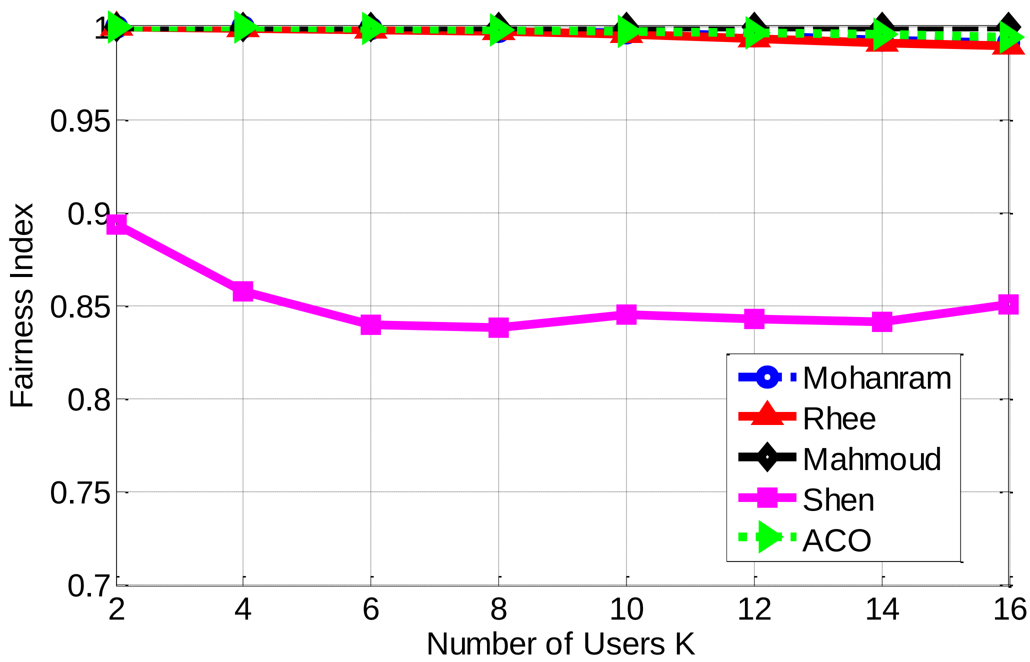

Figure 4.24: Comparison of fairness index versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1, \forall\, i = 1, 2, \ldots, K$ and 30 ants

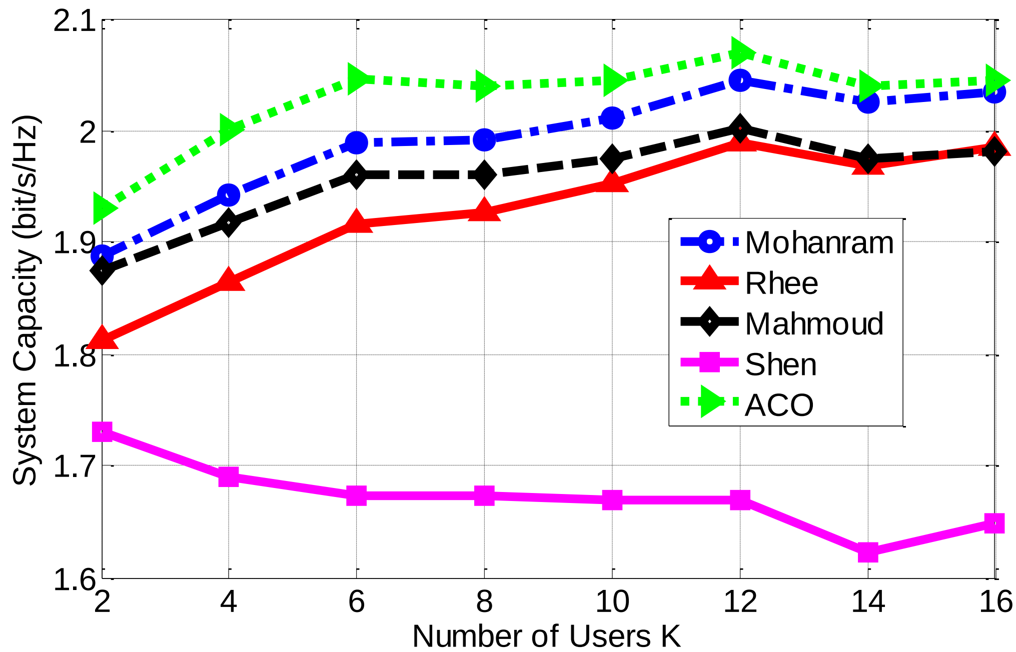

Figure 4.25: Comparison of total system capacity versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1, \forall\ i = 1, 2, \ldots, K$ and 10 ants

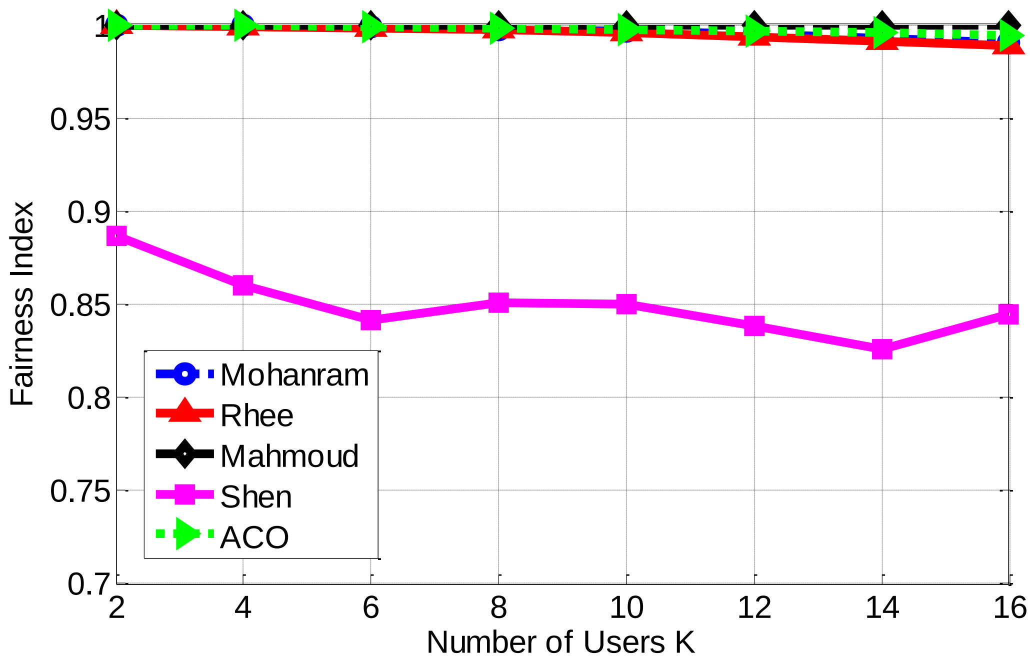

Figure 4.26: Comparison of fairness index versus number of users for ACO-based algorithm version 3 and other algorithms for $\gamma_i = 1, \forall\ i = 1, 2, \ldots, K$ and 10 ants

Based on the previous results, table 4-1 compares the last version of the ACO-based algorithm with the other algorithms in [1, 5, 7, 8]. The table shows the advantages and disadvantages of each algorithm.

Table 4-1: Comparison between Algorithms

	Advantages	**Disadvantages**
Mohanram [7]	- Allocates sub-channels to the users with the least data rate over proportional rate constants. - Uses water-filling algorithm after each assignment to redistribute the user's power on the assigned sub-channels. - Gets solution in a short time (0.013sec*).	- Initially assumes equal power for all sub-channels. - The allocated power for each user is proportional to the number of sub-channels allocated to that user.
Rhee [1]	- Gets solution in a short time (0.0067sec*).	- Does sub-channels allocations only. - Assumes equal power for each sub-channel. - Water-filling algorithm is not used. - Sub-channels are assigned to the users with the minimum data rate without taking care of the proportional rate constraint.

Mahmoud [5]	- Does power allocations for a given sub-channels allocations to ensure 100% of fairness between users. - Does not make assumptions about power as Rhee [1] and Mohanram [7] algorithms. - Computes the optimal power allocations and uses water-filling algorithm. - Satisfies the proportional rate constraint in the strictest sense. - Gets solution in a short time (0.11sec*).	- Does not make sub-channels allocations.
Shen [8]	- Does only power allocations for a given sub-channel allocations. - Finds the optimal power allocation and uses water-filling algorithm. - Gets solution in a short time (0.14sec*).	- Does not make sub-channels allocations. - Does not satisfy the proportional rate constraint for the general case because it assumes high and comparable sub-channel gains across the system bandwidth [5].
ACO	- Does both sub-channels and power allocations. - Does not make any assumptions about the power and uses water-filling for power allocations. - Looking comprehensively to find the best solution. - Allocates sub-channels to the users with the least data rate over proportional rate constants.	- Gets solution in a long time (809sec*).

* For OFDMA system in section 4.1 with 64 sub-channels and 16 users.

In the literature, most of the methods that solve the resource allocation problem of multiuser OFDMA systems find a suboptimal solution. None of the previous work compares its suboptimal solution with a reference solution like an optimal solution. Therefore, we will propose a novel method to synthesize an optimization problem with a known answer. The optimal solution will be our reference to compare our solution (by the ACO-based algorithm) and the other solutions in [1, 5, 7, 8] with it.

The optimal solution is synthesized by modifying the channel gain matrix. The process of synthesizing the required optimal solution is described as follows:

1. Get the maximum channel gain for each sub-channel.
2. Use water-filling algorithm to distribute the total power over the obtained sub-channels in step 1.
3. Assign the set of maximum channel gains to the users in a certain order (e.g. in round-robin manner).
4. Compute the users rates and the proportional rate constraints based on the following parameters:
    a. Channel gains obtained in step 1.
    b. Power allocation obtained in step 2.
    c. Channel assignments obtained in step 3.

The proportional rate constants is defined in [5] as “**the desired capacities resulting after solving the sub-channel and power allocation problem follow some specified ratios**” [5]. The proportional rate constants are estimated as a result of dividing the users’ rates by one of the user’s rate.

In order to evaluate the quality of the solutions, we normalize the solutions by the optimal solution. Then we find the distance between the normalized solutions and the normalized optimal solution which is the difference between them. Finally, we evaluate the standard deviation and the average distance for the distances. The standard deviation is used to give an idea about the distribution of the distances for each algorithm. The average distance is used to estimate the closeness of the solutions to the optimal solution. The average distance is the sum of the distances for each algorithm divided by the number of distances. Also, we evaluate the difference in the allocations of each algorithm and the allocations of the optimal solution which known as the distance in allocations. The distance in allocations is a result of the number of differences between the allocations of each algorithm and the allocations of the optimal solution divided by 2. The number of differences is the sum of the XOR of the allocation matrix of the optimal solution and the other algorithm that need to be studied.

In the simulations, the ACO-based algorithm and the algorithms in [1, 5, 7, 8] are applied on the new gain matrix specified by step 3. The solutions of the algorithms are compared with the optimal solution. The comparison is based on the average and standard deviation of the distance between the optimal solution of the synthesized problem and of the solutions obtained by other algorithms. Also, a distribution graph of each algorithm in [1, 5, 7, 8] is shown against both the optimal solution and the ACO-based algorithm, i.e. version 3, results. In addition, a graph for the distance in allocations is shown for Rhee [1] algorithm, Mohanram [7] algorithm, and the ACO-based algorithm.

In the performed simulations, two cases are used for the comparison with the optimal solution. One is small for $K = 4$ users. The other is large for $K = 10$ users. The cases are studied for the same OFDMA system in section 4.1 for $N = 64$ sub-channels. The simulations are performed for 20 channel realizations without averaging to show the distribution of the solutions and the distance between them clearly. The ACO-based algorithm parameters are the same as the parameters in section 4.1 with $fairnessLevel = 0.2$ and $\rho = 0.8$. Also, the number of cycles and the number of ants are 100.

Table 4-2 shows the results of the comparison for small size case. The ACO-based algorithm has the best average distance followed by Mohanram [7] algorithm. Also, the table shows that Rhee [1] algorithm has the highest standard deviation and the worst average distance. The distributions of the results are shown in Figure 4.27 to Figure 4.30. The comparison of the allocations of the algorithms against the allocation of the optimal solution is shown in Figure 4.31. For Shen [8] and Mahmoud [5] algorithms, the comparison of allocations are not stated because they use the same allocations of Rhee [1] algorithm. Figure 4.31 shows that the allocations of the ACO-based algorithm are more similar to the allocations of the optimal solution than the allocations of Rhee [1] and Mohanram [7] algorithms.

Table 4-2: Comparison with Optimal Solution for Small Size System

Algorithm \ Metric	Standard Deviation	Average Distance
Mohanram [7]	0.0225	0.0182
Rhee [1]	0.0323	0.0252
Mahmoud [5]	0.0231	0.0189
Shen [8]	0.0249	0.0208
ACO	0.0048	0.0069

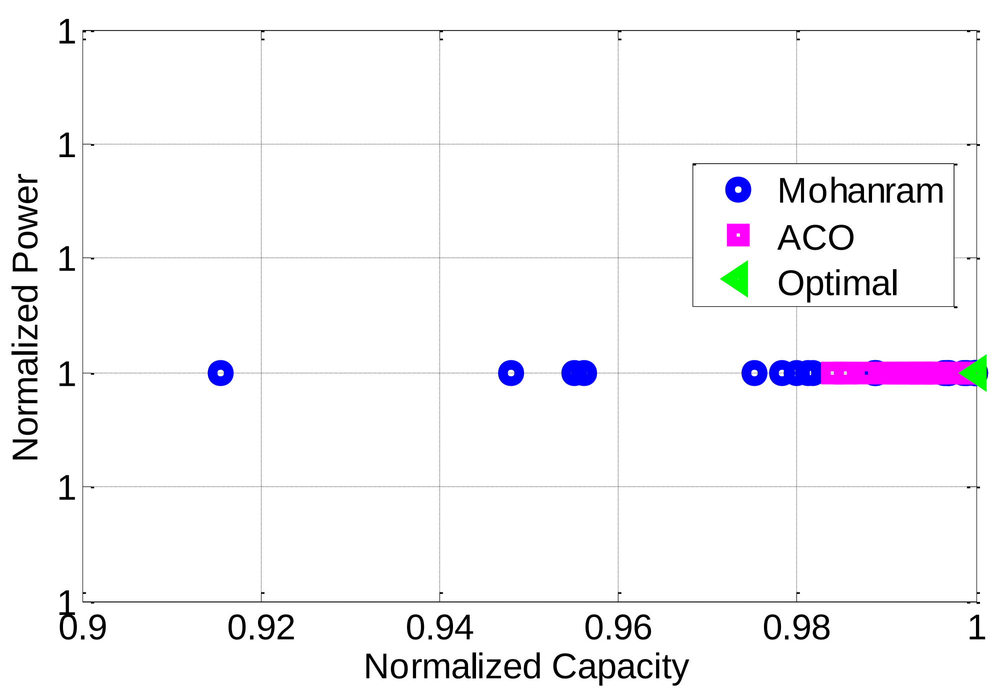

Figure 4.27: Distribution of the Results of Mohanram Algorithm versus the Results of ACO Algorithm and the Optimal Solution for Small Case

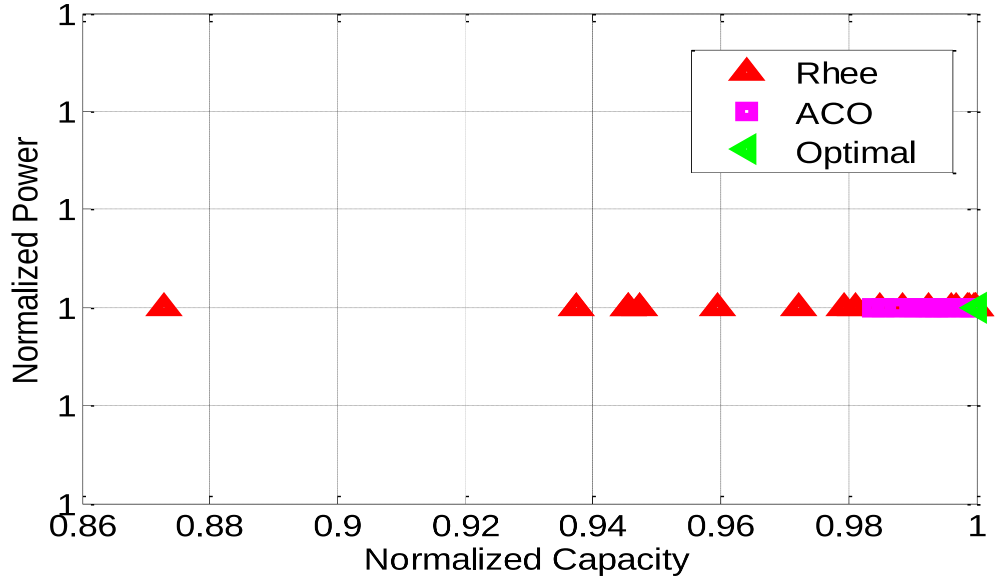

Figure 4.28: Distribution of the Results of Rhee Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Small Case

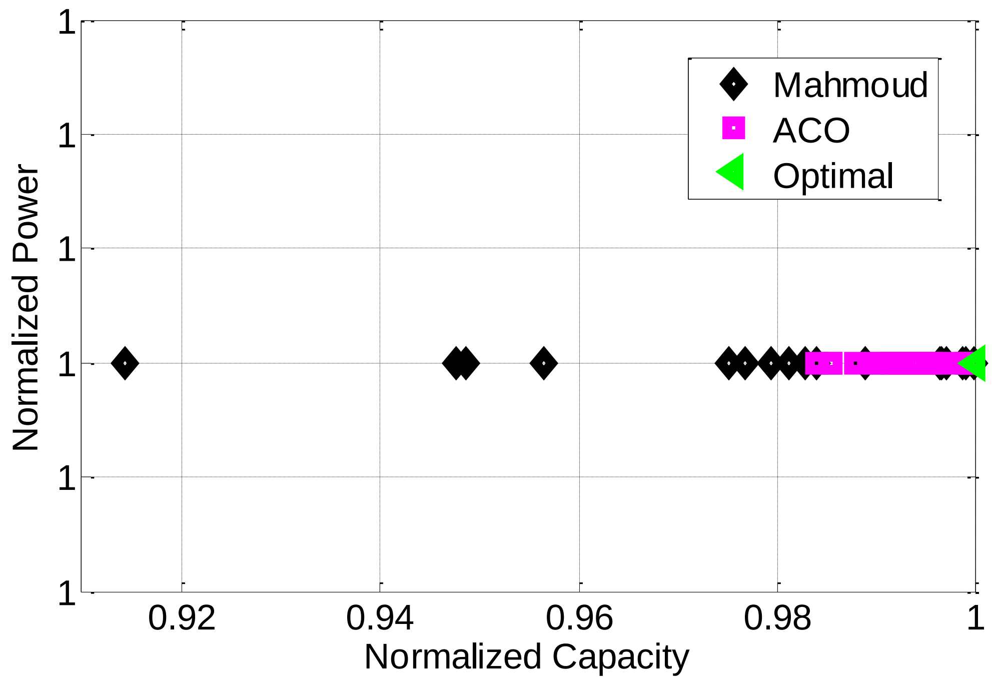

Figure 4.29: Distribution of the Results of Mahmoud Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Small Case

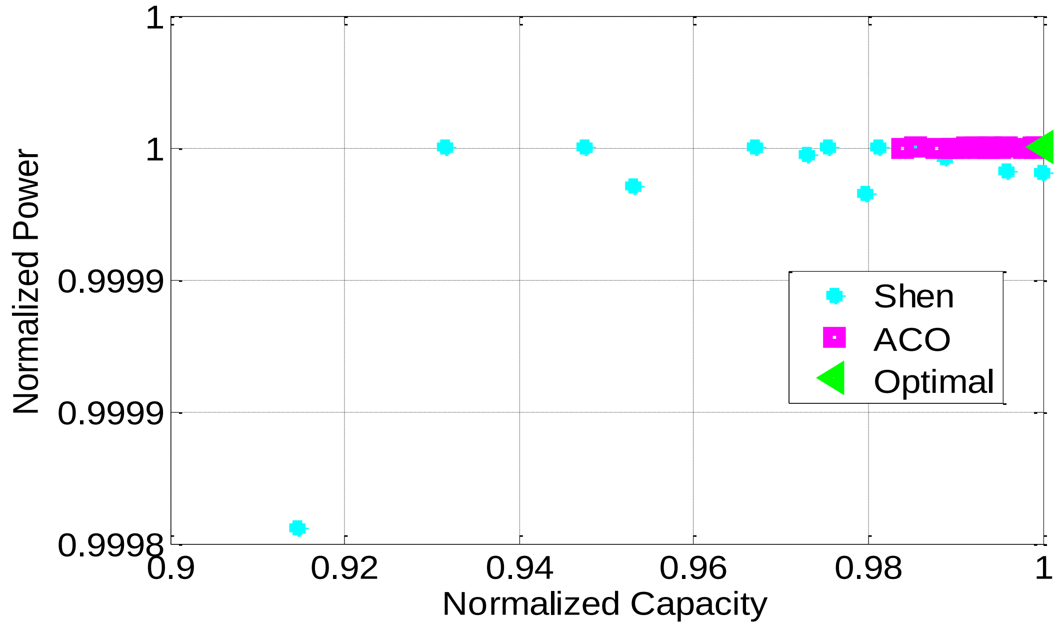

Figure 4.30: Distribution of the Results of Shen Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Small Case

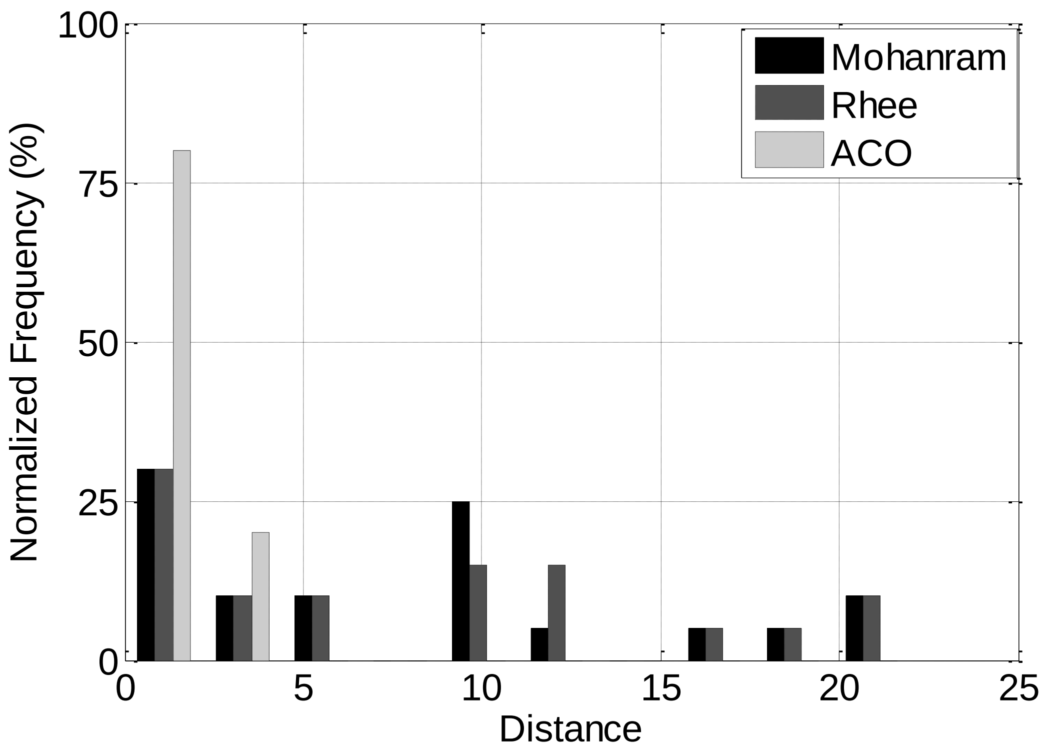

Figure 4.31: Different Allocations from the Optimal Solution for Small Case

The results of the comparison with the optimal solution of the synthesized problem for the large size case of $K = 10$ are shown in Table 4-3 and in Figure 4.32 to Figure 4.36. The results in Table 4-3 show that the ACO-based algorithm has the worst average distance and Mohanram [7] algorithm has the best average distance. At the same time, the allocations of the ACO-based algorithm in Figure 4.36 are more similar to the optimal solution than the allocations of Mohanram [7] and Rhee [1] algorithms. Moreover, some of the solutions of the ACO-based algorithm are better than the solutions of the other algorithms in [1, 5, 7, 8] and some are worst as shown in Figure 4.32 to Figure 4.35. This is due to two reasons. The first one is due to the reorganization of the gain matrix to get the optimal solution. Some users in the original gain matrix don't have sub-channels with the maximum gain values compared to the other users. This will lead to zero proportional rate constants which will prevent the algorithms from work. Therefore, the gain matrix has been reorganized to ensure at least one sub-channel with the maximum gain for each user. This reorganization matches the way of allocations of Rhee [1] and Mohanram [7] algorithm. As a result, Rhee [1] and Mohanram [7] algorithms will select these sub-channels that matches the allocations in the optimal solution and they will have a better solutions. The second reason is the power allocation. Rhee [1] algorithm uses equal power for each allocation. Also, Mohanram [7] algorithm firstly assume equal power for each allocation. Then for each user, it redistributes the total user power on the user's allocated sub-channels using water-filling algorithm. For Shen [8] and Mahmoud [5] algorithms, they use water-filling algorithm to distribute the total power on the final allocation of Rhee [1] algorithm. While in the ACO-based algorithm, water-filling algorithm is used initially to distribute the power on the sub-channels of each user before selection. Therefore, each sub-channel is selected with its power. Then, the power of each user will be redistributed again as in Mohanram [7] algorithm on the user's sub-channels. As a result, the first power allocations waste power on users with weak sub-channels and reduce the available power for the strong sub-channels. This reason plays a significant part in making the solutions of the ACO-based algorithm worst than the other algorithms even when the allocations are more similar to the optimal solution as shown in the results. Therefore, Figure 4.37 shows the results of the algorithms in [1, 5, 7, 8] with the ACO-based algorithm against the maximum available capacity. To evaluate the maximum capacity, we find the assignments with the highest gains across all users and sub-channels where the sub-channels are mutually exclusive. Then, we distribute the system power on the assignments using water-filling algorithm. The simulation is performed for 1000 channel realization and equal PRCs, i.e., $\gamma_{\mathrm{i}} = 1, \forall\ i = 1, 2, \dots, K$. Also, it is performed for $N = 64$ sub-channels and users vary from $K = 2$ to $K = 16$. The other parameters are the same as the parameters for the large size case. The results show that the ACO-based algorithm has the closest results to the maximum. Then Mohanram [7] algorithm followed by Mahmoud [5] algorithm. Rhee [1] algorithm followed by Shen [8] algorithm have the farthest results from the maximum capacity.

Table 4-3: Comparison with Optimal Solution for Large Size System

Algorithm \ Metric	Standard Deviation	Average Distance
Mohanram [7]	0.0107	0.0105
Rhee [1]	0.0110	0.0107
Mahmoud [5]	0.0108	0.0107
Shen [8]	0.0108	0.0107
ACO	0.0058	0.0152

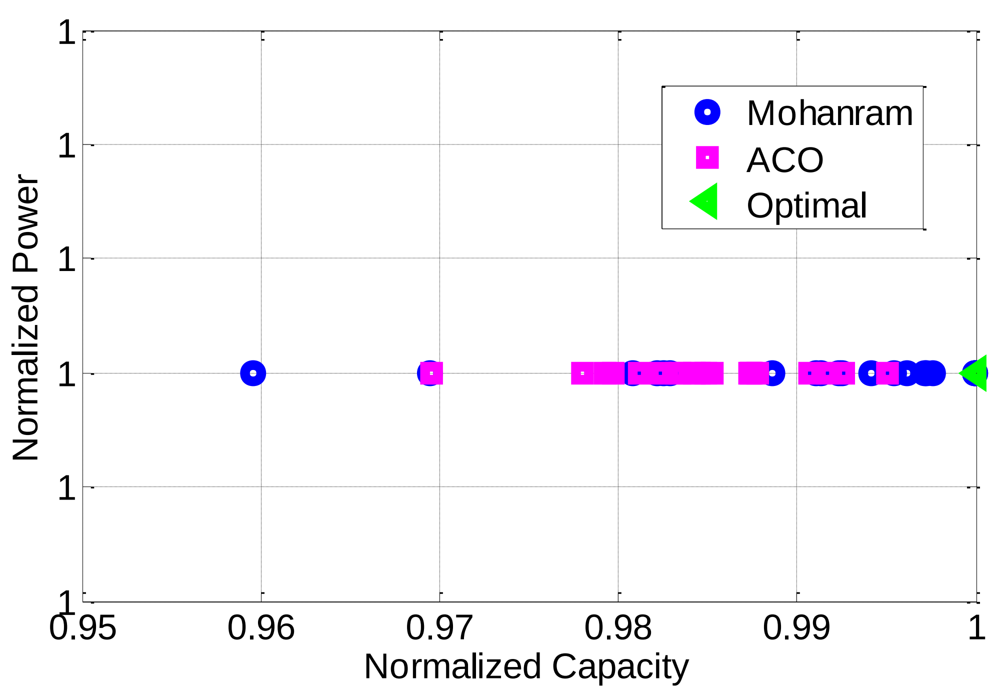

Figure 4.32: Distribution of the Results of Mahanram Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Large Case

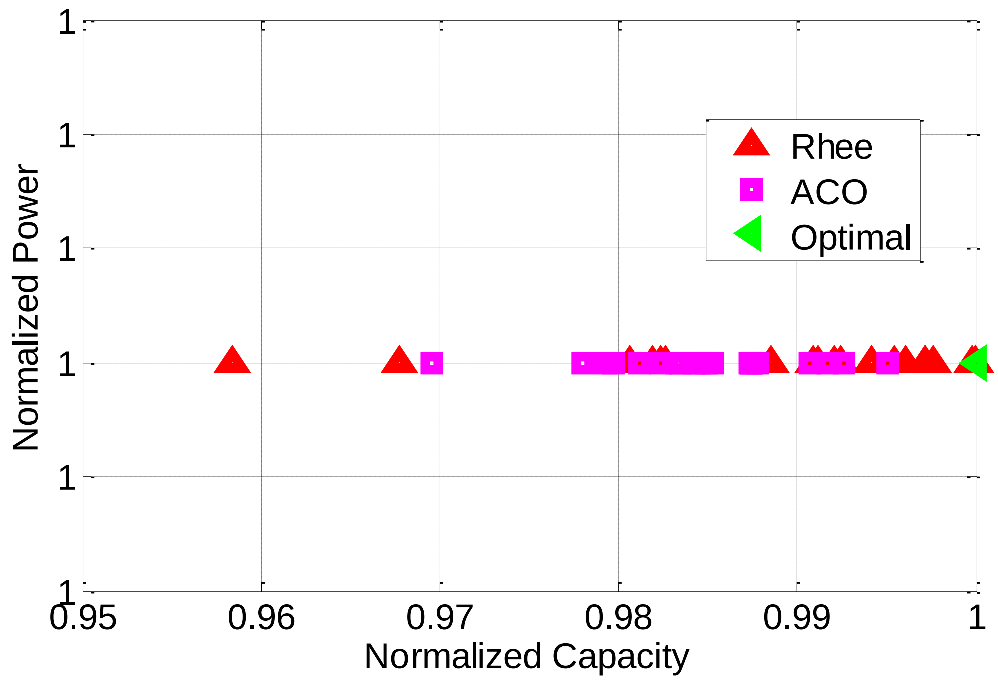

Figure 4.33: Distribution of the Results of Rhee Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Large Case

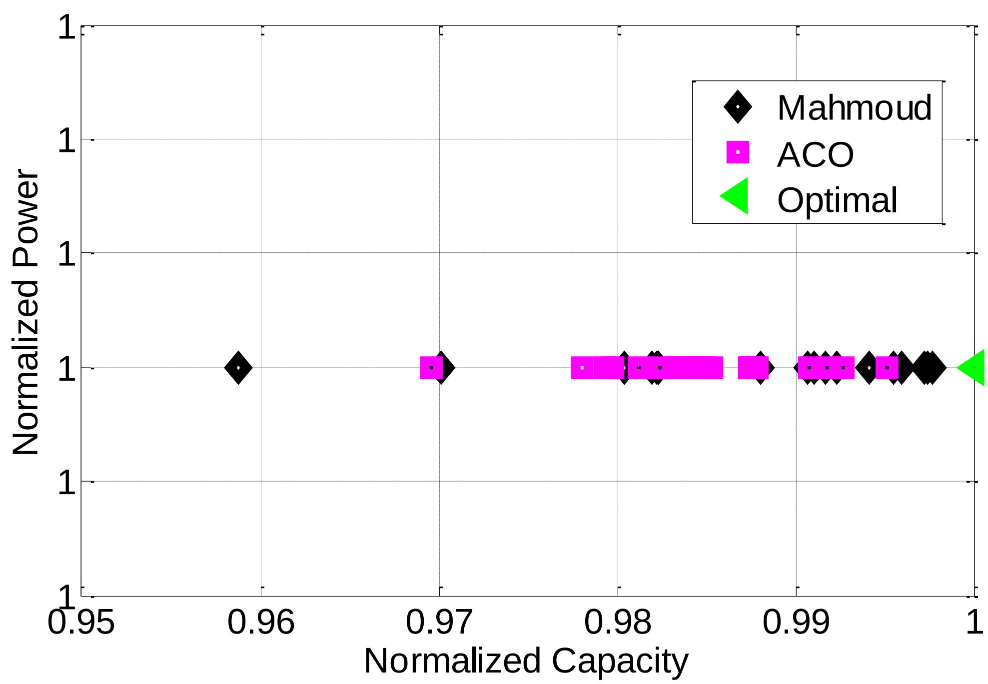

Figure 4.34: Distribution of the Results of Mahmoud Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Large Case

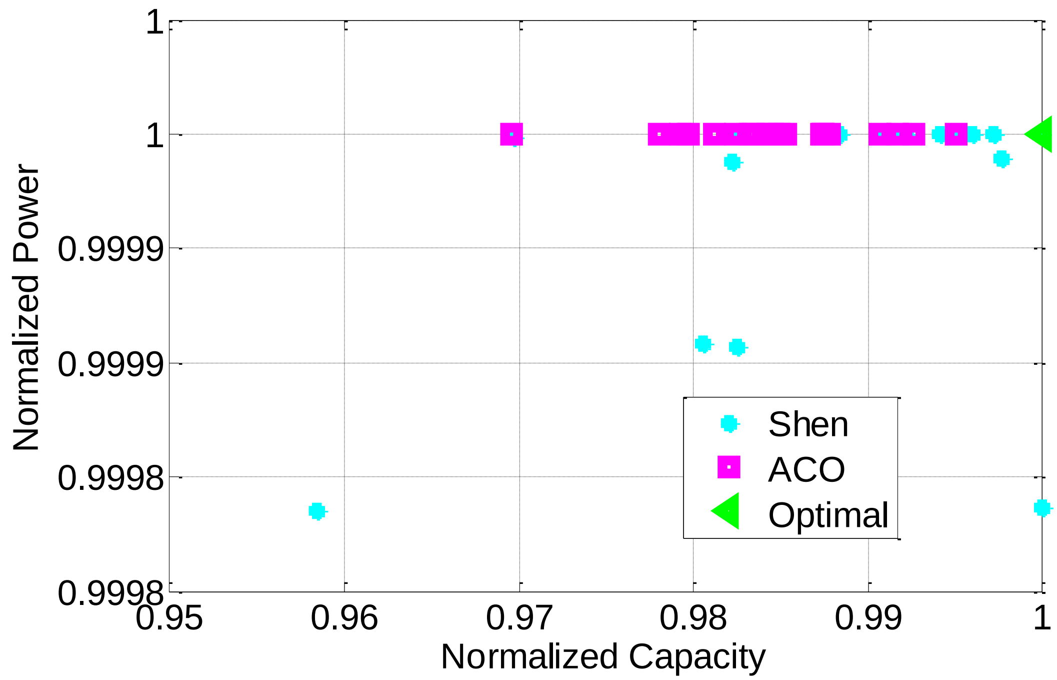

Figure 4.35: Distribution of the Results of Shen Algorithm versus the Results of ACO-based Algorithm and the Optimal Solution for Large Case

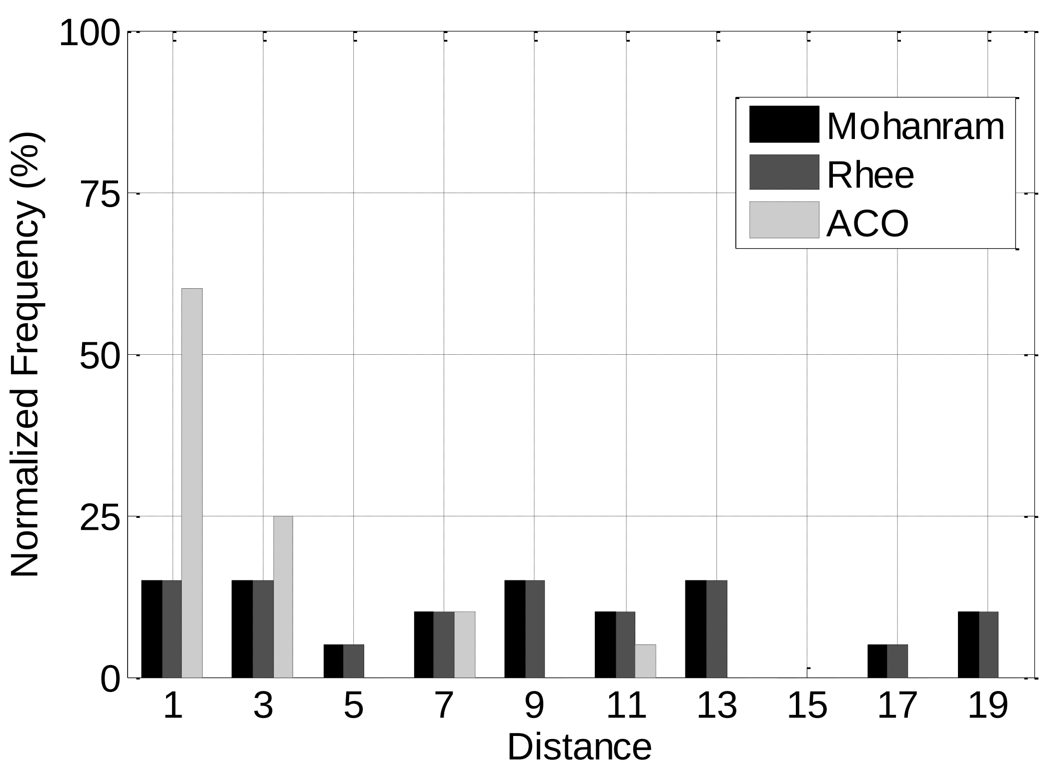

Figure 4.36: Different Allocations from the Optimal Solution for Large Case

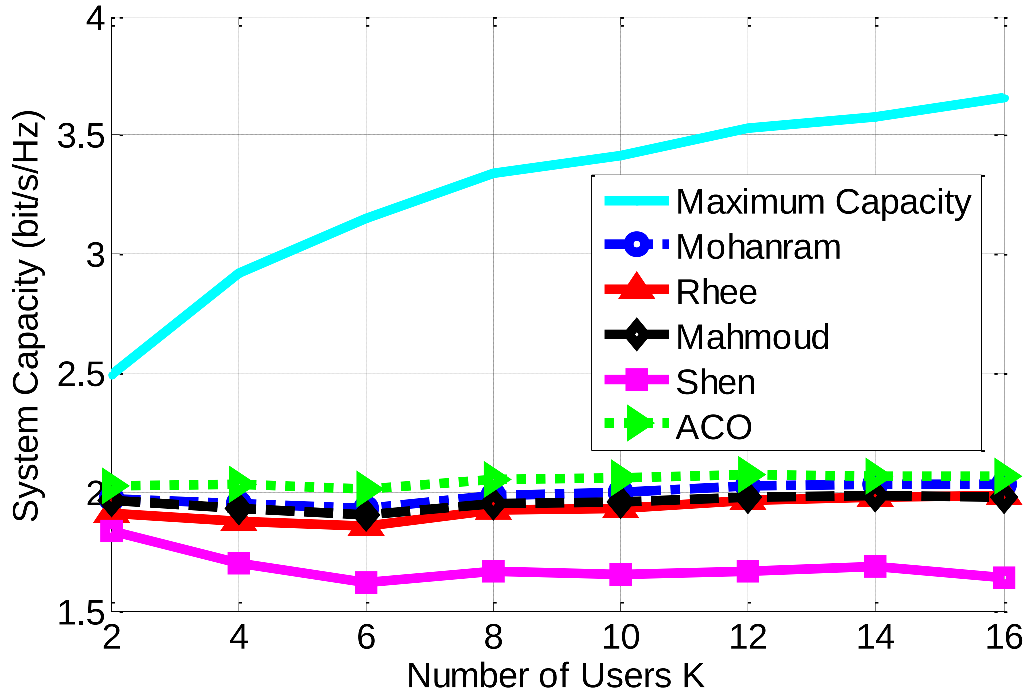

Figure 4.37: Comparison with Maximum Capacity

Chapter 5

CONCLUSION AND FUTURE DIRECTIONS

In this chapter, the thesis work is concluded where the results are summarized. Then, the possible future work is stated.

5.1 CONCLUSIONS

Orthogonal Frequency Division Multiple Access (OFDMA) is the core radio transmission technology for Worldwide Interoperability for Microwave Access (WiMAX) and Long-Term Evolution (LTE) [3]. OFDMA is introduced to solve the problem of Frequency Division Multiple Access (FDMA). OFDMA is able to exploit both multiuser and frequency diversity gains due to its high spectrum and power efficiency. Therefore, huge research is based on OFDMA and OFDM. In this thesis work, the Ant Colony-based Optimization algorithm (ACO-based) is used to solve the resource allocation problem of a multiuser OFDMA system and obtain solutions of acceptable qualities in terms of total system throughput and compliance with the proportional rates constraint, referred to by fairness.

Three versions of the ACO-based algorithm are implemented to achieve the thesis main objective. The first two versions have the best results of total system capacity with very low fairness between users when compared to the results of Rhee [1], Mohanram [7], Shen [8], and Mahmoud [5] algorithms. At the same time, the final implementation of the ACO-based algorithm (version 3) has the best results of total system capacity with high fairness between users.

Moreover, an optimal solution for a synthesized problem is compared the solution of the ACO-based algorithm version 3 in addition to the solutions of the other algorithms in [1, 5, 7, 8]. The ACO-based algorithm has the best results for small number of users. At the same time, the ACO-based algorithm has the worst average distance from the optimal solution for large number of users. In both cases, the sub-channels allocations of the ACO-based algorithm are more similar to the sub-channels allocations of the optimal solution than the other algorithms in [1, 5, 7, 8]. Therefore, the algorithms are compared with the maximum possible capacity that any algorithm can reach. The results show that the ACO-based algorithm is the closest algorithm to the maximum capacity.

5.2 FUTURE DIRECTIONS

This work led to many ideas and many possible future directions. They are as follows:

1. Investigation of how to reduce the execution time of the ACO-based algorithm.
2. Investigation of use of some other bio-inspired algorithms and iterative heuristic algorithms such as fish and bee algorithms.
3. Investigation of how to improve the allocations of other algorithms in the literature.

REFERENCES

[1] W. Rhee and J. M. Cioffi, "Increase in capacity of multiuser OFDM system using dynamic subchannel allocation," in *Proc. IEEE Veh. Tech. Con. (VTC 2000)*, Tokyo, 2000, pp. 1085-1089.

[2] C. Y. Wong, R. S. Cheng, K. B. Lataief, and R. D. Murch, "Multiuser OFDM with adaptive subcarrier, bit, and power allocation," *IEEE J. Select. Areas Commun.*, vol. 17, no.10, pp. 1747-1758, Oct. 1999.

[3] K. H. Teo, Z. Tao, and J. Zhang, "The mobile broadband WiMAX standard [standards in a nutshell]," *IEEE Signal Processing Mag.*, vol. 24, no.5, pp. 144-148, Sept. 2007.

[4] *http://wiki.hsc.com/wiki/Main/OFDM*.

[5] A. S. Mahmoud, A. Y. Al-Rayyah, and T. R. Sheltami, "Adaptive Power Allocation Algorithm to Support Absolute Proportional Rates Constraint for Scalable OFDM Systems," in *Veh. Tech. Conf. (VTC 2010-Spring), 2010 IEEE 71st*, 2010, pp. 1-4.

[6] R. Jian, "The art of computer systems performance analysis: Techniques for experimental design, measurement, simulation, and modeling," vol. 182, ed: John Wiley & Sons New York, 1991.

[7] C. Mohanram and S. Bhashyam, "A sub-optimal joint subcarrier and power allocation algorithm for multiuser OFDM," *IEEE Commun. Lett.*, vol. 9, no.9, pp. 685-687, Aug. 2005.

[8] Z. Shen, J. G. Andrews, and B. L. Evans, "Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints," *IEEE Trans. on Wireless Commun.*, vol. 4, pp. 2726-2737, Nov. 2005.

[9] I. Ahmed and S. P. Majumder, "Adaptive resource allocation based on modified genetic algorithm and particle swarm optimization for multiuser OFDM systems," in *Int. Conf. on Electrical and Computer Engineering, 2008. ICECE'08*, 2008, pp. 211-216.

[10] X. Zhang, W. Ye, S. Feng, and H. Zhuang, "Adaptive Resource Allocation for OFDMA System Based on Ant Colony Algorithm," in *1st Int. Conf. on Information Science and Engineering (ICISE'09)*, 2009, pp. 2526-2529.

[11] H. Ahmadi and Y. H. Chew, "Subcarrier-And-Bit Allocation in Multiclass Multiuser Single-Cell OFDMA Systems Using an Ant Colony Optimization Based Evolutionary Algorithm," in *IEEE Wireless Communications and Networking Conf. (WCNC)*, 2010, pp. 1-5.

[12] H. Ahmadi and Y. H. Chew, "Adaptive subcarrier-and-bit allocation in multiclass multiuser OFDM systems using genetic algorithm," in *2009 IEEE 20th Int. Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC'09)*, 2009, pp. 1883-1887.

[13] Y. B. Reddy, N. Gajendar, P. Taylor, and D. Madden, "Computationally Efficient Resource Allocation in OFDM Systems: Genetic Algorithm Approach," in *4th Int. Conf. on Information Technology (ITNG'07)*, 2007, pp. 36-41.

[14] T. Meiqin, L. Xiaohua, and X. Yalin, "Nonconvex optimal resource allocation for multiuser OFDM systems," in *2010 29th Chinese Control Conf. (CCC)*, 2010, pp. 4128-4131.

[15] Z. Tao and Z. Rui, "A novel Bit and Power allocation scheme for OFDM system over the low voltage power line," in *6th Int. Conf. in Natural Computation (ICNC 2010)*, 2010, pp. 2540-2544.

[16] P. Vasileios D, F. Ioannis G, and K. Stavros A, "A Novel Fairness-Aware Resource Allocation Scheme in Multiuser SISO-OFDMA Downlink," *Int. J. of Veh. Tech.*, vol. 2010, pp. 1-10, June 2010.

[17] Z. Liang, Y. H. Chew, and C. C. Ko, "A linear programming solution to the subcarrier-and-bit allocation of multiclass multiuser OFDM systems," in *IEEE 65th Veh. Tech. Conf. (VTC2007-Spring)*, 2007, pp. 2682-2686.

[18] I. Kim, H. L. Lee, B. Kim, and Y. H. Lee, "On the use of linear programming for dynamic subchannel and bit allocation in multiuser OFDM," in *IEEE Global Telecommun. Conf. (GLOBECOM '01)*, 2001, pp. 3648-3652.

[19] I. Kim, I. S. Park, and Y. H. Lee, "Use of linear programming for dynamic subcarrier and bit allocation in multiuser OFDM," *IEEE Trans. on Veh. Tech.*, vol. 55, no. 4, pp. 1195-1207, July 2006.

[20] B. Liu, M. Jiang, and D. Yuan, "Adaptive Resource Allocation in Multiuser OFDM System Based on Genetic Algorithm," in *WRI Int. Conf. on Communications and Mobile Computing (CMC'09)*, 2009, pp. 270-273.

[21] I. Chatzifotis, K. Tsagkaris, and P. Demestichas, "Ant Colony Optimization for Subcarrier Allocation in OFDMA-based Wireless Systems," in *ICT-MobileSummit 2009 Conf. Proc.*, 2009, pp. 1-9.

[22] Z. Huang, S. Feng, W. Ye, and H. Zhuang, "Ant-colony-based resource allocation in OFDMA MESH network," in *5th Int. Conf. on Wireless Commun., Networking and Mobile Computing (WiCom'09)*, 2009, pp. 1-5.

[23] Y. B. Reddy and V. V. Phoha, "Genetic algorithm approach for resource allocation in multi-user OFDM systems," in *2nd Int. Conf. on Communication Systems Software and Middleware (COMSWARE'07)*, 2007, pp. 1-6.

[24] N. Sharma and K. Anupama, "A Novel Genetic Algorithm for Adaptive Resource Allocation in MIMO-OFDM Systems with Proportional Rate Constraint," *Wireless Pers. Commun.*, vol. 61, no. 1, pp. 113-128, 2011.

[25] N. Sharma, A. K. Tarcar, V. A. Thomas, and K. Anupama, "On the use of particle swarm optimization for adaptive resource allocation in orthogonal frequency division multiple access systems with proportional rate constraints," *Inform. Sci.*, vol. 182, no. 1, pp. 115-124, Jan. 2012.

[26] Y. Yu and W. Zhou, "Resource allocation for OFDMA system based on genetic algorithm," in *Int. Workshop on Cross Layer Design (IWCLD'07)*, 2007, pp. 65-69.

[27] P. Papazoglou, D. Karras, and R. Papademetriou, "On the implementation of ant colony optimization scheme for improved channel allocation in wireless communications," in *4th Int. IEEE Conf. Intelligent Systems (IS '08)*, 2008, pp. 6-41-6-50.

[28] N. Sharma, A. Rao, A. Dewan, and M. Safdari, "Rate adaptive resource allocation for multiuser OFDM using NSGA-II," in *4th Int. Conf. on Wireless Communication and Sensor Networks (WCSN'08)*, 2008, pp. 161-166.

[29] G. Di Caro and M. Dorigo, "Ant Colony Optimization and its application to adaptive routing in telecommunication networks," Unpublished doctoral dissertation, Citeseer, 2004.

[30] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization: Artificial Ants as a Computational Intelligence Technique," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28-39, 2006.

[31] H. M. Botee and E. Bonabeau, "Evolving ant colony optimization," *Adv. Complex Systems*, vol. 1, no. 2, pp. 149-159, 1998.

[32] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. on Evolutionary Computation*, vol. 1, no. 1, pp. 53-66, Apr. 1997.

[33] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 1, pp. 29-41, Feb. 1996.

[34] A. Colorni, M. Dorigo, and V. Maniezzo, "An investigation of some properties of an ant algorithm," in *Proc. Parallel Problem Solving from Nature Conf. (PPSN 92)*, Brussels, Elsevier, 1992, pp. 509-520.

[35] A. Colorni, M. Dorigo, and V. Maniezzo, "Distributed optimization by ant colonies," in *Proc. First Europ. Conf. ArtiJicial Life*, F. Varela and P. Bourgine, Eds. Paris, France: Elsevier, 1991, pp. 134-142.

[36] R. Schoonderwoerd, O. E. Holland, J. L. Brutten, and L. J. M. Rothkrantz, "Ant-based load balancing in telecommunication networks," *Adaptive Behaviour*, vol. 5, no. 2, pp. 169-207, 1997.

[37] G. Di Caro and M. Dorigo, "Mobile agents for adaptive routing," in *Proc. of the 31st Hawaii Int. Conf. on Systems Sciences (HICSS-31)*, Kohala Coast, Hawaii, 1998, pp. 74-83.

[38] G. Di Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," *J. Artif. Int. Res.*, vol. 9, pp. 317-365, 1998.

[39] G. N. Varela and M. C. Sinclair, "Ant colony optimisation for virtual-wavelength-path routing and wavelength allocation," in *Proc. of the 1999 Congr. on Evolutionary Computation (CEC 99)*, Washington, DC, 1999, pp. 1809–1816.

[40] K. Socha and M. Dorigo, "Ant colony optimization for continuous domains," *Eur. J. Oper. Res.*, vol. 185, no.3, pp. 1155-1173, 2008.

[41] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic," in *Proc. Congr. Evol. Comput. (CEC'99)*, 1999, pp. 1470–1477.

[42] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28-39, Nov. 2006.

[43] C. Blum, "Ant colony optimization: Introduction and recent trends," *Physics of Life reviews*, vol. 2, no. 4, pp. 353-373, 2005.

[44] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theoretical computer science*, vol. 344, no. 2-3, pp. 243-278, Nov. 2005.

[45] M. Dorigo and T. Stützle, "The ant colony optimization metaheuristic: Algorithms, applications, and advances," *Handbook of metaheuristics,* vol. 57, pp. 250-285, 2003.

[46] C. Blum, "Beam-ACO--hybridizing ant colony optimization with beam search: an application to open shop scheduling," *Computers & Operations Research,* vol. 32, no. 6, pp. 1565-1591, 2005.

[47] R. S. Parpinelli, H. S. Lopes, and A. A. Freitas, "Data mining with an ant colony optimization algorithm," *IEEE Trans. Evol. Comput.,* vol. 6, no. 4, pp. 321-332, 2002.

[48] B. Liu, H. Abbas, and B. McKay, "Classification rule discovery with ant colony optimization," 2004, pp. 83-88.

[49] J. Gottlieb, M. Puchta, and C. Solnon, "A study of greedy, local search, and ant colony optimization approaches for car sequencing problems," *Applications of evolutionary computing,* vol. 2611, pp. 245-282, 2003.

[50] P. R. McMullen, "An ant colony optimization approach to addressing a JIT sequencing problem with multiple objectives," *Artificial Intelligence in Engineering,* vol. 15, no. 3, pp. 309-317, July 2001.

[51] J. Levine and F. Ducatelle, "Ant colony optimization and local search for bin packing and cutting stock problems," *J. Oper. Res. Soc.,* vol. 55, no. 7, pp. 705-716, 2004.

[52] D. Merkle, M. Middendorf, and H. Schmeck, "Ant colony optimization for resource-constrained project scheduling," *IEEE Trans. Evol. Comput.,* vol. 6, no. 4, pp. 333-346, 2002.

[53] M. Den Besten, T. Stützle, and M. Dorigo, "Ant colony optimization for the total weighted tardiness problem," in *Lecture Notes in Computer Science*, Berlin, Germany: Springer-Verlag, 2000, pp. 611-620.

[54] M. Gravel, W. L. Price, and C. Gagne, "Scheduling continuous casting of aluminum using a multiple objective ant colony optimization metaheuristic," *European Journal of Operational Research,* vol. 143, no. 1, pp. 218-229, 2002.

[55] J. E. Bell and P. R. McMullen, "Ant colony optimization techniques for the vehicle routing problem," *Advanced Engineering Informatics,* vol. 18, no. 1, pp. 41-48, 2004.

[56] B. Latni, *Signal processing and linear systems*. Carmichael, CA: Berkeley-Cambridge Press, 1998.

[57] B. A. B. Sarif, "Modified ant colony algorithm for combinational logic circuits design," Master’s thesis, King Fahd University of Petroleum and Minerals, Dhahran, KSA, 2003.

[58] S. Goss, S. Aron, J. L. Deneubourg, and J. M. Pasteels, "Self-organized shortcuts in the Argentine ant," *Naturwissenschaften,* vol. 76, no. 12, pp. 579-581, 1989.

VITA

Name: MOHAMMAD KHALAF

Place of Birth: Jerusalem, Palestine.

Nationality: Jordanian

Permanent Address: Al-Asfahani St.,

Jerusalem,

Palestine.

Telephone: +970-599545313, +970-22961645

Email Address: mohkhalaf@gmail.com

Educational Qualification:

M.S (Computer Networks)

Jan 2012

King Fahd University of Petroleum and Minerals

Dhahran, Saudi Arabia.

B. Tech. (Computer Engineering)

June 2008

An-Najah National University, Palestine.