December 2019

Design and Evaluation of Flow Mapping Systems for Heterogeneous Wireless Networks

Jianwei Liu
Clemson University, ljw725@gmail.com

Design and evaluation of flow mapping systems for heterogeneous wireless networks

---

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Computer Science

---

by
Jianwei Liu
December 2019

Accepted by:
Dr. Jim Martin, Committee Chair
Dr. Brian Dean
Dr. Kuang-Ching Wang
Dr. James Westall

Abstract

Mobile wireless networks are always challenged by growing application demand. The increasing heterogeneity of both mobile device connection capability and wireless network coverage forms a general heterogeneous wireless network (HetNet). This type of HetNet contains sub-networks of different Radio Access Technologies. How to better coordinate the mappings of flows between Access Points (AP) and User Equipment (UE) inside this type of HetNet to improve system and user-level performance is an interesting research problem. The flow mapping systems used by off-the-shelf mobile devices make policy-based decisions from local information. Several global information based flow mapping systems that use Generalized Proportional Fairness (GPF) as the optimization objectives have been proposed to improve the system-level performance. However, they have not been compared with both the local-policy based approaches and the optimal solution under the same assumptions with variations of system parameters. Therefore, it is still unclear to the community whether it is worthwhile to construct a flow mapping system for HetNets composed by LTE and WiFi networks, even under a simplified assumption of only optimizing throughput related system performance metrics. In this dissertation, we evaluate three types of flow mapping systems: Global Information based Flow Mapping Systems (GIFMS), Local Information based Flow Mapping Systems (LIFMS), and Semi-GIFMS. We evaluate these systems with metrics related to both the spectrum efficiency and flow-level fairness under the following variations of system parameters: 1) topologies of UEs; 2) coverage of APs; 3) number of UEs; 4) number of non-participating UEs; 5) *on-off* session dynamics; 6) UE mobility. We also discuss options to implement each type of flow mapping systems and any relevant trade-offs.

From the evaluations, we find that the currently-in-use WiFi preferred local greedy flow mapping system provides far poorer spectral efficiency and generalized proportional fairness than all the other tested flow mapping systems, including the local greedy flow mapping systems that

give LTE and WiFi equal opportunities (local-greedy-equal-chance) in most settings. This finding indicates that the flow mapping system in use has much room for improvement in terms of GPF and aggregate throughput. The performance of local-greedy-equal-chance is close to that of the global and AP-level information based systems under some UE topologies. However, their performance is not as consistent as the global and AP-level based systems when UEs form clusters that produce AP load imbalance.

We also derive the incremental evaluations of GPF for both proportional and max-min fair scheduled APs. Based on these derivations, we propose a design for AP-level information based flow mapping system or Semi-GIFMS. It is an event-triggered flow mapping system based on minimum AP-level metrics monitoring and dissemination. From our evaluation and analysis, this flow mapping system performs equivalent to or better than GIFMS in terms of both GPF and aggregate throughput in all the tested scenarios. It also owns the advantages of lower overhead and not requiring an additional scheduling server. We think it is the best choice for the next generation HetNets where APs can be modified to monitor and broadcast the minimum information identified.

Furthermore, we find that the number of UEs, number of non-participating UEs, coverage of APs, bandwidth sharing types of APs, *on-off* session and UE mobility dynamics do not have a major impact on the relative performance difference among various flow mapping systems.

Dedication

To my parents, my wife Hui, and my son Aaron.

Acknowledgments

First, I would like to thank my advisor, Dr. Jim Martin, for providing me the opportunity and funding to conduct this research and all his support in the middle.

Many thanks to Dr. Westall, who led this research and taught me how to identify, explore, solve, and present a research problem in a better way.

I also would like to thank other committee members Dr. Brian Dean and Dr. Kuang-Ching Wang for the valuable advice during the proposal phase of the research and feedback in the middle. Thanks to Manveen Kaur for helping to proofread some parts of the manuscript.

I am grateful to my parents, who have provided me with financial and emotional support to finish this long journey.

Finally, I would like to thank my wife Hui Chang. Without her encouragement and accompany, I might have given up several years ago.

Thanks to all the people who helped me!

Table of Contents

List of Tables

List of Figures

Glossary

- 3GPP - 3rd Generation Partnership Project
- AP - Access Point
- BS - Cellular (3GPP) Base Station
- CDF - Cumulative Distribution Function
- GIFMS - Global Information based Flow Mapping System
- GPF - Generalized Proportional Fairness
- HetNet - Heterogeneous Network
- LIFMS - Local Information based Flow Mapping System
- LTE - Long Term Evolution
- MCS - Modulation and Coding Scheme
- PF - Proportional Fairness
- RAT - Radio Access Technology
- S-GIFMS - Semi-Global Information based Flow Mapping System
- SINR - Signal to Interference and Noise Ratio
- SNR - Signal to Noise Ratio
- UE - User Equipment
- WiFi - Wireless Fidelity

Chapter 1

Introduction

1.1 Research Motivation

The scarcity of wireless spectrum and the continuous growth of mobile data always challenge the players of modern Internet. For network operators (such as Verizon and AT&T), Content Delivery Network (CDN) providers (such as Akamai), and application providers (such as Youtube and Netflix), how to achieve better system performance using the same amount of resources, and therefore higher user satisfaction is always an operational objective.

Mobile devices now have both heterogeneous connection capabilities and heterogeneous network coverages. An example of heterogeneous connection capability is the fact that most mobile phones have both the LTE and WiFi interfaces. Heterogeneous network coverages can have different meanings in different research contexts. However, the essence of it is always a network that contains potentially overlapping sub-networks with certain differences. For example, it can be the heterogeneous coverage formed by overlapping LTE base stations (BS) with different power levels. This heterogeneity can also come from the ownership of sub-networks, e.g. multiple overlapping LTE base stations with the same power level but belonging to different operators. It may also be overlapping LTE and WiFi Access Point (AP) with the same/different operator(s), which vary in Radio Access Technology (RAT) used (In this dissertation, we will use the term AP as a generalized representation for both BS and AP.). These increases in the heterogeneity of connections and coverages create an opportunity to improve system performance by more carefully coordinating how flows connect to APs.

The term *flow* here refers to an application flow on mobile devices or user equipments (UE) in general (even though with the simplifying assumption in this dissertation of only one application flow per mobile device, it is equivalent to a device level flow). *Flow mapping* refers to deciding which flow should bound to which interface and which AP. We call this decision a mapping or an association plan. Because of the heterogeneous connection capability of each UE and their diverse connection status, an "appropriate" mapping considering the user diversity can potentially boost the resource utilization rate and/or fairness among mobile devices. A *flow mapping system* is a system that can produce a flow mapping. It usually consists of a mapping algorithm and the required protocol for information collection or association plan enforcement.

The flow mapping systems currently in use and those proposed in the previous literature can be classified into three categories based on the scope of the information they rely on when making decisions.

The default flow mapping system currently used by mobile devices is based on local policies. For example, most Android devices use the policy of " connecting to WiFi APs whenever they are available" [17]. IOS devices use a similar policy while further considering the user preference [5, 3]. This type of flow mapping systems only rely on local information. We call it an LTEocal Information based Flow Mapping System (LIFMS). On the other hand, the flow mapping systems proposed in previous research such as [13, 33] require global information about the connection status between all the APs and UEs in the system. Therefore, we call this kind of mapping system a Global Information based Flow Mapping System (GIFMS). Flow mapping systems such as the ones in [43] and [16] can be considered as a type of systems in the middle, which only requires AP-level scheduling information. We call such systems a Semi-Global Information based Flow Mapping System (S-GIFMS) or an AP-Assisted Flow Mapping System (AAFMS).

Even though recent literature has indicated the policy based approach currently in use is not ideal [13, 17], and suggested various global/AP level information based methods [13, 43, 33, 16] to better map the flows, the performance of these methods are not compared under the same assumptions and settings. Therefore, it is difficult to answer the question of "whether it is worthwhile to build a flow mapping system using global or AP-level information for HetNets". We attempt to answer this question in this dissertation.

The term *heterogeneous network* (HetNet), as the name indicates, refers to the combined and coordinated usage of networks that are different in various layers or characteristics. Early work

on HetNets is limited to the coordinated usage of macrocell and small cells in cellular networks. The work in [13] has identified, analyzed and proposed solutions for the flow mapping systems for this type of HetNet using Generalized Proportional Fairness (GPF) as the optimization objective. GPF is a natural extension of the Proportional Fairness from one AP to multiple APs. We show its mathematical form in Eq. (3.1). From the results [13] showed, we see that using GPF as the primary objective has increased both the spectrum efficiency and fairness among users under the scenarios tested. However, there are several limitations to that work. First, the main algorithm they propose is highly dependent on the assumption of cellular APs using proportional fairness schedulers. Because of that, it cannot be extended to a more general HetNet with WiFi networks involved. The work in [43] aims at the same GPF objective as in [13]. However, they provide a hybrid solution by forming the problem as a convex optimization problem and use Lagrangian dual decomposition to break it into two subproblems. The two subproblems can then be solved at UEs and APs respectively.

ATOM [33] and MOTA [16] proposed solutions for flow mapping systems in a more general HetNet which involves both LTE and WiFi networks. ATOM proposes a centralized heuristic that offloads flows from LTE to WiFi APs greedily based on GPF changes. The offloading is batched by sets of UEs under the same APs. MOTA works on a similar problem, but with the addition of flow costs. Authors working on MOTA assume APs can be modified to broadcast information related to the loads similar to [43] in the Single-RAT version of the problem. They designed greedy algorithms that can be executed on UEs locally based on the flow weights and the received load factors.

The previous work above have the following in common,

1) They use GPF as their optimization objectives (Note that the objective in MOTA is equivalent to optimizing GPF if the prices are proportional to the logarithm of throughput and equal for all flows).

2) They assume backlogged traffic. With these the above two assumptions, the objective of GPF can be expressed as optimizing the sum of the logarithms of user throughputs [26].

1.2 Research Objective and Problem Formulation

This research has two objectives:

1. To evaluate the performance of some representative flow mapping algorithms in all three

types of flow mapping systems with metrics that can capture both the spectrum efficiency and fairness among flows under various scenarios and system parameters. This evaluation helps us in answering the question of whether a GIFMS or S-GIFMS is worth constructing from the perspective of throughput related system performance.

2. To discuss the implementation options of the various flow mapping systems and their trade-offs.

However, with the wide range of HetNet types and possible system parameters, we further make the following assumptions to focus this research.

- The system to be optimized is limited to the scope of a small area, e.g. the size of an LTETE macrocell. The service providers use a divide and conquer approach as in [13] to achieve horizontal scalability.

- We assume that the system only has downstream traffic.

- We assume that all the flows are elastic.

- We assume that only two types of radio access technologies are operational in the system, LTE and WiFi. Each mobile device has one LTE interface and one WiFi interface.

- We do not model the cost of data flows, as the same cost can have different meanings for various users.

- We assume that only one interface can be used at a time.

Under these assumptions, we evaluate some representative mapping systems from both the class of Global Information based Flow Mapping System and Local Information based Flow Mapping System, and compare them with the optimal GPF solution.

1.3 Summary of Methodology

- We test with the static and dynamic simulations. In the static simulation, we assume each flow is always-on. We run the simulation with multiple runs, and re-initialize the location of UEs at the beginning of each run. In the dynamic simulation, we add the dynamics from both the *on-off* sessions and the UEs' leaving-and-joining behaviors.

- We evaluate mapping algorithms with metrics including the GPF value, aggregate throughput, and several other system-level aggregate metrics. Flow-level throughput statistics are also added for reference. We show the definitions of these metrics and what they represent in Chapter 3.

- Our network models are based on LTE and WiFi standards. We use fading models to generate Signal Noise Ratio (SNR) from distance, and then map SNR to Modulation and Coding Schemes (MCS) in the standards.

- We assume LTE uses proportionally fair packet scheduling and WiFi uses throughput fair packet scheduling. We use the models in the literature to model the UE resource contention under these two types of APs.

- We evaluate the following representative mapping algorithms for GIFMS and LIFMS.

  - GIFMS: {ATOM, global-greedy}
  - LIFMS: {local-greedy-equal-chance, local-greedy-wifi-preferred, random-assignment}
  - S-GIFMS: {load-aware-local-greedy}

  We discuss details of these algorithms in Section 5.3.

- We also discuss the possible implementations for both mapping systems in LIFMS and GIFMS.

1.4 Summary of Results and Contributions

From the evaluation results, we find that,

1. The currently-used WiFi-preferred local policy based mapping system has much room for improvement.

2. Local information based mapping systems can work very well under certain UE placements. However, with UE placements introducing load imbalance among APs, the advantages of the global/AP level information based mapping systems over the local information based ones are noticeable.

3. Both increasing the coverage radius of APs and changing WiFi AP to PF scheduled can increase the magnitude of results. However, the relative differences and rankings of algorithms remain the same.

4. The AP level information based flow mapping system can achieve system performance close to that from global information based systems while having less deployment cost.

5. Non-participants can potentially degrade system performance. However, the degradation is almost linear. This linear degradation means that a low ratio of participants can still benefit the system proportionally. The technique of throughput correction for participants can only help to improve the system performance in some cases.

6. The tested *on-off* and user leaving and joining dynamics do not have a significant impact on system performance.

The main contributions of this dissertation include,

1. We have derived and applied network models for an LTE-WiFi HetNet considering the protocol overhead and the discrete nature of modulation and coding schemes, which owns higher fidelity compared to the approach used in the previous literature.

2. We have evaluated the performance of several representative LIFMS, GIFMS, and S-GIFMS in terms of various metrics related to both spectrum efficiency and fairness among flows. The evaluations were conducted both with the comparison of the optimal solution on a smaller scale and without the optimal solution in a larger scale. We have evaluated the flow mapping systems using both static simulations and dynamic simulations with various system parameters such as AP power levels and ratio of non-participants in the system. As far as we know, it is the first time all the representative algorithms in the three types of flow mapping systems have been systematically evaluated and compared. The results provide useful insights for various service providers to reconsider their options when trying to optimize the flow mappings under various scenarios.

3. We have discussed several implementation options for the three types of flow mapping systems. The system designs and trade-off discussions can help service providers to further think about the trade-off between the performance we have shown and the cost to implement and deploy the systems.

4. We have derived the incremental GPF evaluations for two types of APs, i.e. PF and max-min scheduled APs, and proposed an implementation option for a Semi-GIFMS based on the derivation. With this proposal, Semi-GIFMS can be implemented with a simple extension to the AP metrics monitoring and association protocol, and triggered by *off-on* transition events. This implementation will require no additional scheduling server(s), no handover overhead, and low control overhead compared to GIFMS. The change to the association protocol only requires the addition of one/two field(s) to beacon frames while the metrics which need to be monitored are easily measurable for both types of APs. From simulations, we also show that, in the scenarios considered, this Semi-GIFMS can provide a performance that approximates or exceeds that of the GIFMS solutions which requires global network knowledge and impose disruptive reassociations. This result provides important guidance for the minimum information that should be monitored and broadcasted for the component APs in the next generation HetNets.

5. We have verified the evaluation results from two indecently constructed simulators in MATLAB and C, and provided the code as open-source. This provides tools and a baseline for further study of this problem.

1.5 Dissertation Outline

This dissertation is organized as follows. We first introduce background knowledge and review the previous literature in Chapter 2. We then describe the performance metrics we use in the evaluations in Chapter 3, and the network model we use in Chapter 4. In Chapter 5, we provide details about our evaluation methods. We show and analyze the static simulation results in Chapter 6. Then, we show the results for the impact of system dynamics in Chapter 7. In Chapter 8 we discuss possible implementations to the mapping systems and their trade-offs. Finally, we make our conclusions in Chapter 9.

Chapter 2

Background and Related Work

2.1 Basic Concepts of Fairness Metrics

In this dissertation, we use variable names with a hat symbol to denote a vector, e.g. $\hat{x} = (x_1, ..., x_n)$. x_i can be considered as the throughput/rate of UE_i. We call $\hat{x}$ a throughput/rate vector.

2.1.1 Max-Min Fairness

Max-min fairness is said to be achieved by a rate vector if and only if the rate vector is feasible and an attempt to increase the allocation of any participant necessarily results in the decrease in the allocation of some other participant with an equal or smaller allocation. Or, we can define it more mathematically as follows,

A rate vector $\hat{x}$ is max-min fair if and only if, for any other rate vector $\hat{y}$ that is feasible, the following is true: if $y_s > x_s$ for some $s \in I$, then there exists a $t \in I$ such that $x_t \leq x_s$ and $y_t < x_t$, where I is the set of resource contention participants. In the context of flow mapping systems in this dissertation, the resource contention participants are the flows to be scheduled in the system.

With the assumption of elastic traffic and equal-weighted flows, max-min fairness implies all flows have the same throughput. Clearly, this has a significant negative impact on spectrum efficiency because it limits the throughput of every UE to the one with the poorest connection.

2.1.2 Proportional Fairness

There are two equivalent definitions to proportional fairness (PF).

Definition 1: Rate vector $\hat{x}$ is proportionally fair if for any other rate vector $\hat{y}$, $\sum_i \frac{y_i - x_i}{x_i} \leq 0$.

Definition 2: The proportionally fair rate vector $\hat{x}$ maximizes $\sum_i log(x_i)$, i.e. maximizing the sum of the logarithm utility of each component in the rate vector.

The two definitions are equivalent. We provide a brief proof of it as follows,

Let $\hat{x}$ be the optimal rate vector and $\hat{y}$ another vector. Let $y_i = x_i + \Delta_i$.

$$\begin{aligned}\sum_i log(y_i) &= \sum_i log(x_i + \Delta_i) \\ &= \sum_i log(x_i) + \sum_i \frac{\Delta_i}{x_i} + o(\Delta^2) \\ &\approx \sum_i log(x_i) + \sum_i \frac{\Delta_i}{x_i}\end{aligned}$$

Since $\hat{x_i}$ is the optimal solution, we know that,

$$\sum_i \frac{\Delta_i}{x_i} \leq 0 \Leftrightarrow \sum_i \frac{y_i - x_i}{x_i} \leq 0$$

Therefore, the two definitions are equivalent.

Proportional fairness is widely adopted as a scheduling principle for resource allocation at a single router or access point, such as a cellular base station [11]. It achieves a compromise between fairness and spectral efficiency, and provides higher spectrum efficiency compared to max-min fairness [12].

2.2 Types of HetNets

As discussed in the introduction, HetNet is a broad concept because of the wide range of sub-network types, how the sub-networks differ, and the ways in which the sub-networks are aggregated. We can classify HetNets into *Single-RAT HetNets* and *Multi-RAT HetNets* based on whether the sub-networks of the HetNet use one or multiple radio access technologies. The survey in [38] has provided an introduction to the other ways of classifying the existing HetNet literature and various bandwidth aggregation approaches. We list the most relevant information here.

2.2.1 Single-RAT HetNet

As the name suggested, Single-RAT HetNet refers to HetNets with only one RAT. If the single RAT involved is a cellular technology such as LTE, we also call it a **Cellular HetNet**. A cellular HetNet combines macrocells with small cells such as picocells, femtocells, and distributed antenna systems [21]. A small cell is a base station with a lower power level compared with macrocells that is installed inside/outside buildings to allow better cellular coverage. Distributed antenna systems is a network of spatially separated antenna nodes connected to a common source to solve the bandwidth bottleneck problems in ultra-dense environments such as football stadiums. Cellular HetNets have been well standardized and deployed [42]. If the single RAT involved is WiFi, it is a Wireless Local Area Network (WLAN). Previous literature such as [10, 37] has explored flow mapping systems for a WLAN.

2.2.2 Multi-RAT HetNet

Multi-RAT HetNet is a type of HetNet of which the sub-networks use different RATs. In this dissertation, it refers to a HetNet comprising LTE and WiFi sub-networks, which we also call an **LTE-WiFi HetNet**. Since using different RATs, the sub-networks need to be further aggregated. There are several ways to aggregate the LTE and WiFi sub-networks.

3rd Generation Partnership Project (3GPP) currently has several paths to integrate LTE and WiFi. In the licensed spectrum, there are IP Flow Mobility (IFOM) [39] and Multiple Access Packet Data Network Connectivity (MAPCON) [6], while in the unlicensed spectrum, there are LTE WLAN integration with IPSec tunnel (LWIP), License Assisted Access (LAA) and LTE-WLAN Aggregation (LWA), etc [35]. The former is also called WiFi offloading, while the latter also called LTE-WiFi coexistence.

MAPCON offers the UEs to establish multiple connections to different Packet Data Networks (PDN) via different access networks and also a selective transfer of PDN connections between access networks. The usage of multiple PDNs is typically controlled by the network operators policies. By using MAPCON, offloading can be achieved relatively easily without requiring the UE to support client-based mobility protocols such as Dual Stack Mobile IPv6 (DSMIPv6). In contrast to MAPCON, IFOM uses DSMIPv6 to implement UE mobility. It requires both UEs and PDN servers to support DSMIPv6. However, comparing with MAPCON, its data offloading logic is more

UE-centric. At mobility between 3GPP and WLAN access, the UEs, instead of servers, determine which IP flows of a PDN connection are offloaded.

LWIP, LAA, and LWA use the technique called *carrier aggregation* to solve the coexistence issues of LTE and WiFi in the unlicensed spectrum. It essentially tunnels the LTE data from certain network layers through unlicensed media. Among them, only LWA allows the usage of LTE and WiFi simultaneously.

Besides the above, transport layer protocols such as MultiPath TCP can also be used in aggregating LTE and WiFi sub-networks, which has been tested in research, and commercial LTE networks [36, 30].

This dissertation assumes an LTE-WiFi HetNet in which LTE and WiFi use different spectra. Therefore, we assume there is no carrier aggregation in the system.

2.3 HetNet User Association Optimization

Flow mapping systems can be considered as a HetNet optimization approach that indirectly controls the resource allocation of a HetNet by only controlling the user/flow associations. This is in contrast with another group of previous work that assumes control to resource allocation at APs directly [8, 10, 20]. As stated in [13], this indirect control has the advantage of fewer changes to the network infrastructure and protocols, and therefore reduces the upgrading cost for network providers. We introduce more user association control schemes in both cellular HetNets and LTE-WiFi HetNets as follows.

2.3.1 Cellular HetNets

The cell selection in a 3GPP network without any WiFi network integrated is controlled by connection status metrics, such as Received Signal Strength (RSS). It is controlled by the Mobility Management Entity (MME) which has global information over a group of APs [32]. MME usually connects a UE to the best available AP based on the connection status metrics between APs and the UE. It is also called "Always Best Connected", or ABC in the literature.

Since the decision is based on connection status metrics, one way to control the user association in a cellular HetNet is based on a technique called Cell Range Expansion. The essence of this technique is to add a bias factor that boosts UEs' interpretation to the RSS of small cells compared

with that of the macrocell. It gives a higher probability for UEs to connect to the small cells. For example, the works in [31, 43] both rely on this technique. We will call the previous works using this technique as *indirect association control*, and those without using it *direct association control*.

1. Direct association control

Previous works such as [13, 40, 9] control the UE connections directly.

Bu. *et al.* [13] raised the concept of Generalized Proportional Fairness (GPF) in the context of an overlapping cellular system managed by one operator. GPF is the proportional rate fairness among all the UEs across all the APs in a wireless system. By assuming elastic flows and all APs are PF scheduled, they mapped the GPF problem to a 3-d maximum weight matching problem. This mapping proves the GPF problem is NP-hard. Based on this simplification, they proposed an offline algorithm that searches all the possible configurations, and runs maximum weighted matching for each configuration. Two faster event-based greedy algorithms were also provided. However, they did not provide how those event-based algorithms can be implemented by incremental calculation and minimum information at APs as we will describe in this dissertation. Their result shows that the greedy algorithm is close to the optimal algorithm and better than the Best-Signal and Max-Min algorithms. Their result is limited to the assumption that all APs are PF scheduled.

[40] explores the benefit to achieve α-fairness in a MultiPath TCP enabled cellular HetNet. It evaluated a centralized greedy heuristic compared to several local-view random assignment based approaches. The result suggests that greedy devices that utilize all available interfaces may result in a non-Pareto optimal allocation. Judiciously enabling multipath connections can achieve potential gains in fairness.

There are also distributed mapping systems designed based on game theory, such as [15]. [15] made the assumption that APs can send UEs reward feedback based on the UEs' usage of resources. They designed a learning algorithm based on a "trial and error" approach considering the resource contention among users as a non-cooperative game.

2. Indirect association control

Some other work used the SNR bias factors mentioned above to control the user associations in the cellular HetNet. [31, 43] considered the joint user association and spectrum management problem. The advantage of this indirect association control is that it requires fewer changes to the existing network infrastructure, compared with the direct control approaches. However, it can only control with a rougher granularity.

[43] used Lagrangian dual decomposition to divide the problem into two sub-problems which can be solved at the UE and AP separately. Each AP keeps tracks of a parameter μ which is related to the number of UE connections on it. Then it sends the μ to UEs. Each UE uses μ and its locally estimated peak rate to decide with which AP it should associate. The authors also provided one approach to use SINR biasing to control user associations. The Load-aware local greedy mapping system we will evaluate in Chapter 7 is similar to this approach in broadcasting load-related information from APs. However, we assume direct user association control.

The work in [31] also relies on range expansion. However, it used stochastic geometry to derive the theoretical mean proportionally fair utility of the network based on the coverage rate, and verified the theoretical optimal using numerical evaluation.

2.3.2 LTE-WiFi HetNets

When WiFi offloading is involved, 3GPP standard suggests using an Access Network Discovery and Selection Function (ANDSF) server [1] to control the network selection for UEs. Even though the standard has defined the protocol to use, the logic of how to select networks are left to operators. Therefore, there are many research proposals on how to implement that.

ATOM [33] provided a solution for the GPF optimization problem in an LTE-WiFi HetNet. Their work assumed sub-network aggregation using application layer protocols. The authors of ATOM provided a greedy algorithm that offloads UEs from LTE to WiFi greedily based on utility function changes. We will evaluate this algorithm as one of the representative global information based flow mapping system.

MOTA [16] designed an optimization framework for optimizing application flows over multiple RATs and multiple providers with assumptions different from [13]. The first difference is that MOTA included cost in its objective function. Another difference is that MOTA assumed all the APs can be changed to broadcast certain objective function related information to all the UEs inside its coverage no matter they are connected or not. This assumption is similar to the assumption in [43] we introduced in Section 2.3.1. The authors of MOTA provided two solutions called MOTA-STATIC and MOTA-MOBILE. The MOTA-STATIC algorithm assigns the associations of applications sequentially based on their weights. For each application, it greedily picks the operator that can maximize that UE's utility. The difference from the load-aware local greedy algorithm evaluated in Chapter 7 is that load-aware local greedy takes into consideration both the utility gain

from the new UE that tries to connect and the utility change from the UEs connected to the target AP. MOTA-MOBILE essentially treats the problem as a Generalized Assignment Problem (GAP), and used an iterative heuristic algorithm to solve it. This requires multiple rounds of broadcasting and communication between UEs and APs to converge for one scenario. We doubt that can be a good choice for a mobile environment.

[24] proposed a network-assisted user-centric WiFi-offloading model for maximizing per-user throughput. It used a discrete-time Markov chain to model the WiFi throughput, and a trust-region-dogleg algorithm to solve the non-linear optimization problem. [23] worked on intracell fairness optimization assuming the controls on both the mode selection and bearer-split scheduling. They proposed greedy algorithms based on bandwidth and delay estimations respectively. From their results, the proposed LTW-W system has up to 75% improvement compared with the results with the default Multipath TCP clients in terms of Jainś fairness index.

[27] studied the implication of more heterogeneous throughput models to the game theory approaches. From their analysis, the mixture of throughput models can lead to an improvement path that can be repeated infinitely. However, they show that by introducing appropriate hysteresis policies, the game can still converge to equilibria. [18] used matching games to solve the mode selection problem in an LTE-WiFi HetNets with LWA enabled APs.

The tests of Global Information Based Flow Mapping System in this dissertation can be considered as an evaluation of the value of adding a centralized ANDSF server to control UE associations for various LTE-WiFi HetNets.

2.4 Centralized Vs. Distributed

Another perspective on the previous work is based on whether they have a centralized control, no matter they belong to Cellular HetNets or LTE-WiFi HetNets.

For example, the literature we have mentioned above can be classified as follows.

1. Centralized: [13, 33, 23, 24, 31]
2. Distributed: [9, 18, 16, 43]

This dissertation will evaluate some representative mapping systems in both categories.

Chapter 3

Problem Formulation and Performance Metrics

3.1 Objective Functions of Mapping Systems

As described in Chapter 1, the Global Information based Flow Mapping Systems have explicit objective functions. For Local Information based Flow Mapping Systems, even though not having an explicit objective, their performance will still be measured against the same objective function. This dissertation focuses on using Generalized Proportional Fairness (GPF) as this objective function similar to the previous literature [13, 33]. In this section, we introduce some other options and explain why we choose to use GPF.

First, there are different metrics we can use as the inputs of the objective function, e.g. user throughput, cost, etc. Most previous literature uses throughput related metrics, such as the objective functions in [13, 43, 33, 40]. Previous work such as [16] adds cost to the objective functions. This dissertation focuses on throughput based objective functions. We do not include cost in the objective because it is very likely the same cost can have different meanings for various users. It is infeasible to devise a simple model to normalize costs.

3.2 Why GPF If Using Throughput Based Objectives

Even if we chose to use a throughput based objective, there are many options. A simple objective can be to maximize the sum of UE throughputs, which is called a *utilitarian solution* in [12]. However, as most of previous literature has found, this can easily lead to starvation of UEs which have poor connections to APs [12, 13]. An improvement to that approach can be adding starvation-free constraints to the optimization problem. However, as described in [29, 26], the proportional fairness naturally mitigates this problem by adding a logarithm utility function to each UE throughput before the sum operation in the objective function. As we have proved in Section 2.1.2, proportional fairness is equivalent to maximizing the sum of the logarithms of UE throughputs,

$$\sum_{i=1:N} \log(T_i)$$

where T_i is the throughput of the AP to UE_i. Section 2.1.2 provides two definitions of proportional fairness. Intuitively, the logarithm function puts more weight to the smaller throughput and less weight to the larger throughput. However, proportional fairness still maintains a better level of spectrum efficiency compared with max-min fairness [12, 13]. The maximization of the sum-of-logarithms objective is usually approximated at a single AP with the event-based scheme proposed in [29, 14]. The nice trade-off between the spectrum efficiency and fairness among users, and the simple event-based approximation that can converge to the optimal, make proportional fairness widely used in 3GPP cellular systems and the latest WiFi system [14, 28].

The generalized proportional fairness (GPF) is a natural extension to the concept of proportional fairness which tries to maximize the sum of throughput logarithms for UEs across different APs, i.e.

$$\sum_{j=1:M} \sum_{i=1:N} \log(T_{ij})$$

where T_{ij} is the throughput from AP_j to UE_i. The index i is for the indices of UEs, while j is for those of APs. N is the total number of UEs, while M is that for APs. We will use this convention throughout this dissertation.

GPF has been used in previous HetNet optimization literature such as [13, 43, 33]. Even

though the evaluation framework is general enough to use other types of fairness metrics as the objective function, such as alpha fairness [12, 40], we choose GPF because of its simple form and wide use. The evaluations of other types of objective functions will be considered as future work.

3.3 Mathematical Formulation of the HetNet Flow Mapping Problem

We model the HetNet flow mapping problem mathematically as follows. It is similar to the models in [13, 33].

$$\begin{aligned} &\text{Maximize} \quad \sum_{j=1\ldots M} \sum_{i=1\ldots N} log(\ T_{ij}) * x_{ij} \\ &\text{subject to} \\ &\qquad \sum_{j} x_{ij} = 1, \\ &\qquad x_{ij} \in \{0, 1\}, \\ &\qquad r_{ij} = u(S_{ij}), \\ &\qquad T_{ij} = v(\hat{x_j}, r_{ij}) \end{aligned} \tag{3.1}$$

As we can see, the optimization objective is the sum of throughput logarithms, i.e. the Generalized Proportional Fairness as we discussed in Section 3.2. x_{ij} is the association variable that represents whether the traffic of UE_i should come through AP_j. T_{ij} is the apportioned throughput of UE_i when it is connected to AP_j after considering resource contention, while r_{ij} the throughput before considering resource contention. We will detail the concepts of different types of throughputs in Section 3.5.

The first constraint means that every UE can only use exactly one interface at a time. The second constraint indicates that the association variable x_{ij} must be an integer in {0, 1}. This means there is no flow splitting. The third constraint expresses that r_{ij} is a function of the connection status of the link from AP_j to UE_i (S_{ij}). The fourth constraint shows the resource contention at AP_j can be modeled as a function of the effective rate UE_i can achieve if connected to AP_j (r_{ij}) and the associate variable vector for all the UEs under AP_j ($\hat{x_j}$). Based on the scheduling principle adopted by an AP of a specific RAT, the function can have different forms, which we will detail in

Section 3.5.

3.4 Alternative Fairness Metrics

We measure the performance of flow mapping systems using the following alternative fairness metrics.

3.4.1 Jain's Fairness Index

Jain's fairness index is a fairness index introduced by Raj Jain *et al.* in 1984 [22], which has been widely used in telecommunication. It is defined as follows,

$$JFI = \frac{(\sum_i x_i/y_i)^2}{n\sum_i (x_i/y_i)^2} \tag{3.2}$$

where $\hat{x} = \{x_1, ..., x_n\}$ is a vector of throughput while $\hat{y} = \{y_1, ..., y_n\}$ is the throughput vector from the optimal solution of a target fairness objective. We will use the same notation for the descriptions of the rest of the fairness indices.

It has the following good properties,

1. Independent of scale
2. Continuous
3. Applies to any number of users
4. Bounded between 0 and 1. 0 means the most unfair while 1 the fairest, which is intuitive for interpretation. For example, if all the x_i are equal, the system reaches a JFI of 1 from Eq. (3.2).

For the proof of these properties, please refer to [22]. In our evaluation, we use the optimal solution which can achieve the best GPF value as the baseline vector $\hat{y}$.

3.4.2 Normalized Throughput Fairness Index

The normalized throughput fairness index is used in [13]. It is defined as,

$$TFI = \frac{(\sum_i T_i)^2}{n\sum_i T_i^2} \tag{3.3}$$

where T_i is the throughput of UE_i. It can be seen as a special case of Jain's fairness in which the target fairness objective is equal throughput fairness. Note that the magnitude of TFI is an indicator of throughput fairness, but does not capture the resource usage efficiency at all. For example, if all the flows are assigned to a single WiFi AP, TFI will be 1 while the total throughput can be low.

3.4.3 Max-Min Rate Fairness Index

It is defined as,

$$MMR = \frac{max(x_i)}{min(x_i)} \tag{3.4}$$

which is the rate between the maximum and minimum elements of the vector $\hat{x}$.

3.5 Models and Examples of Bandwidth Sharing Effects

3.5.1 Terms Used for Different Types of Throughput

In this dissertation, we will discuss the following two types of throughputs and use the following convention to denote them.

1. Effective throughput/bit rate (r_{ij})

   It is the throughput before considering the resource contention with the other UEs connected to the same AP.

2. Apportioned throughput (T_{ij})

   It is the throughput after considering the resource contention with the other UEs connected to the same AP.

3.5.2 Proportional Fair Sharing

As proved in previous literature [13, 33], under the assumption of elastic traffic, proportional fair becomes equal time fairness. If there are K UEs connected to a proportional fair AP, the

normalized time share UE_i can get (τ_{ij}) will be,

$$\tau_{ij} = \frac{1}{K} \tag{3.5}$$

Since the throughput of a UE equals its effective rate multiplied by the time share it achieves, the final apportioned throughput of the UE (T_{ij}) can be modeled as,

$$T_{ij} = \frac{r_{ij}}{K} \tag{3.6}$$

where r_{ij} is the effective rate from AP_j to UE_i. From Eq. (3.6), we see that the apportioned throughput of a UE under a PF scheduled AP is only related to its effective throughput (r_{ij}) and the total number of UE connected to that AP (K). For example, if we have two UEs connected with the effective rates of (4, 8) Mbps, the final apportioned throughput will be (4/2, 8/2) Mbps. The aggregate throughput of the two UEs will be 6 Mbps.

3.5.3 Max-Min Fair Sharing

As described in Section 2.1.1, when the traffic types of UEs are all elastic, max-min fair is equivalent to equal throughput fair sharing. Let the set of UEs connected to AP_j be A_j. Assume every UE obtains an opportunity to send L bytes in a scheduling round. The total time for a scheduling round will be $t_a = \sum_{i \in A_j} \frac{L}{r_{ij}}$. UE_i's normalized time share τ_{ij} can be then calculated as follows,

$$\begin{aligned} \tau_{ij} &= \frac{\frac{L}{r_{ij}}}{t_a} \\ &= \frac{\frac{L}{r_{ij}}}{\sum_{i \in A_j} \frac{L}{r_{ij}}} \\ &= \frac{\frac{1}{r_{ij}}}{\sum_{i \in A_j} \frac{1}{r_{ij}}} \end{aligned} \tag{3.7}$$

Therefore, the final throughput of UE_i is,

$$T_{ij} = r_{ij} \times \tau_{ij} = \frac{1}{\sum_{i \in A_j} \frac{1}{r_{ij}}} \tag{3.8}$$

With the two flows (with the effective rates of (4, 8)) in the example in Section 3.5.2 connected to a max-min fair AP, the final throughput can be calculated as,

$$T_{ij} = \frac{1}{\frac{1}{4} + \frac{1}{8}} = \frac{8}{3} \approx 2.67$$

Both UEs will get an apportioned throughput of 2.67 Mbps. The aggregate throughput will be 5.33 Mbps. Note comparing with the result in Section 3.5.2, the minimum throughput increases while the aggregate throughput decreases. The results of this example and the one in Section 3.5.2 demonstrate that proportional fair gains higher spectrum efficiency with a slight decrease in minimum throughput, when comparing with max-min fair.

3.6 Incremental GPF Evaluation

In this section, we analyze the impacts when adding/removing a flow to/from an AP in terms of the objective function value, i.e. GPF value. In general, the change of GPF has two parts, the GPF change of the existing flows on an AP and the GPF change of the moving flow. We detail the derivations for the GPF changes when adding/removing a flow on PF scheduled and max-min fair scheduled APs respectively.

3.6.1 Adding a Flow to a Proportional Fair AP

For a proportional fair AP_j with K UEs ($K \geq 0$) connected to it and an effective throughput vector of $r_j = < r_{1j}, r_{2j}, ..., r_{Kj} >$, if we add another UE_q to it with an effective throughput of r_{qj}, the delta of the objective function value can be calculated as follows.

Let A_j be the set of UEs connected to AP_j. Before UE_q connects to AP_j, $|A_j| = K$. After UE_q is added, $|A_j| = K + 1$. Based on Eq. (3.6), we know that the new apportioned throughput of flow k becomes $T'_{kj} = T_{kj} \times \frac{K}{K+1}$, for $k \in A_j$. The change of GPF (ΔU) can be calculated as,

$$
\begin{aligned}
\Delta U &= (\sum_{k \in A_j} log(T'_{kj}) - log(T_{kj})) + log(T_{qj}) \\
&= (\sum_{k \in A_j} log(T_{kj} \times \frac{K}{K+1}) - log(T_{kj})) + log(T_{qj}) \\
&= (\sum_{k \in A_j} log(T_{kj}) + log(\frac{K}{K+1}) - log(T_{kj})) + log(T_{qj}) \\
&= K \times log(\frac{K}{K+1}) + log(T_{qj}) \\
&= K \times log(\frac{K}{K+1}) + log(\frac{r_{qj}}{K+1})
\end{aligned}
\tag{3.9}
$$

Note $K \geq 0$. When $K = 0$, the first part becomes 0, and the delta will be the utility of the new flow, i.e. $log(r_{qj})$. It is also interesting to notice that the GPF delta in this case is not related to the throughput values of the existing flows at all. It is only related to the number of UEs connected to the AP ($|A_j|$) and the effective rate of the moving flow (r_{qj}).

3.6.2 Removing a Flow from a Proportional Fair AP

Similarly, if we are going to remove a flow from the same AP, the new total number of flows will be $K - 1$. The objective function value change will be,

$$
\begin{aligned}
\Delta U &= (\sum_{k \in A_j - q} log(T'_{kj}) - log(T_{kj})) - log(T_{qj}) \\
&= (\sum_{k \in A_j - q} log(T_{kj} \times \frac{K}{K-1}) - log(T_{kj})) - log(T_{qj}) \\
&= (\sum_{k \in A_j - q} log(T_{kj}) + log(\frac{K}{K-1}) - log(T_{kj})) - log(T_{qj}) \\
&= (K-1) \times log(\frac{K}{K-1}) - log(T_{qj}) \\
&= (K-1) \times log(\frac{K}{K-1}) - log(\frac{r_{qj}}{K})
\end{aligned}
\tag{3.10}
$$

Note Eq. (3.10) only works for $K \geq 2$. Since removing a flow, $K \geq 1$. If $K = 1$, the delta will be $-log(r_{qs})$. We also notice that the information needed to calculate the GPF delta for removing a flow is the same as that for adding a flow.

3.6.3 Adding a Flow to a Max-Min Fair AP

For a max-min fair AP_j with K ($K \geq 0$) UEs connected with effective throughput vector $r_j = < r_{1j}, r_{2j}, ..., r_{Kj} >$, if another UE q is added to it with an effective throughput of r_{qj}, the delta of total objective function value can be calculated as follows,

Let the round time $R = \sum_{k=1...K} \frac{1}{r_{kj}}$, the change of objective function value is,

$$\Delta U = (K+1) \times log(\frac{1}{R + \frac{1}{r_{qj}}}) - K \times log(\frac{1}{R}) \tag{3.11}$$

Note, if UE_q is the first flow on that AP, Eq. (3.11) cannot work as $R = 0$. In that case, $\Delta U = log(r_{qj})$. We notice that for max-min scheduled APs, besides the number of existing UEs connected on the AP (K) and the effective rate of the new flow (r_{qj}), it requires one more piece of information, i.e. the sum of round time of all the existing UEs (R).

3.6.4 Removing a Flow from a Max-Min Fair AP

Similarly, if a flow needs to be removed from a max-min fair AP, the change of objective function value is,

$$\Delta U = (K-1) \times log(\frac{1}{R}) - K \times log(\frac{1}{R - \frac{1}{r_{qj}}}) \tag{3.12}$$

Note Eq. (3.12) only works for $K \geq 2$. Since we are removing a flow, $K \geq 1$. If $K = 1$, $\Delta U = -log(r_{qj})$. We also find that the information needed to calculate the GPF delta for removing a flow is the same as that for adding a flow.

Chapter 4

Network Models for Simulation

4.1 Fading Model

In this dissertation, we use the following fading model that ignores shadowing, antenna direction and gains, interference among APs, etc. We consider them as a constant factor for all the flows in the system.

We use the fading model and the parameter value from [25],

$$S/N = \frac{SP}{NP} = \frac{\kappa}{d^{\eta}} \tag{4.1}$$

where κ is the constant factor, and $\eta = 2.6$. The κ values of various radio access technologies are usually different. This model has faster fading compared with the free-space model ($\eta = 2$), but not as fast as expected in a dense urban environment.

4.2 Bit Rate Model

Wireless protocols use multiple levels of Modulation and Coding Schemes (MCS) to modulate and code the raw information. They adapt the level of MCS in various channel conditions to limit error rates. Table 4.1 shows an example of that. Each row in the table is one MCS. The second column "Signal State" is how many signal states (N_s) an MCS can code using one symbol. For example, QPSK can code 4 signal states in one symbol. The third column is the number of bits

per symbol (N_b). The relation between N_s and N_b is,

$$N_b = log_2(N_s) \tag{4.2}$$

Table 4.1: Modulation Schemes

MCS	Signal States (N_s)	Bits Per Symbol (N_b)
BPSK	2	1
QPSK	4	2
16-QAM	16	4
64-QAM	64	6
256-QAM	256	8

For a given wireless protocol, the coding scheme that can encode more bits per symbol requires higher S/N. This can be observed in the analysis in Section 4.6.2 based on Shannon-Hartley theorem.

4.3 Common Concepts and Assumptions

Wireless protocols use Error Correction Code (ECC) to further decrease the error rates at the receiver. For coding N_{raw} bits of raw information, if the number of ECC bits added is N_{ecc}, the *code rate* (Cr) of this coding scheme is defined as the $Cr = \frac{N_{raw}}{N_{raw}+N_{ecc}}$. We call the bit rate before applying the code rate *coded bit rate* or *coded rate* (r_c), and the one after applying the code rate as *nominal bit rate* or *nominal rate* (r_n). From the definition, we know that,

$$r_n = r_c * C_r \tag{4.3}$$

Since both 802.11n and LTE use Orthogonal Frequency-Division Multiplexing (OFDM), the coded bit rate can be calculated as follows. OFDM broadband channels are subdivided using frequency division multiplexing into subcarriers of more narrow bandwidth. Data is transmitted in parallel in all the subcarriers. For example, with a 20 MHz channel, LTE has 1200 data carrying subcarriers, while WiFi only has 52 data carrying subcarriers. The *symbol time* (t_{sym}) is defined as the time that is required to transmit a symbol. The *symbol rate* (r_s) is defined as the number of

symbols that can be transmitted per time unit. From the definition, we know,

$$r_s = \frac{N_{ch}}{t_{sym}} \tag{4.4}$$

where N_{ch} is the number of data carrying subcarriers. The coded bit rate can be calculated as,

$$r_c = r_s \times N_b = \frac{N_{ch} \times N_b}{t_{sym}} \tag{4.5}$$

Furthermore, there is overhead in various wireless protocols, such as control overhead and arbitration overhead. We call the bit rate after adding this overhead as the *effective rate/throughput* (r_e).

We define the *protocol efficiency* as,

$$e = \frac{r_e}{r_n} \tag{4.6}$$

In the following two sections, we will introduce how to calculate these bit rates and protocol efficiencies for both 802.11 and LTE.

We assume the bandwidth of 802.11 and LTE are both 20 MHz, no channel bonding unless otherwise stated. This ensures the results are free from the impact of spectrum bandwidth difference.

4.4 802.11 Model

Even though the evaluation framework we design in this dissertation is general to any 802.11 protocols, we choose 802.11n as the standard to use in our simulation. In this section, we detail how we model the throughput of 802.11n.

We make the following assumptions in this analysis. The protocol efficiencies of the other types of 802.11 networks can be derived with minor changes in parameter values.

1. We assume a 20 MHz bandwidth and 400 ns guard interval over 5 GHz.
2. We focus on the analysis of downlink.
3. We use the video traffic category as an example.

4. We assume unicast messages (instead of broadcast messages).

5. We assume no Transmit Opportunity or Media Access Control (MAC) layer frame aggregation for this analysis. The analysis can be extended to those cases by multiplying the results by constant factors.

6. In the Physica (PHY) layer, We assume High Throughput (HT) mixed mode is used which contains a preamble compatible with HT and non-HT receivers. We make this assumption because support for HT-mixed format is mandatory for 802.11n. Section 20 of IEEE Std. 802.11-2009 defines and describes the HT modes in the PHY layer [2].

4.4.1 Nominal Rate

In OFDM, subcarrier spacing is deliberately selected to cancel out inter-carrier interference without the need for guard bands or expensive bandpass filters. This implies that the spacing of the subcarriers is the reciprocal of the useful symbol time (t_{usym}). For 802.11n OFDM using 64 point fast Fourier transform (FFT) in a 20 MHz channel, it uses 64 subcarriers spaced 312.5 KHz apart. The useful symbol time can be calculated as,

$$t_{usym} = \frac{1}{312.5\ KHz} = 3.2\ \mu s \tag{4.7}$$

As we assume a 400 ns guard interval, the symbol time $t_{sym} = t_{usym} + t_{gi} = 3.2\ \mu s + 0.4\ \mu s = 3.6\ \mu s$.

In the 64 subcarriers, only 52 are used for data. There are 4 of them as pilot subcarriers, and 8 of them serving as a guard band. Based on the standard, the nominal rate of 802.11n can be calculated using Eq. (4.3) as shown in Table 4.2.

Table 4.2: 802.11n Nominal Rate (400 ns GI)

Modulation	Code Rate	Nominal Rate
BPSK	1/2	7.2
QPSK	1/2	14.4
QPSK	3/4	21.7
16-QAM	1/2	28.9
16-QAM	3/4	43.3
64-QAM	2/3	57.8
64-QAM	3/4	65.0
64-QAM	5/6	72.2

4.4.2 Analysis of 802.11n Protocol Efficiency

Besides the nominal throughput in Table 4.2, other factors can add overhead and therefore reduce the real throughput measured from the application layer (It is the reason for calling the throughput in the previous section nominal rate.). For example, those factors can include time gaps from arbitration, ACKs, frame overhead in the MAC layer, and preambles in the PHY layer.

With different 802.11 variants, there will be minor differences in the calculation process of the overhead. However, the general process is the same. We demonstrate how the protocol efficiency of 802.11n can be calculated as an example.

4.4.2.1 Overhead Analysis

There are mainly four types of overheads,

1. PHY layer overhead
2. MAC layer overhead
3. Arbitration overhead
4. Backoff overhead

We detail how we can calculate the efficiency factors for each type of overhead in the following.

1. PHY layer overhead

   PHY layer adds a 16-bit header called the *service* and a 6-bit tail called the *tail*. Besides that, the minimum preamble overhead with HT-mixed format in 802.11n is 9 symbols, which is 32.4 μs when t_{sym} =3.6 μs.

2. MAC frame overhead.

   The MAC layer of 802.11n adds a minimum of 28 bytes, which includes 24 bytes of header and 4 bytes of Frame Check Sequence (FCS) at the end. We assume the data from IP layer is 1500 bytes. Therefore, the number of bytes in MAC Protocol Data Unit (MPDU) (n_{mpdu}) is 1528.

3. Arbitration overhead

The 802.11 family of protocols use the distributed coordination function (DCF) protocol for controlling access to the physical medium. DCF employs the Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) using a truncated exponential backoff algorithm. A UE must sense the status of the wireless medium before transmission. If it finds that the medium is continuously idle for a certain duration, it is then permitted to transmit a frame. If the channel is found busy during the interval, the UE should defer its transmission with a certain backoff time. 802.11n, inheriting the Enhanced Distributed Coordination Function (ECDF) from 802.11e, uses the following two types of inter-frame time intervals,

(a) Short Inter-frame Spacing (SIFS)

(b) Arbitration Inter-frame Spacing (AIFS)

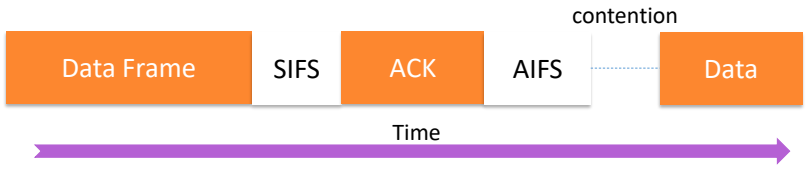

Figure 4.1: 802.11 arbitration time sequence.

Fig. 4.1 shows the process of MAC layer arbitration. As we can, a UE needs to sense the medium for an SIFS between any data frame and the ACK frame after it. An Arbitration Inter-frame Spacing (AIFS) must be added after any ACK frame. The time length of AIFS varies according to traffic types. Assuming video traffic in this analysis, the time length of AIFS is defined as,

$$t_{AIFS} = t_{SIFS} + 2 * t_{slot} \tag{4.8}$$

In 802.11n over 5 GHz, $t_{SIFS} = 16\ \mu s$ and $t_{slot} = 9\ \mu s$. Therefore, $t_{AIFS} = 34\ \mu s$.

The ACK frame is 14 bytes. Adding the 22-bit MAC layer overhead, it is 134 bits. Since the ACK frame also needs the preamble, the total time for the ACK frame can be calculated as the following,

$$t_{ack} = \lceil \frac{134}{52 * N_{bpsym}} \rceil * t_{sym} + t_{pre} \tag{4.9}$$

Using 64-QAM 2/3 as an example, the number of symbols required by the ACK frame is $\lceil \frac{134}{52*4} \rceil = 1$. Therefore, $t_{ack} = 3.6 + 32.4 = 36\ \mu s$.

4. Backoff overhead

   In DCF, a random backoff time is also defined to solve the possible collisions from multiple UEs which sense the channel busy and defer their access simultaneously. The standard has defined the range of contention window size for various traffic types. For video traffic, the minimum (CWMin) and maximum (CWMax) of the window sizes are 7 and 15 time slots respectively. The mean number of backoff time slots will be CWMin / 2 = 3.5. As the arbitration time slot in 802.11n is 9 μs, the mean backoff time ($t_{backoff}$) is $3.5 \times 9 = 31.5\ \mu s$.

After the above calculations, the only left part is the time of data frames. The time to send a Physical Protocol Data Unit (PPDU) can be calculated as follows.

$$t_{data} = t_{sym} \times \lceil \frac{n_{phy} + n_{mpdu} \times 8}{n_{bpsym} \times n_{sub}} \rceil \tag{4.10}$$

where n_{ip} is the number of data bytes sent from IP layer in one frame, while n_{phy} and n_{mac} are the number of frame overhead in PHY layer (in bit) and MAC layer (in byte). n_{bpsym} is the number of bits per symbol. n_{sub} is the number of subcarriers. Using 64-QAM 2/3 as an example,

$$t_{data} = 3.6\ \mu s\ \times\ \lceil \frac{22 + (1528) \times 8}{4 \times 52} \rceil \approx 212.4\ \mu s$$

Based on the analysis above, the efficiency factor (e) can be expressed as,

$$e = \frac{t_{data}}{t_{data} + t_{pre} + t_{arb} + t_{sifs} + t_{ack} + t_{backoff}} \tag{4.11}$$

where t_{pre} is the time for PHY layer preambles, t_{arb} the time for MAC layer arbitration time, and $t_{backoff}$ the MAC layer backoff time analyzed above. Using 64-QAM 2/3 as an example,

$$e = \frac{212.4}{212.4 + 32.4 + 34 + 36 + 16 + 31.5} \approx 0.584$$

The effective rates can be therefore calculated using Eq. (4.6). The resulting effective rates are shown in Table 4.3.

Table 4.3: 802.11n Effective Rate (400 ns GI)

Modulation	Code Rate	Nominal Rate	Protocol Efficiency	Effective Rate
BPSK	1/2	7.2	0.908	6.560
QPSK	1/2	14.4	0.841	12.143
QPSK	3/4	21.7	0.785	17.008
16-QAM	1/2	28.9	0.732	21.138
16-QAM	3/4	43.3	0.650	28.146
64-QAM	2/3	57.8	0.584	33.740
64-QAM	3/4	65.0	0.552	35.879
64-QAM	5/6	72.2	0.524	37.880

4.5 LTE Model

We assume LTE-Frequency Division Duplex (FDD) over a 20 MHz channel in this analysis. We conduct the analysis with the downlink user plane Physical Downlink Shared Channel (PDSCH). The PDSCH is the main data bearing channel in LTE.

4.5.1 Coded Bit Rate

LTE employs Orthogonal Frequency-Division Multiple Access (OFDMA) which is a multi-user version of the Orthogonal Frequency-Division Multiplexing (OFDM) digital modulation scheme. In OFDMA, the radio resources are divided into two-dimensional (2D) regions over time and frequency. Each user gets subcarriers grouped in 2D, as shown in Fig. 4.2, compared to only in the time dimension in OFDM. This allows multiple users to transmit in low rates simultaneously. This also simplifies the collision avoidance procedure. It does not need a CSMA/CA process as in the 802.11 protocol.

The coded bit rate of LTE without considering code rate or any overhead can be calculated as follows.

PDSCH carries data in Transport Blocks (TB), which is a MAC Protocol Data Unit. Each transport block is passed from the MAC layer to the PHY layer once per Transmission Time Interval (TTI) which is 1 ms. In the time dimension, one LTE frame is 10 ms which consists of ten 1 ms subframes. Each subframe contains two slots. Resources are assigned on a basis of two slots. One Physical Resource Block (PRB) is one slot (0.5 ms) long in time and 180 Hz wide in frequency. Fig. 4.2 shows two PRBs. Each PRB has 12 subcarriers in the frequency domain and 7 symbols (if with a normal Cyclic Prefix) in the time domain. One subcarrier by one symbol is called a Resource Element (RE), which is the smallest unit of resources in LTE. Each cell in Fig. 4.2 is a RE. We can

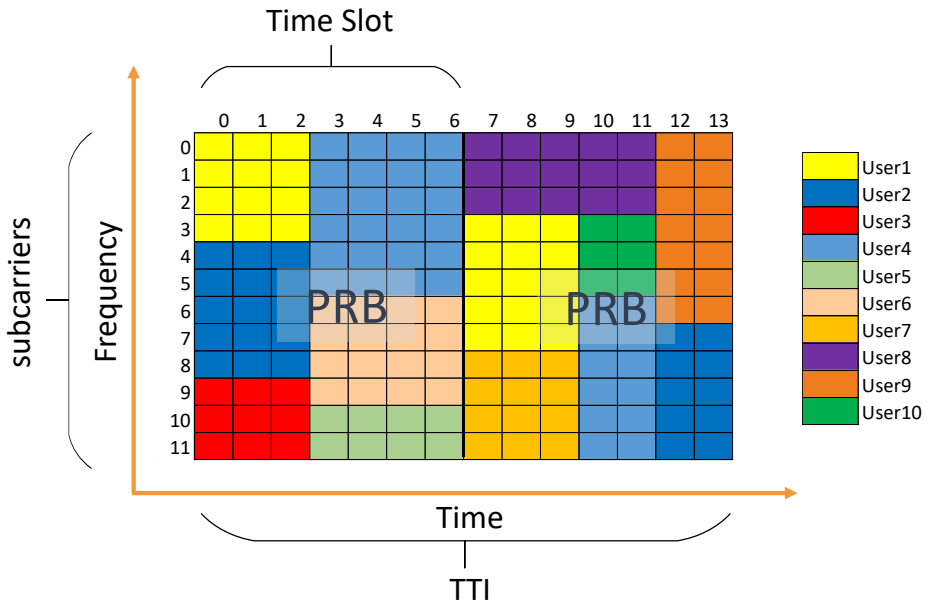

Figure 4.2: OFDMA of LTE downlink

see that one PRB has $12 \times 7 = 84$ REs. For LTE with 20 MHz bandwidth, there are 100 PRBs. Since each subframe has 2 slots, the number of REs in one subframe $N_{re} = 84 \times 2 \times 100 = 16800$. Therefore, the coded bit rate (r_c) can be calculated as,

$$\begin{aligned} r_c &= \frac{N_{re} \times N_b}{t_{sf}} \\ &= \frac{16800 \times N_b}{1\ ms} \end{aligned} \tag{4.12}$$

where N_b is the number of bits per symbol for a specific MCS and t_{sf} the time length for a subframe. From the LTE standard [19], we know there are only three modulation schemes, i.e. (QPSK, 16QAM, 64QAM). The coded bits per symbol for them are (2, 4, 6) respectively. Based on the number of bits for the MCS, we can obtain the raw rate for each MCS as shown in the 4th column in Table 4.4.

4.5.2 Nominal Rate

With coded bit rate, we can calculate the nominal rate if we know the code rate using Eq. 4.3. The code rate we use is from the 3GPP documentation [34]. It is showed in the 5th column in Table 4.4. The resulting nominal rates are shown in the 6th column of Table 4.4.

4.5.3 Effective Rate

For the effective rates which exclude the overhead, we use the rates derived from the transport block size (TBS) in Table 7.1.7.2.1-1 of 3GPP standard [4]. Transport block size is a limit in the 3GPP standard that defines how many bits are allowed to be transferred in 1ms, given the MCS index and number of PRBs. Therefore, given the transport block size (T_{tbs}), the final effective rate will be,

$$T_{eff} = \frac{T_{tbs}}{1 \times 10^{-3}} \ bps \tag{4.13}$$

With the analysis above, we can get the LTE rate table as follows,

Table 4.4: LTE Rate Table

MCS Index	Modulation	N_b	Coded Rate	Code Rate	Nominal Rate	Effective Rate
0	QPSK	2	33.6	0.117	3.938	2.792
1	QPSK	2	33.6	0.153	5.152	3.624
2	QPSK	2	33.6	0.188	6.333	4.584
3	QPSK	2	33.6	0.245	8.236	5.736
4	QPSK	2	33.6	0.301	10.106	7.224
5	QPSK	2	33.6	0.370	12.436	8.761
6	QPSK	2	33.6	0.438	14.733	10.296
7	QPSK	2	33.6	0.514	17.259	12.216
8	QPSK	2	33.6	0.588	19.753	14.112
9	QPSK	2	33.6	0.663	22.280	15.84
10	16 QAM	4	67.2	0.332	22.313	15.84
11	16 QAM	4	67.2	0.369	24.806	17.658
12	16 QAM	4	67.2	0.424	28.481	19.848
13	16 QAM	4	67.2	0.479	32.156	22.92
14	16 QAM	4	67.2	0.540	36.291	25.456
15	16 QAM	4	67.2	0.602	40.425	28.336
16	16 QAM	4	67.2	0.643	43.181	30.576
17	64QAM	6	100.8	0.428	43.116	30.576
18	64QAM	6	100.8	0.455	45.872	32.856
19	64QAM	6	100.8	0.505	50.892	36.696
20	64QAM	6	100.8	0.554	55.814	39.232
21	64QAM	6	100.8	0.602	60.638	43.816
22	64QAM	6	100.8	0.650	65.559	46.888
23	64QAM	6	100.8	0.702	70.777	51.024
24	64QAM	6	100.8	0.754	75.994	55.056
25	64QAM	6	100.8	0.803	80.916	57.336
26	64QAM	6	100.8	0.853	85.936	61.664
27	64QAM	6	100.8	0.889	89.578	63.776
28	64QAM	6	100.8	0.926	93.319	75.376

Note: all the bit rates are in Mbps.

4.6 Mapping UE-AP Distance to MCS Entry

4.6.1 Searching Strategy

To use the tables we derived from the Sections 4.4 and 4.5 to model UEs' effective throughput, we need a mapping from UE-AP distance to MCS entry. Introduced in Section 4.1, we already have a fading model that can map UE-AP distance to S/N. Then the part missing is only a mapping from S/N to MCS entry.

If we know the minimum S/N to support each MCS, given the S/N a UE can achieve at a certain distance (S/N_d), we can use the following searching strategy to determine which MCS entry should be used. If we search the "Minimum S/N" column of the MCS table from bottom to top, the entry to use is the one of which the minimum S/N is the closest lower bound for the target S/N_d. In other words, it picks the first entry that has a minimum S/N less than the target S/N_d.

For example, if the target S/N_d is 31, the entry at the bottom with MCS index 3 is first checked. The minimum S/N to support it is 50, which can not be used since it is larger than 31. Then, MCS index 2 with a minimum SNR of 30 is checked. We find it is less than 31 which is the first one that is less than the target. Therefore, we should use MCS index 2.

Table 4.5: Fake Table to Demonstrate MCS Index Searching

New MCS Index	Nominal Rate	Minimum S/N
0	0	0
1	5	10
2	10	30
3	15	50

Note to make sure we get a valid MCS index when the target S/N is smaller than the minimum S/N in the whole table, we add one more entry at the top of the table. Therefore, we use the term *New MCS Index* here (and in the tables in Section 4.6.2) to indicate that they are one larger than the MCS index in the standards.

4.6.2 Mapping Tables

We derive the mapping tables required for the above searching process for 802.11n and LTE using the following method.

Given S/N = $\frac{SP}{NP}$ for one UE, we can calculate its max sustainable user data throughput using,

$$r = Blog(1 + S/N) = Blog(1 + \frac{SP}{NP}) \tag{4.14}$$

Therefore, if the nominal bit rate associated with an MCS is r_n, then the Shannon-Hartley minimum S/N required to support the nominal rate is,

$$S/N_{min} = 2^{\frac{r_n}{B}} - 1 \tag{4.15}$$

Since we assume both 802.11n and LTE use 20 MHz bandwidth, $S/N_{min} = 2^{r_n/20} - 1$ in this case.

It is well-known that the Shannon-Hartley bound is only a theoretical bound that is not reachable within a single spatial channel. Furthermore, if we use the calculated S/N_{min} directly, it can cause MCS "flapping" when UE is around a distance that maps to SINRs between two MCS entries. Consequently, the values used in the MCS shadow tables of S/N minima are approximated as

$$S/N_{min} = \phi \times \psi \times (2^{\frac{T}{B}} - 1) \tag{4.16}$$

where ϕ is a constant factor used to avoid MCS flapping while ψ a factor that approximates the ratio between the "real world" S/N_{min} and the Shannon-Hartley S/N_{min}. It is clear that the effects of ϕ and ψ can be subsumed in the constant κ in Eq. 4.1.

The resulting table for LTE is shown in Table 4.6.

Table 4.6: Minimum S/N Table for LTE

New MCS Index	Nominal Rate	Minimum SINR
0	0.000	0.000
1	3.938	0.146
2	5.152	0.195
3	6.333	0.245
4	8.236	0.330
5	10.106	0.419
6	12.436	0.539
7	14.733	0.666
8	17.259	0.819
9	19.753	0.983
10	22.280	1.164
11	22.313	1.167
12	24.806	1.362
13	28.481	1.683
14	32.156	2.048
15	36.291	2.517
16	40.425	3.059
17	43.181	3.466
18	43.116	3.456
19	45.872	3.903
20	50.892	4.835
21	55.814	5.920
22	60.638	7.179
23	65.559	8.700
24	70.777	10.622
25	75.994	12.926
26	80.916	15.516
27	85.936	18.655
28	89.578	21.299
29	93.319	24.386

The resulting table for WiFi is shown in Table 4.7.

Table 4.7: Minimum S/N Table for WiFi

New MCS Index	Nominal Rate	Minimum S/N
0	0.0	0.000
1	7.2	0.284
2	14.4	0.650
3	21.7	1.119
4	28.9	1.722
5	43.3	3.490
6	57.8	6.407
7	65.0	8.514
8	72.2	11.219

Chapter 5

Simulation Methodology

5.1 Simulation Topologies

Simulation topologies specify the following,

- The locations and coverage of APs
- The locations of UEs

5.1.1 AP Placement

As was stated in the introduction, we assume the system to be optimized is divided into areas for scalability. The UE mapping systems under consideration in this research only optimize flows from APs to UEs inside the same area. In our simulations, we use a circle of nominal radius 1 unit to represent this area. We call this area a *simulated cell*. Note it is a conceptual cell that might contain multiple LTE cells. The circle is centered at the origin. The long-dashed line in Fig. 5.1 represents a simulated cell. There are M APs located inside the area. They can be either LTE or WiFi APs.

More specifically, we simulate with the following numbers of LTE APs (N_{lte}) and WiFi APs (N_{wifi}). We use $N_{lte} = 1$ and $N_{wifi} = M - 1$. The LTE and WiFi APs are placed in the simulated cell as follows:

1) The LTE AP is always placed at the center/origin;

2) The $M-1$ WiFi APs are evenly distributed on a circle with the same center as the simulated cell. The radius of that circle is denoted as a_w, where $0 < a_w \leq 1$. In polar coordinates, the location of WiFi AP_j is $(a_w, \frac{2\pi \times j}{M-1})$, where $j = \{0, ..., M-2\}$.

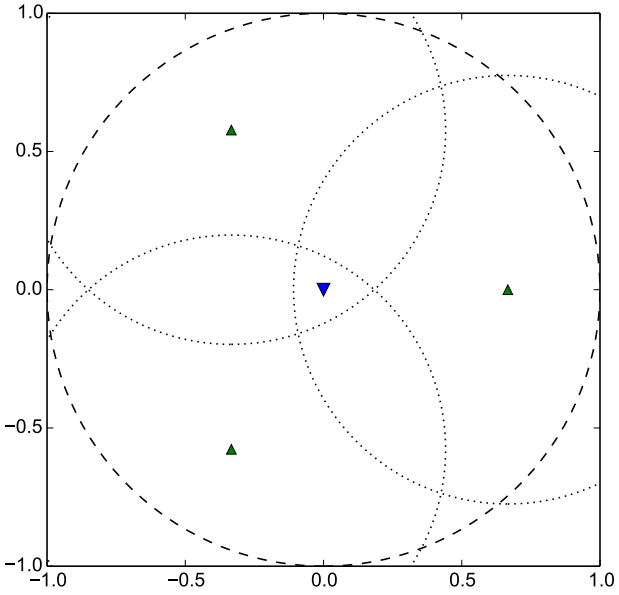

Figure 5.1: Exemplar topology.

For example, Fig. 5.1 shows a topology we use in our baseline evaluations in Section 6.1 with 1 LTE AP and 3 WiFi APs. We choose a_w=2/3 as it is tested that this a_w value maximizes the GPF values of the flow mapping systems in the scenarios with uniform UE topology.

5.1.2 AP Coverage

From the fading model we described in Section 4.1, we know that the coverage of each AP is determined by the value of κ. Unless otherwise noted, we use the following κ value for all the LTE and WiFi APs.

As described in Table 4.6, the minimum S/N that supports the lowest-rate LTE MCS is 0.146. To ensure the coverage of the LTE AP is exactly the range of the simulated cell, i.e. the

coverage radius of the LTE AP $R_{lte} = 1$, we can calculate the κ value based on Eq. (4.1) as,

$$\begin{aligned} \kappa &= S/N_{min} \times d^{\eta} \\ &= 0.146 \times 1^{2.6} \\ &= 0.146 \end{aligned} \tag{5.1}$$

We by default use $\kappa = 0.146$ for both LTE and WiFi APs. However, since WiFi has a higher S/N_{min} (0.284) in the analysis for Table 4.7, the coverage radius of WiFi APs are smaller than the LTE APs, which can be calculated as follows,

$$\begin{aligned} R_{wifi} &= (\frac{\kappa}{S/N_{min}})^{\frac{1}{\eta}} \\ &= (\frac{0.146}{0.284})^{\frac{1}{2.6}} \\ &\approx 0.774 \end{aligned} \tag{5.2}$$

5.1.3 UE Placement

5.1.3.1 Placement Strategies

For UE placements, we use two placement strategies,

1. Uniform

   UEs are uniformly distributed in the simulated cell.

2. Clustered

   We use cluster constraints to represent hot spots where UEs aggregate. Two types of constraints are used. With both types of clustered topologies, UEs are distributed inside a selected cluster constraint following a uniform distribution.

   (a) Circular clusters

   Circular clusters are used to represent hot spots with differing locations. We form k clusters with radius $(r_1, r_2, ..., r_k)$, and use different strategies to place the centers of the clusters inside the circle in a way that all clusters are contained in the simulated cell.

(b) Rectangular clusters

We use two rectangles to form a T-shaped cluster. It is used to represent hot spots that are aligned with some major roads in an urban area.

5.1.3.2 Placement Procedure

The UEs are placed into the clusters as follows. For each placement, we generate a random number s in [0, 1] and compare it with the values in a threshold table to decide into which cluster the UE should be placed. For example, Table 5.1 is an example of a threshold table.

Table 5.1: Threshold Table for UE Placement to Clusters

Cluster Index	Threshold
1	0.1
2	0.4
3	0.7
4	1

We show the placement procedure by way of example using Table 5.1. If $0 \leq s < 0.1$, we will assign the UE to $Cluster_1$; if $0.1 \leq s < 0.4$, we will assign the UE to $Cluster_2$; if $0.4 \leq s < 0.7$, we will assign the UE to $Cluster_3$; if $0.7 \leq s \leq 1$, we will assign the UE to $Cluster_4$.

Let p_k be the probability to place the UE to $Cluster_k$ at step k. The values in the thresh table can be seen as the cumulative probability of p_k, i.e. $\sum_{1 \ldots k} p_k$. Therefore, the last entry in the "Threshold" column of the table should always be 1.

5.2 Static/Dynamic Simulations

We use both static and dynamic simulations to evaluate the flow mapping systems.

5.2.1 Static Simulation

We first use a static simulation method which uses a completely new placement for all the UEs independent of the historical UE placements for each run. This is to represent a system that periodically runs a scheduling algorithm which potentially can change the associations of all the UEs. However, this method cannot capture the UEs' mobility and the *on-off* session dynamics. Therefore, we designed and implemented the second dynamic simulation method.

5.2.2 Dynamic Simulation

The dynamic simulation is event-driven which incorporates both the session *on-off* dynamics and UE mobility dynamics. The session *on-off* events consist of *on*-event and *off*-event. An *off*-event represents UE stopping receiving data or disassociating with the AP it is connected to. An *on*-event represents the resumption of one UE's data session which implies it must associate with an AP. A mobility event represents the movement of one UE. There can be other types of events in the system such as a timer event. Certain events can trigger the re-scheduling according to system requirements and design. For example, the re-scheduling can be triggered either by an *on*-event or a timer event.

In our simulation, we use a simplifying exit-and-replacement scheme to model the UE mobility. We further use a Bernoulli process to determine the number of *off-on* transitions before a UE exits the system and is replaced. The UE that resumes has a P_m probability of moving. For the lengths of the *on/off* sessions, we can use exponential or Pareto distributions to model them.

Since the system states change with sessions of varying lengths, we calculate the weighted average of the system performance metrics using the session lengths as weights.

5.3 Algorithms to Be Evaluated

We evaluate the following representative flow mapping algorithms in the three types of flow mapping systems.

1. Global Information based Flow Mapping Systems (GIFMS)
    - Optimal (opt)
    - ATOM (atom)
    - Global greedy (gg, global-greedy)
2. Local Information based Flow Mapping Systems (LIFMS)
    - Local greedy - WiFi preferred (lgw, local-greedy-wifi-preferred)
    - Local greedy - equal chance(lge, local-greedy-equal-chance)
    - Random (rand, random-assignment)
3. Semi-Global Information based Flow Mapping Systems (S-GIFMS)

- Load-aware local greedy - equal chance (llg, load-aware-local-greedy)

We introduce the details of each algorithm as follows,

1. Optimal (opt)

   Both [13] and [41] have proved the problem in Section 1.2 is an NP-hard problem. Therefore, we use a brute-force method to achieve the optimal solution. It iterates through all the possible user association configurations, and returns the one with the largest objective function value. We use the incremental evaluation technique introduced in Section 3.6 to speed up the GPF value evaluations. The time complexities of all the algorithms listed here except the random assignment depend on how we evaluate one association plan. A simple evaluation of an association plan without storing the results of a previous evaluation requires O(MN^2), where M is the number of APs and N the number of UEs. However, if we use the incremental evaluation as we described in Section 3.6, the time complexity of one evaluation for changing one UE from or to one AP are both O(1). In the following analysis, we will always assume the incremental evaluation is used. Note that the time complexity to achieve the optimal solution is $O(M^N)$ after using the incremental evaluation technique.

2. ATOM (atom)

   It is the Algorithm 1 in [33]. We show it in Algorithm 1 below. The algorithm determines the association plan for one WiFi AP at a time. We first define the terms and notions used in the algorithm similar to [33]. Let π be the set of undetermined WiFi APs and D the set of determined WiFi APs. Let $\mathbb{U}$ be the set of all UEs. As the convention, we use index j where $1 \leq j \leq M$ to denote the AP index. AP_1 is the LTE AP while AP_2 to AP_M are the WiFi APs. Let S_j be the set of UEs that can associate with AP_j; and L_j be the subset of S_j in which the UEs have not been assigned to any APs in D. Let A_j be the set of UEs that are assigned to AP_j. In the beginning, π is the set of all the WiFi APs. For L_j, where $2 \leq j \leq M$, each of them is initialized to S_j. A_1 is initialized to the set of UEs that cannot connect to any WiFi APs, i.e. $\mathbb{U}$ - $\cup_{j=2...M}(S_j)$. A_j where $2 \leq j \leq M$ are all initialized to an empty set.

   [33] uses the term "outer loop" and "inner loop" to name the two major conceptual loops in the algorithm. The outer loop tries to find the WiFi AP in the set π which can produce the largest utility function change based on its offloading plan (This utility function change can

be negative. But that situation rarely happens when the negative effect of adding UEs to LTE outweighs the positive effect of adding WiFi UEs to an empty AP.). The inner loop deals with the offloading plan evaluation for a WiFi AP. Similar to the greedy strategy in the outer loop, it moves the UE that can produce the largest utility function change from the LTE AP to the WiFi AP. However, it will stop if the largest one selected has a utility change that is not positive anymore. This stop condition ensures the "Moving-To-WiFi" operation for each WiFi AP can only increase the utility function.

Note a UE can have three states, i.e. {Unassigned, Assigned to LTE, Assigned to WiFi} in this algorithm. If one UE is assigned to WiFi, its plan is permanently fixed. It will be deleted from L_j for j $\in \pi$. However, if one UE is assigned to LTE, even though treated as a UE under the LTE AP when evaluating GPF, it may still appear in an L_j in a later inner loop and have a second chance to be moved to another WiFi AP, since it is not deleted from L_j for $j \in \pi$.

We try to analyze the time complexity of the ATOM algorithm as follows.

The time complexity of ATOM is more difficult to analyze compared to the other algorithms presented here. Because each UE may appear in multiple APs in π_j, while the size of π_j is changing over the outer loop iterations. Let denote the outer loop iteration number as t, and $m = M - 1$ as the number of WiFi APs. We know the algorithm always needs to run m iterations. Each iteration will set one WiFi AP's association plan. If we denote the size of π_j at iteration t as p_{jt}, evaluating the WiFi AP_j at iteration t requires $p_{jt} + (p_{jt} - 1) + ... + (p_{jt} - k_{jt})$, where k_{jt} is the number of UEs that needs to be tried before the stop condition is met in the inner loop. Or, we can express it as $\sum_{q=0...k_{jt}} p_{jt} - q$. Note p_{jt} changes over t. The only thing we know about p_{jt} is $0 < p_{jt} \leq$ N. So, we can only get a rough upper bound for this operation as O(kN). Additionally, at the beginning of each iteration, the algorithm moves all UEs in π_j to the LTE AP. That operation is O(p_{jt}) which is dominated by the other operations.

In the first iteration, the algorithm evaluates m APs. In the second iteration, since the first AP is set, it only evaluates m-1 APs. For the k_{th} iteration, it needs $m - k + 1$ AP evaluations. The last iteration only evaluates 1 AP. From this analysis, we find that it requires O(M^2) AP evaluations in total. So, overall the time complexity of the algorithm is O(kNM^2). A rough upper bound for k can be N. Therefore, the final worst-case time complexity for ATOM can be expressed as O(N^2M^2). If M is much smaller than N, it can also be expressed as O(N^2).

Algorithm 1: ATOM

```
   // Let B be the set of all the WiFi APs and 𝕌 the set of all UEs
   // Let A_j be the set of UEs that are assigned to AP_j
   // Let U_ij be the utility function change if UE_i is moved to AP_j
   // Initialization
 1 π = B
 2 initialize S_j based on effective rates
 3 A_1 = 𝕌 - ∪_{2...M}(S_j)
 4 L_j = S_j, ∀ j ∈ B
   // Outer loop
 5 while π ≠ φ do
 6 |   for WiFi AP_j ∈ π do
 7 |   |   A_j = φ;
 8 |   |   A_1j = A_1 ∩ L_j;
 9 |   |   T_j = L_j;
   |   |   // Inner loop
10 |   |   while T_j ≠ φ do
11 |   |   |   i* = argmax_i(ΔU_ij);
12 |   |   |   if ΔU_i*j ≤ 0 then
13 |   |   |   |   break;
14 |   |   |   end
15 |   |   |   ΔU_j = ΔU_j + ΔU_i*j;
16 |   |   |   A_j = A_j ∩ i*;
17 |   |   |   A_1j = A_1j − i*;
18 |   |   |   T_j = T_j − i*;
19 |   |   end
20 |   end
21 |   b = argmax_j(ΔU_j);
22 |   π = π - b;
23 |   L_j = S_j − A_b, ∀j ∈ B, j ≠ b;
24 |   A_1 = A_1b;
25 end
```

3. Global greedy (gg)

The algorithm is shown in Algorithm 2. First, there are two conceptual sets of UEs in the algorithm, the determined set $\mathbb{S}$ and the undetermined set $\mathbb{R}$. If the universal set of UEs is $\mathbb{U}$, $\mathbb{S} \cup \mathbb{R} = \mathbb{U}$ and $\mathbb{S} \cap \mathbb{R} = \emptyset$. The set $\mathbb{R}$ is initialized to the universal set of UEs while the set $\mathbb{S}$ an empty set. The algorithm runs in multiple iterations. In each iteration, it finds the (UE_i, AP_j) pair from $\mathbb{R}$ that can maximize the total objective function value of the UEs in $\mathbb{S}$, as if UE_i was connected to AP_j. Then it fixes the association of UE_i to AP_j, and moves UE_i from $\mathbb{R}$ to $\mathbb{S}$. The algorithm continues to run until $\mathbb{R}$ is empty. Note that it is possible that all the moves from $\mathbb{R}$ to $\mathbb{S}$ reduce the objective function value. In that case, the move that least reduces the objective function value is selected.

The time complexities of the algorithms depend on how we implement the plan evaluation part of the algorithm. If we use incremental evaluation as we described in Section 3.6, the time complexity of one evaluation for adding one UE to LTE AP is O(N) while that for WiFi AP is O(1). For each UE, it needs to try 1 LTE AP and M-1 WiFi APs. So, the time is O(M). We need to evaluate N, N-1, ..., 1 UEs in the outer loop, which is O(N^2) in total. So, overall the time complexity is O(N^2M). If M is much smaller than N, it can be expressed as O(N^2).

Algorithm 2: Global greedy heuristic.

```
1  ℝ = 𝔸
2  𝕊 = φ
3  currentT = 0
4  while not isEmpty(ℝ) do
5  |   maxDiff = -∞;
6  |   i' = 0;
7  |   j' = 0;
8  |   for i in ℝ do
9  |   |   S' = S ∪ i;
10 |   |   newT = Σ_{s∈S'} T_sj;
11 |   |   ΔT = Diff(currentT, newT);
12 |   |   if ΔT > maxDiff then
13 |   |   |   i' = i;
14 |   |   |   j'= j;
15 |   |   |   maxDiff = ΔT;
16 |   |   end
17 |   end
18 |   𝕊 = 𝕊 ∪ i';
19 |   ℝ = ℝ - i';
20 |   x_{i'j'} = 1;
21 end
```

4. Local greedy - WiFi preferred (lgw)

   Each UE selects the WiFi AP with the highest effective rate. If no WiFi available, it will select the highest-effective-rate LTE BS. In the case of one LTE in the simulated cell, the second phase becomes trivial. The time complexity for this algorithm to run for one UE is O(M), since it needs to compare all the APs it is accessible and the worst case is it can connect to all the APs. If we sum up the time complexity of all the UEs in the system so that it is comparable to the GIFMS algorithms, the time complexity will be O(MN).

5. Local greedy - equal chance (lge)

   Each UE selects the interface that can generate the highest effective throughput based on the connection status seen locally. This strategy is commonly used in the existing HetNets, but we will show it performs poorly with respect to generalized proportional fairness and aggregate throughput. This algorithm has the same time complexity as lgw, i.e. O(M) for one UE and O(MN) for all the UEs in the system.

6. Random assignment (rand)

   Each UE randomly selects from all its accessible APs. AP_j is accessible by UE_i if the effective throughput from AP_j to UE_i is greater than 0. In all the local greedy algorithms, ties are broken by random selection among the tie set. The time complexity of this algorithm is O(1) for one UE, and O(N) for all the UEs.

7. Load-aware local greedy - equal chance (llg)

   This is an algorithm similar to the Greedy-0 algorithm in [13]. It is fundamentally different from the GIFMS in that it is only triggered by either *off-on* events or mobility events, instead of periodically run for all the UEs. In both types of events, only one flow will try to connect to an AP. We call this flow the *coming flow*. The flow mapping decision to make is only for the coming flow. In other words, it has an additional constraint of "no moving of existing flows" comparing with the GIFMS. This constraint totally eliminates the enforced handovers when running GIFMS which remaps all the flows. This is one of the advantages of S-GIFMS over GIFMS.

   Even though the algorithm is similar to the one in [13], we further detail how this algorithm can be implemented with minimum additional information monitored and broadcasted at the APs. We classify llg to S-GIFMS in the introduction distinguished from GIFMS such as ATOM and global-greedy since it only requires information from the AP level as opposed to the information from all the UEs in the global system. It uses the incremental GPF evaluation procedures we described in Section 3.6.1 and Section 3.6.3 to evaluate the GPF deltas for PF and max-min schedules APs respectively. As we analyzed in those sections, the AP level information required for a PF scheduled AP is only the number of UEs connected to the AP. A max-min fair scheduled AP further entails one more piece of information compared, which is the round time of all the connected UEs. Section 8.4 discusses more details of how those AP level metrics can be monitored at APs and broadcasted to UEs for various RATs.

   The mapping algorithm evaluates the GPF deltas of adding the coming flow to each AP that is accessible by the coming flow. It then greedily picks the AP that produces the largest GPF delta, similar to the local greedy flow mapping algorithms we have introduced. However, comparing with the local greedy ones, the decision-making is augmented with the AP-level information mentioned above.

   This algorithm has the same time complexity as lge, i.e. O(M) for one flow.

Chapter 6

Evaluations using Static Simulations

In this chapter, we first use static simulation to evaluate the performance of the flow mapping systems under various settings.

6.1 Comparison with the Optimal Solution in Smaller Scale Simulations

We first simulate the flow mapping systems using the methodology in Chapter 5 with the following parameters,

- No. of UEs (N): 12
- No. of APs (M): 4
- No. of runs: 16384
- UE placement strategy: uniform (as explained in Section 5.1.3)
- LTE radius: 1 (equivalent to $\kappa = 0.146$ as discussed in Section 5.1.2)
- κ value (in Eq. (4.1)) of WiFi APs: the same as that of the LTE AP

We only use N=12 when comparing the optimal and the other flow mapping systems because of the high time complexity to generate an optimal solution as analyzed in Section 5.3. We show results with more UEs in Section 6.2.

6.1.1 Aggregate Results

Among all the metrics in Chapter 3, GPF value and Jain's fairness are the more important ones. Because GPF is the optimization objective in all the global flow mapping algorithms while Jain's fairness is just another way to express the distance between the flow-level throughput results of one algorithm and the optimal solution. As we have analyzed in Section 3.2, GPF has a better balance between spectrum efficiency and fairness among flows. Therefore, we will always list GPF value first (For Jain's fairness metric, we can only show the comparison when the optimal solution is computed).

6.1.1.1 GPF Value and Jain's Fairness

1. GPF value

Table 6.1 to 6.3 show the performance of GPF for each algorithm. Table 6.1 lists the raw values while the other two tables further display the relative performance comparing the local-greedy-equal-chance and the optimal solution respectively. Relative difference results, as the one in Table 6.2, are useful for comparison among different algorithms in one scenario. They can also show the relative performance comparison across scenarios when parameters such as the number of UEs or coverage of APs can boost/degrade the performance of all the algorithms. We will show the results of the other metrics in a similar way for the following sections. The relative difference rate D_r is calculated as follows,

$$D_r = \frac{other\_method\_result - baseline\_result}{(|other\_method\_result| + |baseline\_result|)/2} \tag{6.1}$$

The reason we use the above form of relative difference is the rudimentary form of

$$\frac{other\_method\_result - baseline\_result}{baseline\_result}$$

can have problems when *baseline_result* is negative or 0. In our evaluation, the GPF value of

baseline algorithms may become negative. Note the resulting D_r values are bounded to [-2, 2]. If the two results in comparison have different signs or one of them is 0, the results will be $\pm$ 2. We will notice that in the later sections.

Table 6.2 shows the comparison using the local-greedy-equal-chance as the baseline algorithm. Table 6.3 shows the comparison using the optimal as the baseline algorithm.

From the tables, we can see that the random-assignment has the worst performance. It can only achieve about half of the mean GPF value achieved by the ATOM and global-greedy. The local-greedy-wifi-preferred, which is the strategy used in current systems, is only slightly better than random assignment (14.29 compared with 10.57). The local-greedy-equal-chance has much better performance than the local-greedy-wifi-preferred and random assignment. However, it is still slightly worse than ATOM and global-greedy. ATOM and global-greedy have the best GPF performance, which is very close to the optimal solution. However, the difference of the mean values between the local-greedy-equal-chance and the three GIFMS algorithms is very small. With a magnitude of around 20, the GPF difference is only around 1.

Table 6.1: GPF Value [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.v	14.29	2.41	14.32	2.61	24.13
lge.v	18.92	2.92	19.00	5.75	28.96
atom.v	19.73	2.56	19.77	7.91	28.96
gg.v	19.62	2.62	19.67	8.38	28.96
rand.v	10.57	4.05	10.87	-8.47	24.14
opt.v	19.82	2.53	19.85	9.29	28.96

Table 6.2: GPF Value Compared with lge [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.v	-0.28	-0.19	-0.28	-0.75	-0.18
atom.v	0.04	-0.13	0.04	0.31	0.00
gg.v	0.04	-0.11	0.03	0.37	0.00
rand.v	-0.57	0.33	-0.54	-2.00	-0.18
opt.v	0.05	-0.14	0.04	0.47	0.00

Table 6.3: GPF Value Compared with opt [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.v	-0.32	-0.05	-0.32	-1.12	-0.18
lge.v	-0.05	0.14	-0.04	-0.47	0.00
atom.v	-0.00	0.01	-0.00	-0.16	0.00
gg.v	-0.01	0.04	-0.01	-0.10	0.00
rand.v	-0.61	0.46	-0.58	-2.00	-0.18

From Table 6.1, we also notice, as for the standard deviations, all the algorithms except the random-assignment have similar performance. By further looking at the min and max values in Table 6.1, we see that the min value of random-assignment is much lower than the other algorithms while the max values being similar. This means that the non-random algorithms do better than rand-assignment in avoiding bad solutions. Meantime, the mean and median of each algorithm are very close. This implies that the GPF results of each algorithm over all the runs have a symmetrical distribution. The same applies to the means and medians of the other metrics below.

Fig. 6.1 shows the cumulative distribution function (CDF) of the GPF values achieved by all the algorithms over the 16K runs. Fig. 6.2 further uses a time series diagram to show the GPF value comparison in the first 256 runs. From Fig. 6.1, we observe the same performance ranking as we observed in the tables. For all CDF figures, we add two horizontal dashed lines, which help to show the 5% and 95% percentiles of the plotted results. The dotted horizontal grid line at y=0.5 further shows the medians of the results. We see that the performance of ATOM, global-greedy and local-greedy-equal-chance are apparently much better than the other two. The performance of local-greedy-equal-chance is somewhat worse than ATOM and global-greedy. However, we should note this is the result of uniform UE placement. We know the APs are located symmetrically inside the simulated cell as we described in Section 5.1.1. If UEs all greedily select the AP based on effective throughput under the circumstance of uniform UE placement, this tends to equalize the AP loads while trying to maximize the total throughput given the only available local information. We will show the results of local-greedy-equal-chance under various UE placement strategies with clusters in Section 6.4.

From Fig. 6.2 we can see similar performance ranking among the algorithms. But it further shows the noise or deviations in separate runs. For example, we can see, around the run sequence number 110, global-greedy achieves a GPF value that is even worse than that of the local-greedy-wifi-preferred. It shows even if one algorithm owns a better average performance result, it can still occasionally have a lower performance metric than another algorithm with an inferior average. From this type of time series graph, we can also more directly observe the standard deviation difference among algorithms.

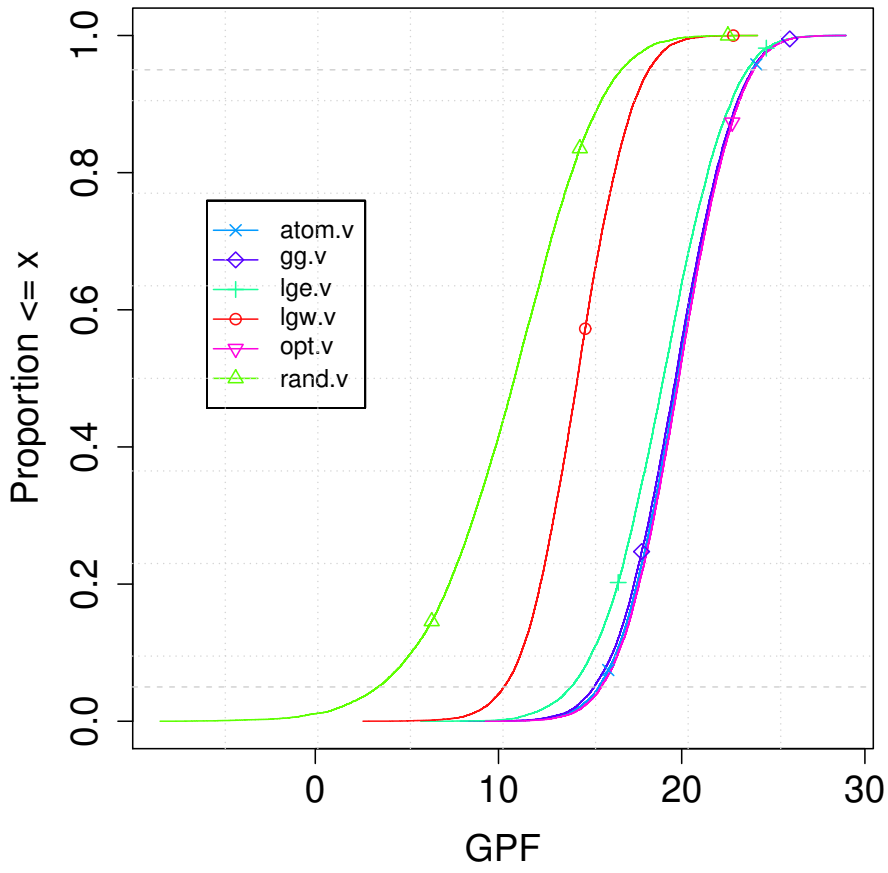

Figure 6.1: CDF of GPF values over multiple runs.

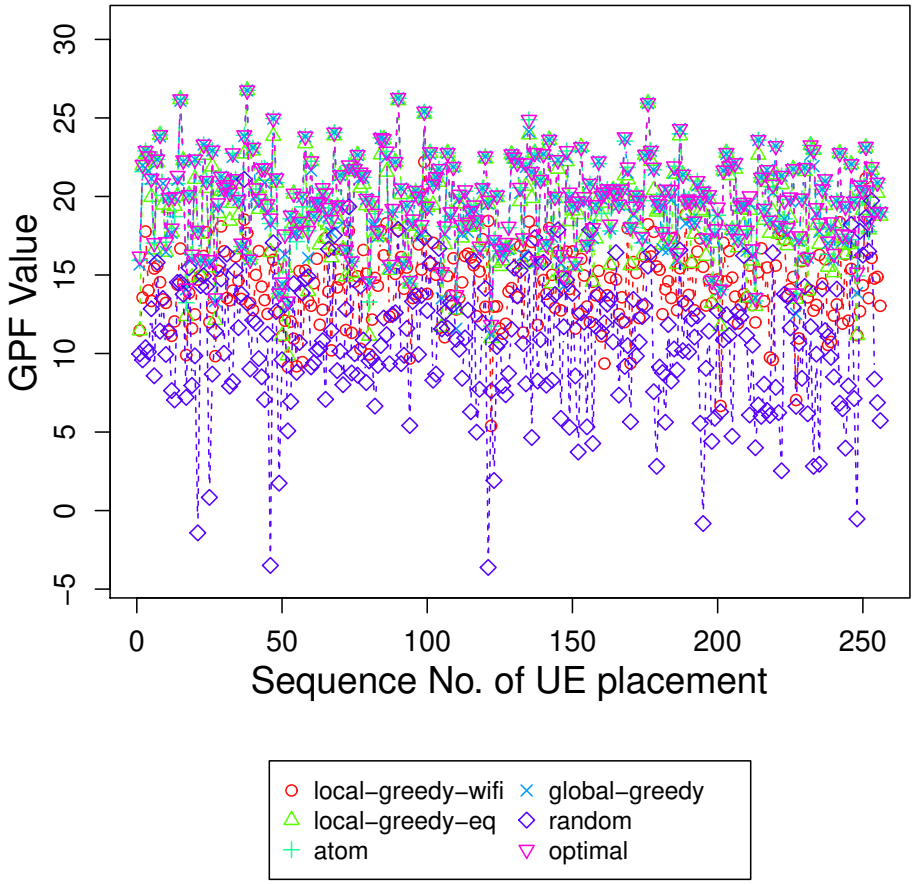

Figure 6.2: Performance of the first 256 runs.

2. Jain's Fairness Index (JFI)

The Jain's fairness index can be used to measure the flow-level throughput difference/distance as achieved by one mapping algorithm when compared with the optimal mapping. Even though we can also calculate relative differences in the mean throughputs of algorithms differences as in Table 6.9, the differences in the results of individual flows may cancel out when calculating the mean values. By using quadratic functions, the JFI better captures the distance between solutions with flow-level difference details.

Interestingly, the JFI results are consistent with the GPF value results after scaling to [0, 1]. The random-assignment is still about half of the optimal solution, ATOM and global-greedy are still the best ones. We can see that ATOM is very close to the optimal solution in this case as its median of the Jain's fairness for 16K runs is 1. From Table 6.4, we observe that ATOM achieves an average JFI of 0.96 while global-greedy achieves 0.92, which are both very close to 1. local-greedy-wifi-preferred achieves 0.72 which is similar to the scale of distance from the optimal as in the GPF comparison. Fig. 6.3 further shows the CDF of JFI for various algorithms. We can see more than 70% of ATOM results are the same as the optimal. Global-greedy only has about half of the solution the same as the optimal. Local-greedy-equal-chance only has about 20% in this case. We list the relative difference compared with the local-greedy-equal-chance and the optimal in Table 6.5 and Table 6.6 respectively. We will use them to compare with the similar tables when the parameters like the number of UEs changed. Note the entries in the standard deviation column of Table 6.6 are all with a value of 2. This is caused by the *baseline_result* in Eq. (6.1) being 0.

Table 6.4: Jain's Fairness Index [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.nj	0.72	0.16	0.75	0.15	1.00
lge.nj	0.81	0.16	0.85	0.22	1.00
atom.nj	0.96	0.10	1.00	0.24	1.00
gg.nj	0.92	0.12	0.97	0.22	1.00
rand.nj	0.53	0.17	0.53	0.12	1.00
opt.nj	1.00	0.00	1.00	1.00	1.00

Table 6.5: Jain's Fairness Index Compared with lge [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.nj	-0.12	0.01	-0.12	-0.36	0.00
atom.nj	0.16	-0.48	0.17	0.06	0.00
gg.nj	0.12	-0.30	0.13	-0.01	0.00
rand.nj	-0.42	0.08	-0.45	-0.62	0.00
opt.nj	0.21	-2.00	0.17	1.27	0.00

Table 6.6: Jain's Fairness Index Compared with opt [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.nj	-0.33	2.00	-0.29	-1.46	0.00
lge.nj	-0.21	2.00	-0.17	-1.27	0.00
atom.nj	-0.05	2.00	0.00	-1.24	0.00
gg.nj	-0.08	2.00	-0.03	-1.28	0.00
rand.nj	-0.61	2.00	-0.61	-1.58	0.00

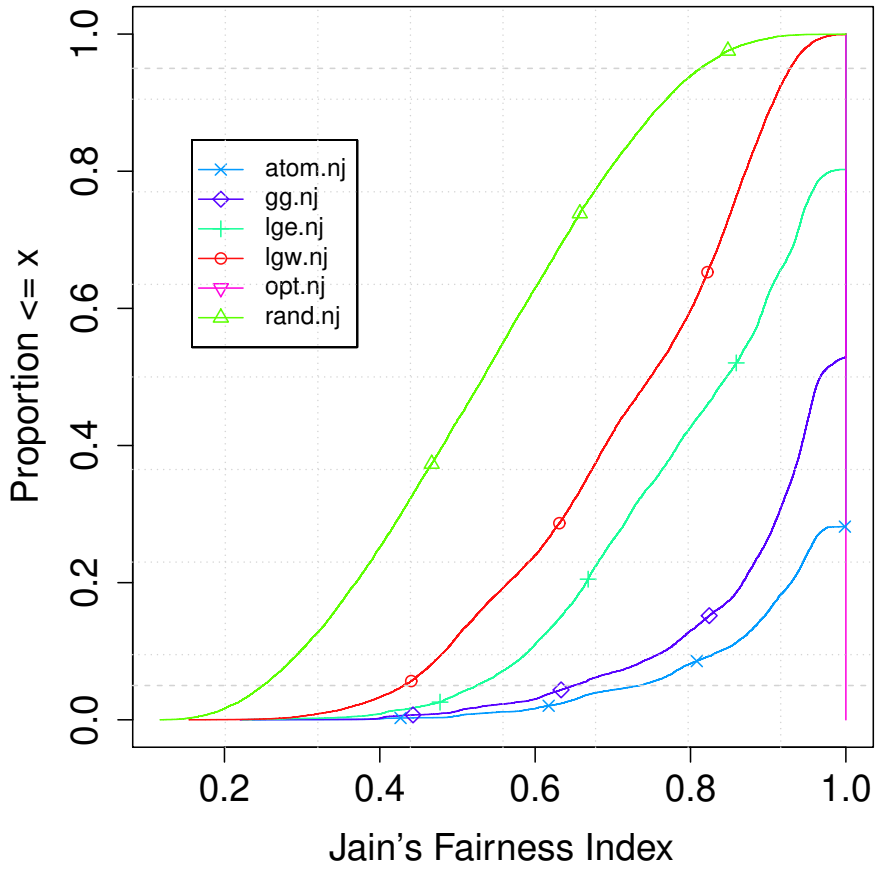

Figure 6.3: CDF of Jain's Fairness Index values over multiple runs.

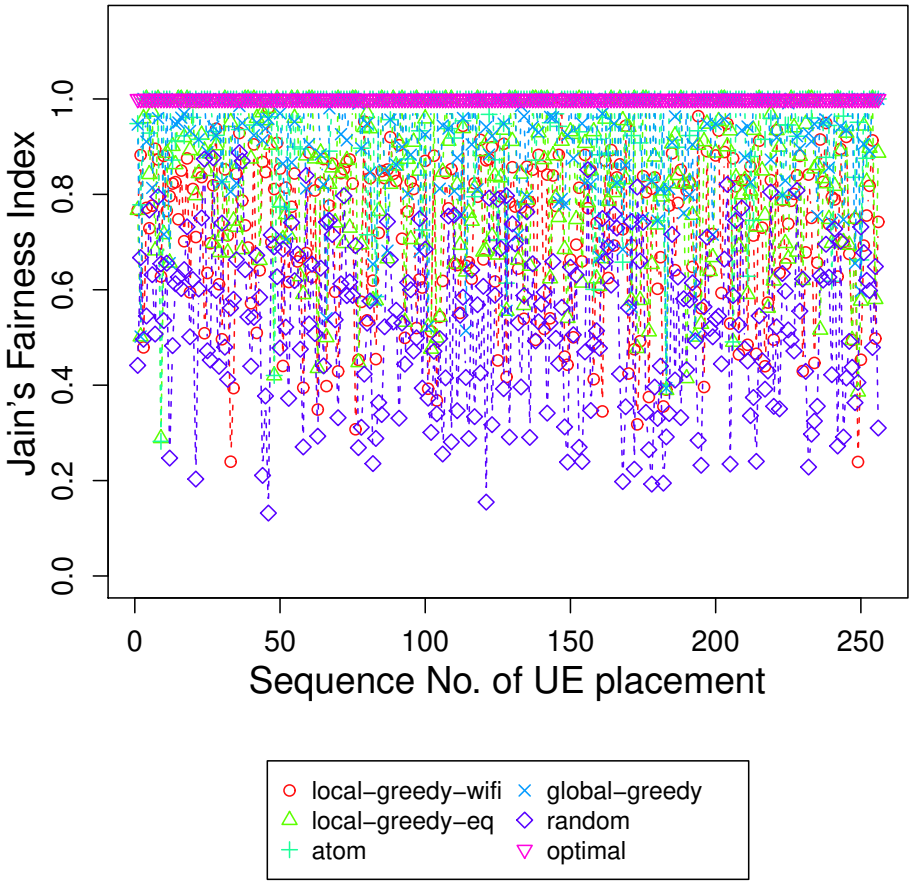

Figure 6.4: Performance of the first 256 runs.

6.1.1.2 Other Metrics

1. Aggregate Throughput

Maximizing aggregate throughput is not a desirable objective function because it can produce extreme unfairness including starvation. Nevertheless, it remains a useful metric to be considered.

Table 6.7 to Table 6.9 shows the results of aggregate throughput for all the algorithms. Similar to the GPF results, local-greedy-wifi-preferred and random-assignment are still much worse than the other four methods. It is interesting to observe that the mean of random-assignment is even higher than the local-greedy-wifi-preferred. However, the result of the random-assignment has a much higher standard deviation.

The aggregate throughput difference between local-greedy-equal-chance and the algorithms of the GIFMS, i.e. ATOM and global-greedy is so small (as shown in Table 6.2) that we would argue that the difference may not make any sensible difference to users in practical deployments under this setting.

Table 6.7: Aggregate Throughput (Mbps) [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.t	46.11	10.20	45.02	18.28	94.32
lge.t	78.16	20.01	76.99	23.14	155.77
atom.t	78.50	16.55	77.62	29.52	150.48
gg.t	79.59	16.80	78.83	31.92	155.77
rand.t	50.43	15.88	48.77	10.03	121.25
opt.t	78.58	16.46	77.68	31.78	150.48

Table 6.8: Aggregate Throughput (Mbps) Compared with lge [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.t	-0.52	-0.65	-0.52	-0.23	-0.49
atom.t	0.00	-0.19	0.01	0.24	-0.03
gg.t	0.02	-0.17	0.02	0.32	0.00
rand.t	-0.43	-0.23	-0.45	-0.79	-0.25
opt.t	0.01	-0.19	0.01	0.31	-0.03

Table 6.9: Aggregate Throughput (Mbps) Compared with opt [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.t	-0.52	-0.47	-0.53	-0.54	-0.46
lge.t	-0.01	0.19	-0.01	-0.31	0.03
atom.t	-0.00	0.01	-0.00	-0.07	0.00
gg.t	0.01	0.02	0.01	0.00	0.03
rand.t	-0.44	-0.04	-0.46	-1.04	-0.22

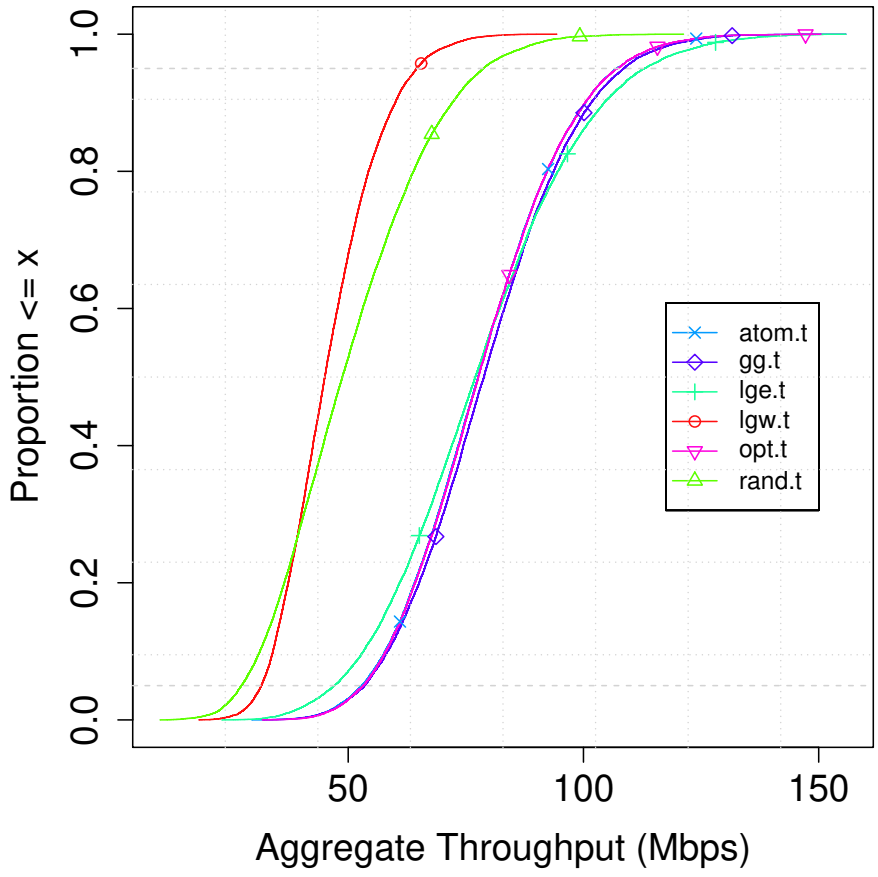

Figure 6.5: CDF of aggregate throughput values over multiple runs.

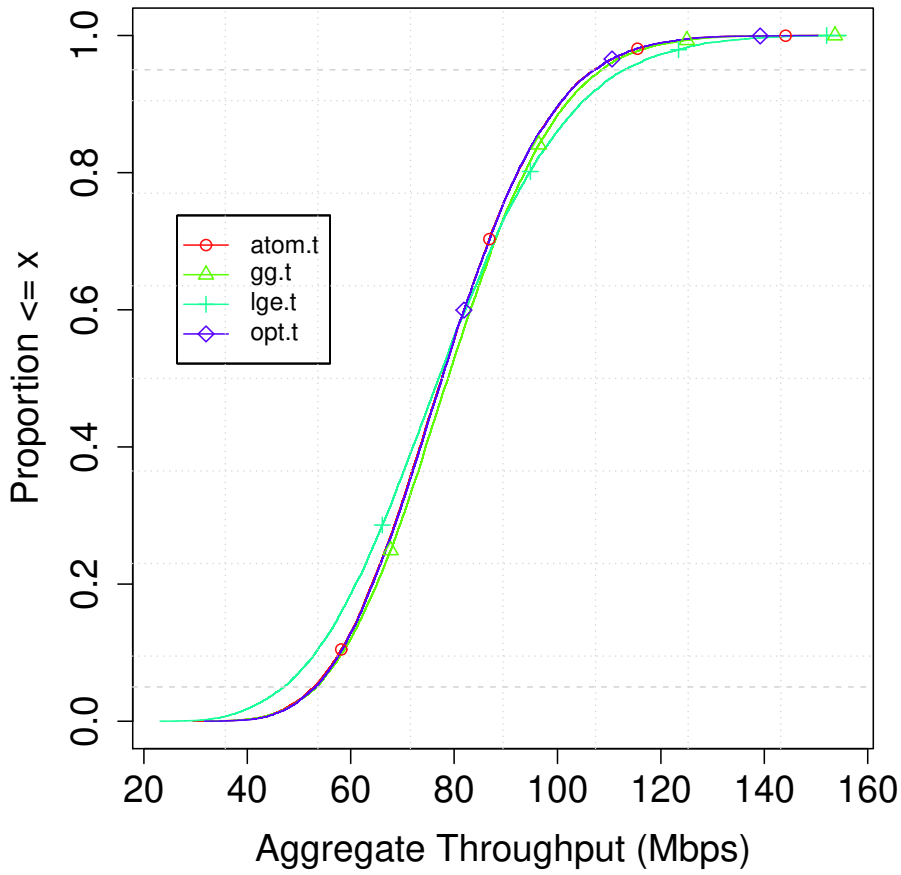

Figure 6.6: CDF of aggregate throughput values over multiple runs.

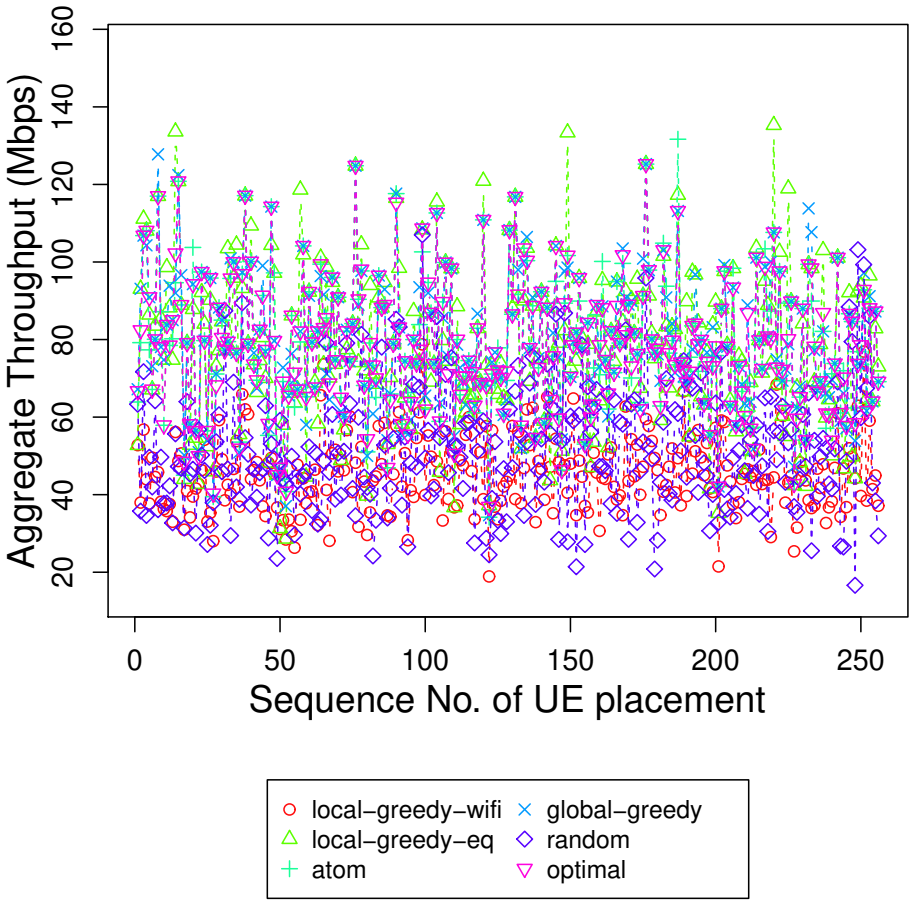

Figure 6.7: Performance of the first 256 runs.

2. Throughput Fairness Index (TFI)

The TFI ranking of algorithms is different from that of GPF and aggregate throughput. This is because it only measures the distance of a throughput vector to a throughput vector that has all equal elements independent of the magnitude of the elements. We notice local-greedy-wifi-preferred has the highest TFI. After further investigating the runs that local-greedy-wifi-preferred produces the top 5 highest TFI values, we find the reason is as follows. Each WiFi AP gives equal throughput to UEs connected to it. If all the UEs connect to the same AP, the system will certainly achieve a TFI of 1. However, even if the UEs are connected to multiple WiFi APs, as long as they are almost equally divided into the WiFi APs, they will receive very similar throughput too. Fig. 6.8 and Fig. 6.9 show the two runs that generate the highest two TFIs for the local-greedy-wifi-preferred. As we can see, they are the cases when UEs are almost evenly divided into two or three APs. We could expect that, as the number of UEs increases, they will be more likely to be more evenly distributed to the WiFi APs which leads to a better JFI. This will be verified in the experiment result using 32 UEs in Section 6.2.2.

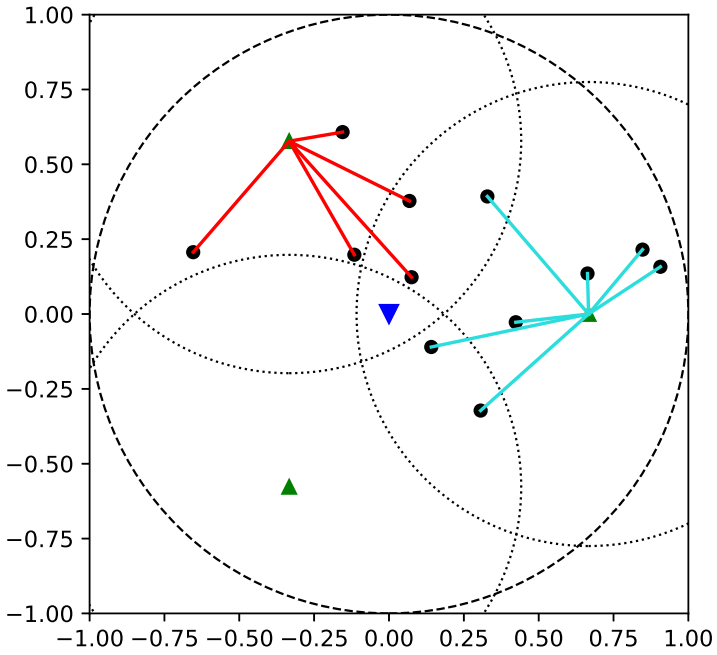

Figure 6.8: The run that lgw produces the highest Throughput Fairness Index.

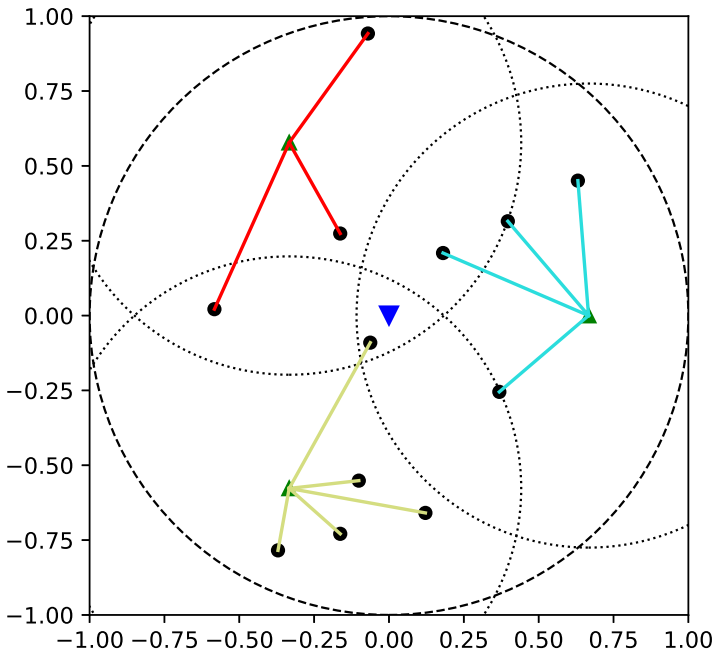

Figure 6.9: The run that lgw produces the highest Throughput Fairness Index.

Table 6.10: Throughput Fairness Index [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.TFI	0.78	0.19	0.84	0.17	1.00
lge.TFI	0.63	0.18	0.64	0.14	1.00
atom.TFI	0.71	0.14	0.73	0.22	1.00
gg.TFI	0.69	0.14	0.71	0.18	1.00
rand.TFI	0.54	0.18	0.54	0.11	1.00
opt.TFI	0.72	0.14	0.73	0.22	1.00

Table 6.11: Throughput Fairness Index Compared with lge [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.TFI	0.22	0.04	0.28	0.22	0.00
atom.TFI	0.12	-0.24	0.13	0.44	-0.00
gg.TFI	0.10	-0.24	0.10	0.28	-0.00
rand.TFI	-0.15	0.02	-0.16	-0.26	-0.00
opt.TFI	0.14	-0.27	0.14	0.46	-0.00

Table 6.12: Throughput Fairness Index Compared with opt [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.TFI	0.09	0.31	0.13	-0.25	0.00
lge.TFI	-0.14	0.27	-0.14	-0.46	0.00
atom.TFI	-0.01	0.03	-0.01	-0.02	-0.00
gg.TFI	-0.04	0.04	-0.04	-0.19	0.00
rand.TFI	-0.29	0.29	-0.30	-0.70	0.00

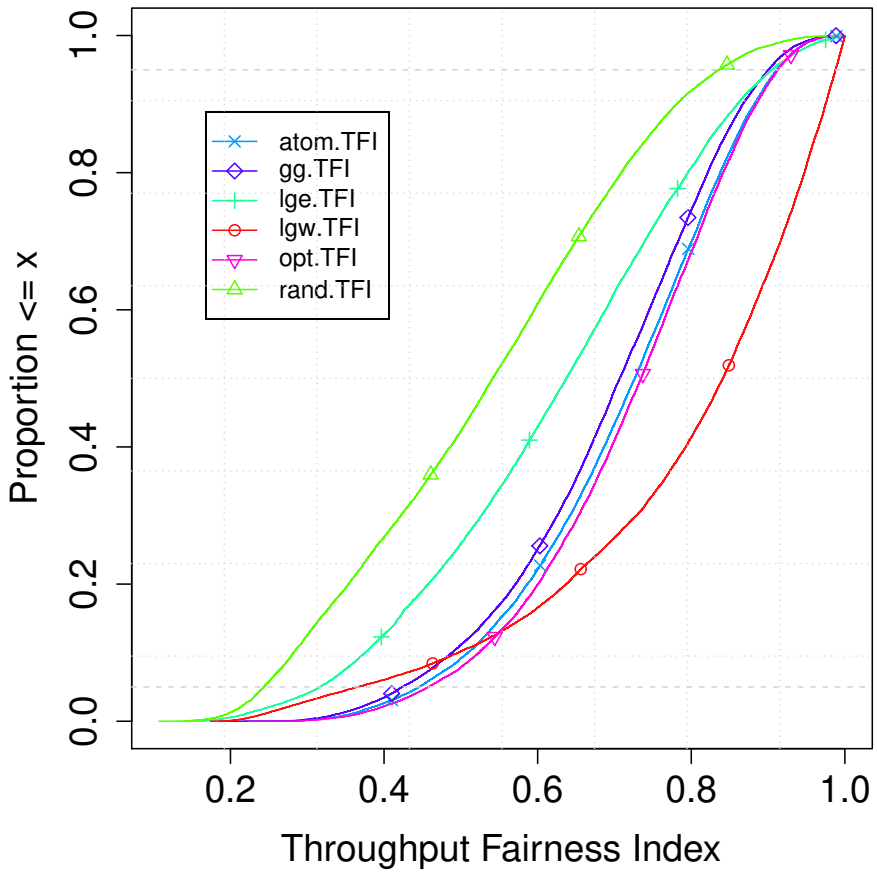

Figure 6.10: CDF of Throughput Fairness Index over multiple runs.

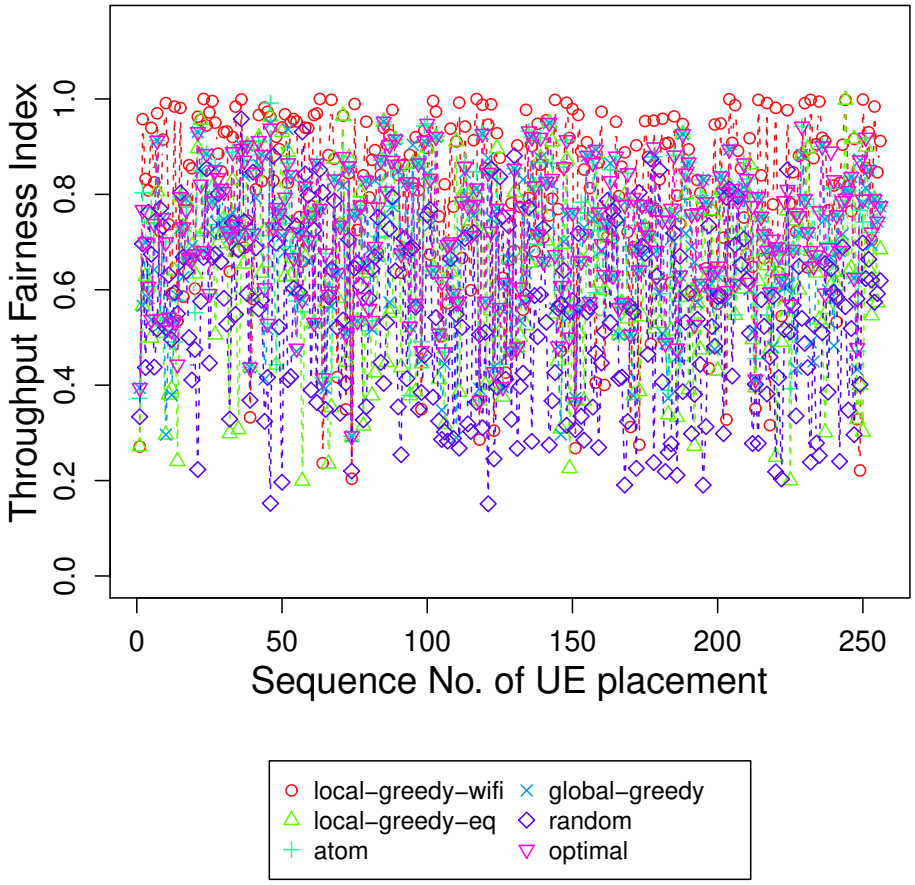

Figure 6.11: Performance of the first 256 runs.

3. Min-Max Fairness

Min-max fairness can be considered as another form of the distance measurement between one solution's throughput vector to that of an equal-throughput vector. However, comparing with the TFI, it only takes into consideration the ratio of the minimum to the maximum, ignoring the detailed difference of the values in the middle. As we can see from Table 6.13 to Table 6.15, the local-greedy-wifi-preferred still has the highest mean in this metric, similar to the ranking in TFI. It has the same reason as local-greedy-wifi-preferred's high TFI value. Basically, it is the combined effect of the following two reasons: 1) with uniform UE placement, it is highly likely to almost evenly divide UEs to WiFi APs when APs are located symmetrically; 2) WiFi APs are throughput fair which will make the throughputs of UEs under each WiFi AP the same. The combined effect of the two is that all the UEs have similar throughput values, therefore a high min-max fairness value. Fig. 6.12 shows the CDF of this metric for all the algorithms. We see that the local-greedy-wifi-preferred has a higher min-max fairness value overall. The other algorithms have similar performance in general. Fig. 6.13 shows the time series of min-max for all the algorithms in the first 256 runs. From it, we can see, even though local-greedy-wifi-preferred has higher min-max fair values, those values have a large standard deviation. This can also be observed from the standard deviation numbers in the tables.

Since it shows similar information as the TFI and may produce invalid results when the maximum value is 0, we will only show the TFI results in the experiments after this one.

Table 6.13: Min-Max Fairness [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.mm	0.38	0.21	0.36	0.02	1.00
lge.mm	0.15	0.12	0.12	0.01	0.98
atom.mm	0.14	0.10	0.11	0.01	0.83
gg.mm	0.12	0.10	0.09	0.01	0.85
rand.mm	0.07	0.06	0.05	0.01	0.93
opt.mm	0.14	0.10	0.11	0.01	0.84

Table 6.14: Min-Max Fairness Compared with lge [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.mm	0.85	0.52	1.00	0.91	0.02
atom.mm	-0.11	-0.18	-0.09	0.34	-0.17
gg.mm	-0.22	-0.23	-0.24	0.29	-0.14
rand.mm	-0.81	-0.62	-0.89	-0.32	-0.05
opt.mm	-0.10	-0.18	-0.08	0.34	-0.16

Table 6.15: Min-Max Fairness Compared with opt [Compare-With-Opt]()

	mean	sd	median	min	max
lgw.mm	0.94	0.68	1.07	0.62	0.17
lge.mm	0.10	0.18	0.08	-0.34	0.16
atom.mm	-0.00	-0.00	-0.00	0.00	-0.01
gg.mm	-0.12	-0.06	-0.16	-0.05	0.02
rand.mm	-0.72	-0.45	-0.82	-0.64	0.10

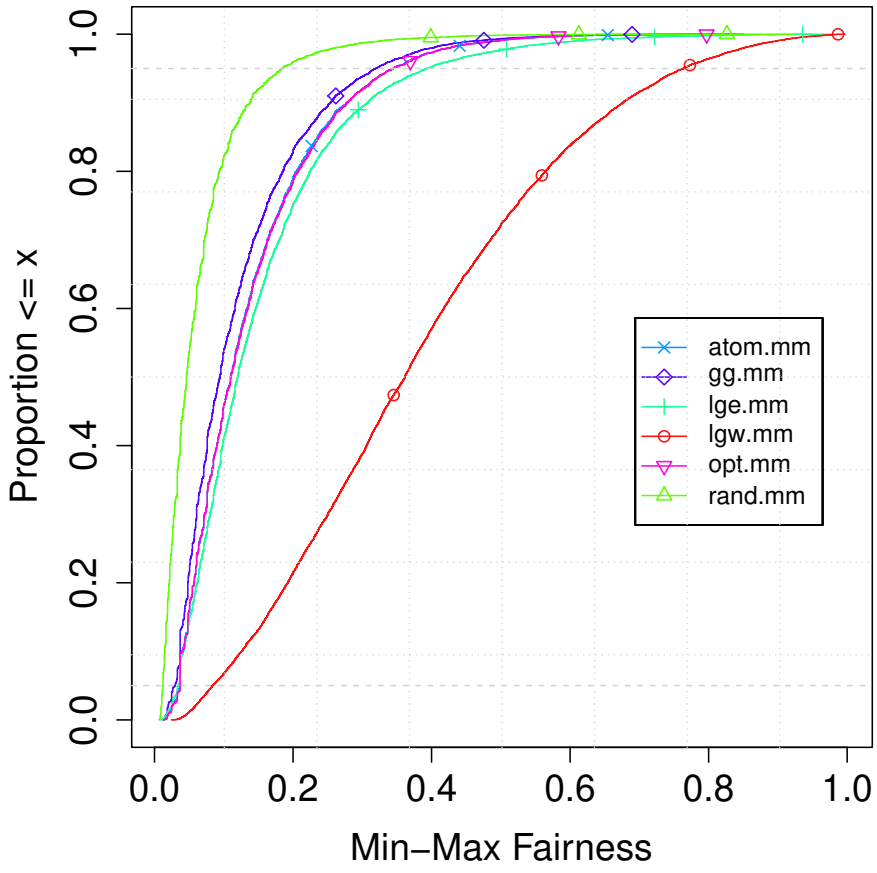

Figure 6.12: CDF of Throughput Fairness Index over multiple runs.

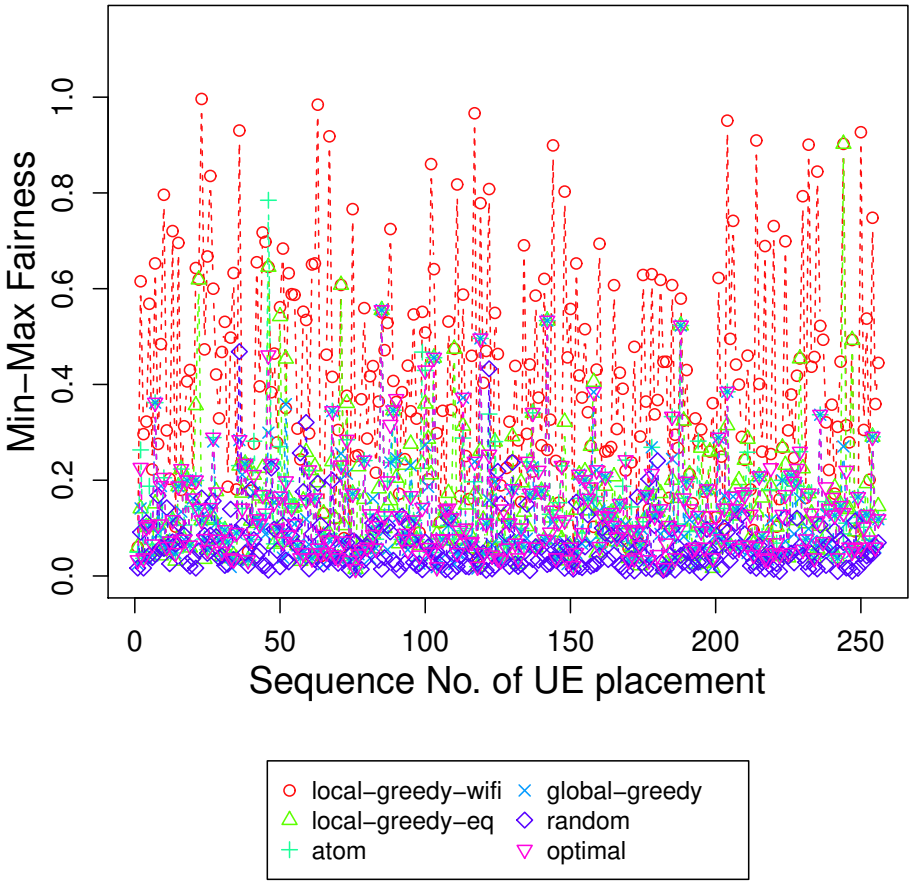

Figure 6.13: Performance of the first 256 runs.

6.1.2 Flow Level Results

We use the following methodology to collect flow level performance results. For each flow, we measure the percentage of runs among the 16384 runs that the throughput achieved by an algorithm is larger than/equal to/less than that achieved by a baseline algorithm. Then, we calculate the statistics of this percentage metric over all the flows. We only show the average of this metric here since the standard deviations are all under 0.01. These low standard deviations show 16K runs is enough to make each flow statistically the same over all the runs.

We use two baseline algorithms, i.e. local-greedy-equal-chance and local-greedy-wifi-preferred.

Table 6.16 shows the flow-level comparison result over the local-greedy-equal-chance while Table 6.17 displays that for the local-greedy-wifi-preferred. For example, in Table 6.16, the column $P_{greater}$ shows, the average percentage of runs in which the resulting throughput of one flow by one algorithm is larger than that by the local-greedy-equal-chance. $P_{greater}$ and P_{less} show the average percentages when that statistic of one algorithm is equal to and less than that of local-greedy-equal-chance respectively. From this table, we can perceive the small difference between local-greedy-equal-chance and the algorithms in the GIFMS category from another perspective. We see, for each flow, it experiences the same throughput about half of the runs/time. The algorithms in GIFMS only show a slight advantage with about 33% versus 17% in the $P_{greater}$ and P_{less} respectively.

However, from Table 6.17, the algorithms in GIFMS show a much larger advantage over local-greedy-wifi-preferred. We see $P_{greater}$, P_{equal} and P_{less} are about (70%, 20%, 10%) respectively.

Table 6.16: Flow Level Comparison over lge

	$P_{greater}(\%)$	$P_{equal}(\%)$	$P_{less}(\%)$
lgw	7.49	33.10	59.41
lge	0.00	100.00	0.00
atom	33.26	48.89	17.85
gg	32.86	46.41	20.73
rand	27.26	5.63	67.11
opt	33.18	49.64	17.18

Table 6.17: Flow Level Comparison over lgw

	$P_{greater}(\%)$	$P_{equal}(\%)$	$P_{less}(\%)$
lgw	0.00	100.00	0.00
lge	59.41	33.10	7.49
atom	69.58	20.29	10.14
gg	69.16	18.84	12.00
rand	44.82	4.45	50.74
opt	69.77	20.32	9.91

6.2 Baseline Evaluation Without the Optimal

In this section, we evaluate the algorithms with more UEs without running the optimal. We use the same system parameters as those in the last section except changing the number of UEs to 32.

6.2.1 GPF Value

Table 6.18 and Table 6.19 show the raw values of GPF for all the algorithms and their relative differences compared with the local-greedy-equal-chance. The first thing we observe is the algorithms in the leading group (i.e. ATOM, global-greedy, local-greedy-equal-chance) still have the best and similar GPF values around 20, as in the 12-flow baseline result in Table 6.1. However, comparing with the results in Table 6.1, the GPF values of local-greedy-wifi-preferred and random-assignment have a much larger decrease. Random-assignment decreases 11.69 (10.57 $\rightarrow$ -1.12) while local-greedy-wifi-preferred decreases 6.13 (14.29 $\rightarrow$ 8.16). The reason for random-assignment's significant decrease in GPF value is as follows. As the number of UE increases, the problem space grows exponentially. It will be more and more difficult for the random-assignment to "accidentally" obtain a correct solution. From Table 6.1, we also observe a rapid decrease in the GPF value of local-greedy-wifi-preferred. It is because lgw prefers WiFi AP whenever they are available. Therefore, as the number of UE increases, it may produce more mappings that lead to congested WiFi APs. This degrades the performance of all the flows under those congested APs, and also the aggregate performance metrics.

From Table 6.19, we can see that the GPF value of local-greedy-equal-chance is still very similar to that of ATOM and global-greedy as in the 12-flow baseline. Comparing with the relative difference table of the 12-flow experiment in Table 6.2, we see that the values for ATOM and global-greedy only change slightly. This means, if the UEs are distributed uniformly, the local-greedy-

equal-chance will have similar performance as ATOM and global-greedy no matter how many UEs in the scenario.

Table 6.18: GPF Value [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.93	4.25	21.93	6.53	40.30
gg.v	21.74	4.34	21.72	5.01	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

Table 6.19: GPF Value Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.v	-0.88	-0.19	-0.88	-2.00	-0.56
atom.v	0.05	-0.04	0.05	0.44	0.00
gg.v	0.04	-0.02	0.04	0.18	0.00
rand.v	-2.00	0.33	-2.00	-2.00	-0.61

Fig. 6.14 shows the CDF of the GPF values. We can clearly see the curves for random-assignment and local-greedy-wifi-preferred have shifted towards the left to a large extent while the other three algorithms remain almost the same comparing with those in Fig. 6.1. The performance of the random assignment is very bad with most of its curve on the negative side of the x-axis. Fig. 6.15 further shows the time series graph of the GPF values. We can clearly see three layers in this graph comparing with Fig. 6.2. ATOM, gg and lge form the first layer. lgw is the second layer while random-assignment is the last layer. This is mainly because of the GPF performance degradation of random-assignment and local-greedy-wifi-preferred compared to their performance in the 12-flow experiment result.

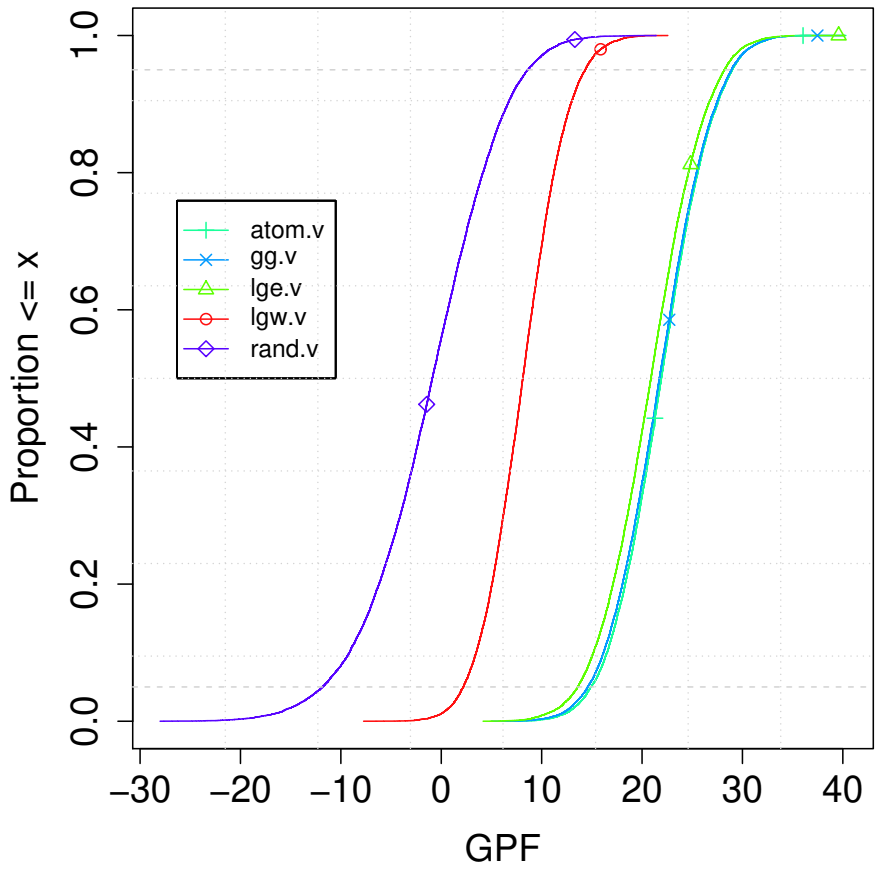

Figure 6.14: CDF of GPF value over multiple runs.

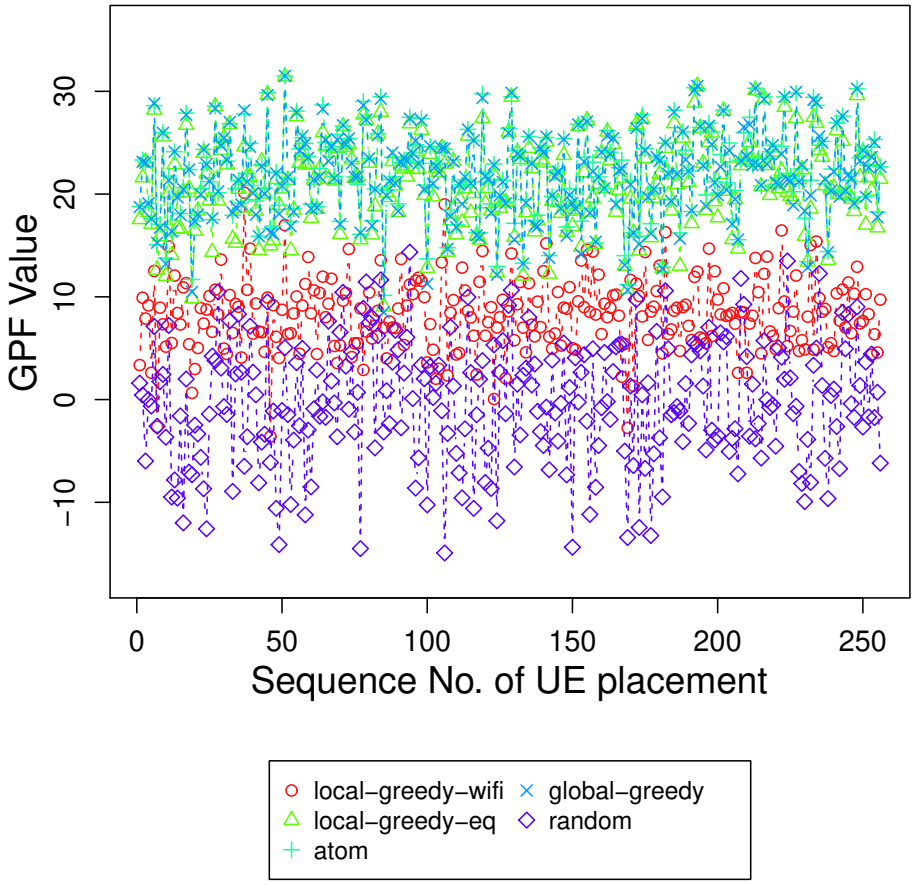

Figure 6.15: Performance of the first 256 runs.

6.2.2 Aggregate Throughput

Table 6.20 and Table 6.21 show the aggregate throughput performance of the algorithms. The first we notice is, even though the GPF of the lgw and random-assignment has decreased significantly, the mean aggregate throughput of those two algorithms have not changed notably. This is because the logarithm of a small number in [0, 1] generates a negative number and decreases more rapidly than linear.

Secondly, even the means of all the algorithms are similar to the values in the 12-flow experiment in Table 6.7, the standard deviations of the algorithms decrease by a large extent. From Table 6.20 , we can see they all have lower max and higher min comparing with the values in Table 6.7. This is because, with more UEs, the uniform placement of UEs will have a higher probability to fill the whole simulated cell more evenly. It can reduce the chance of extreme cases which can make a significant throughput difference. For example, the case of only one UE at one AP versus no UE at the same AP.

Table 6.20: Aggregate Throughput (Mbps) [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	76.71	10.01	76.36	46.01	125.29
gg.t	76.46	10.23	76.01	44.52	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

Table 6.21: Aggregate Throughput (Mbps) Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.t	-0.53	-0.70	-0.53	-0.26	-0.60
atom.t	0.02	-0.08	0.02	0.21	-0.06
gg.t	0.01	-0.06	0.02	0.18	-0.07
rand.t	-0.43	-0.27	-0.44	-0.45	-0.36

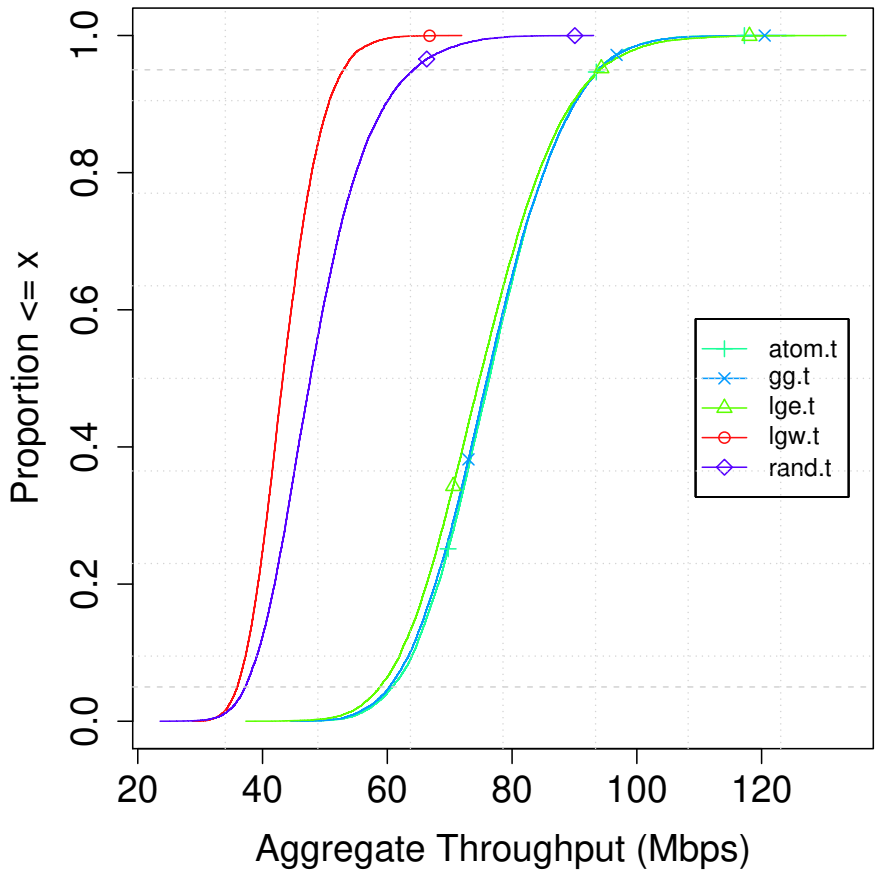

Figure 6.16: CDF of aggregate throughput over multiple runs.

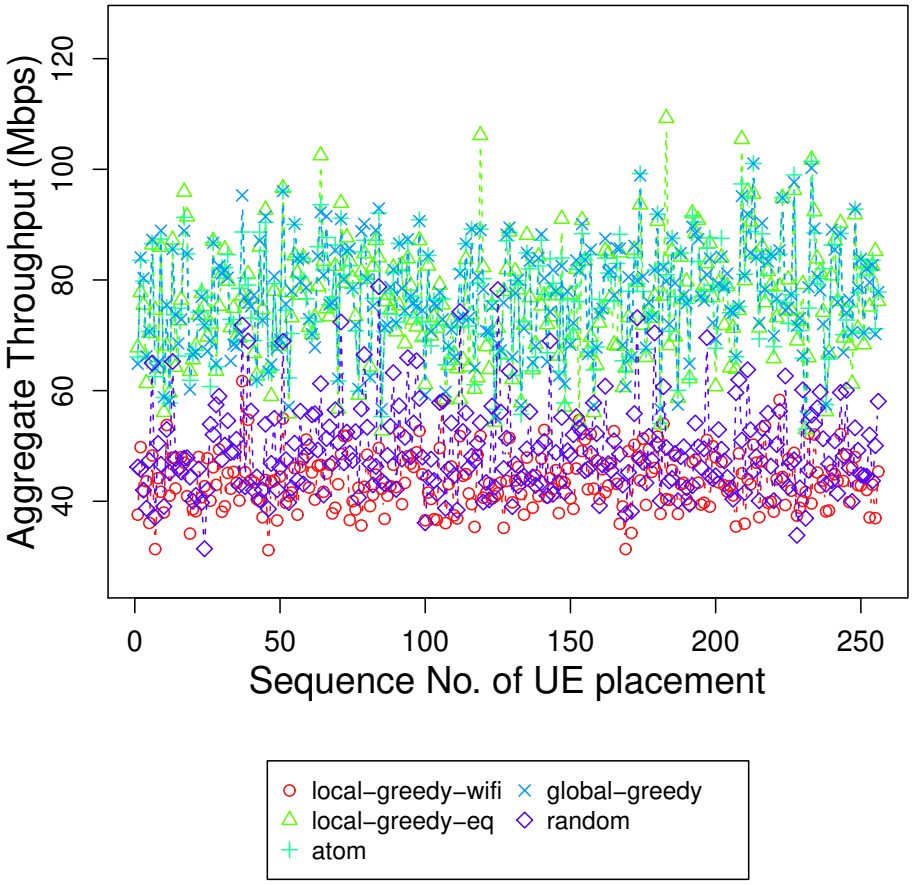

Figure 6.17: Performance of the first 256 runs.

6.2.3 TFI

Table 6.22 and Table 6.23 show the TFI performance results of algorithms. The first we notice is lgw owns a mean TFI of 0.9, which is much higher than the TFI of 0.78 achieved by lgw in the 12-flow experiment in Table 6.10. This agrees with our expectation and verifies the explanation for the reason for the high TFI of lgw we observed in Section 6.1.1.2. Secondly, we see the mean TFI values of the other algorithms remain almost the same as those in the 12-flow experiment as shown in Table 6.10. From Table 6.23, we also notice that the standard deviation of ATOM, global-greedy and lgw have all decreased. This has a similar reason as the decrease of GPF standard deviation.

Fig. 6.18 visualizes the CDF of TFI values. Fig. 6.19 further shows the time series graph of the TFI values. We can see the same rankings among algorithms as we analyzed above.

Table 6.22: Throughput Fairness Index [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.93	0.09	1.00
lge.TFI	0.64	0.15	0.66	0.07	1.00
atom.TFI	0.74	0.09	0.74	0.15	0.97
gg.TFI	0.75	0.08	0.76	0.15	0.98
rand.TFI	0.59	0.14	0.62	0.09	0.96

Table 6.23: Throughput Fairness Index Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.34	-0.53	0.34	0.27	0.00
atom.TFI	0.14	-0.52	0.12	0.76	-0.03
gg.TFI	0.16	-0.55	0.14	0.78	-0.02
rand.TFI	-0.08	-0.03	-0.07	0.23	-0.04

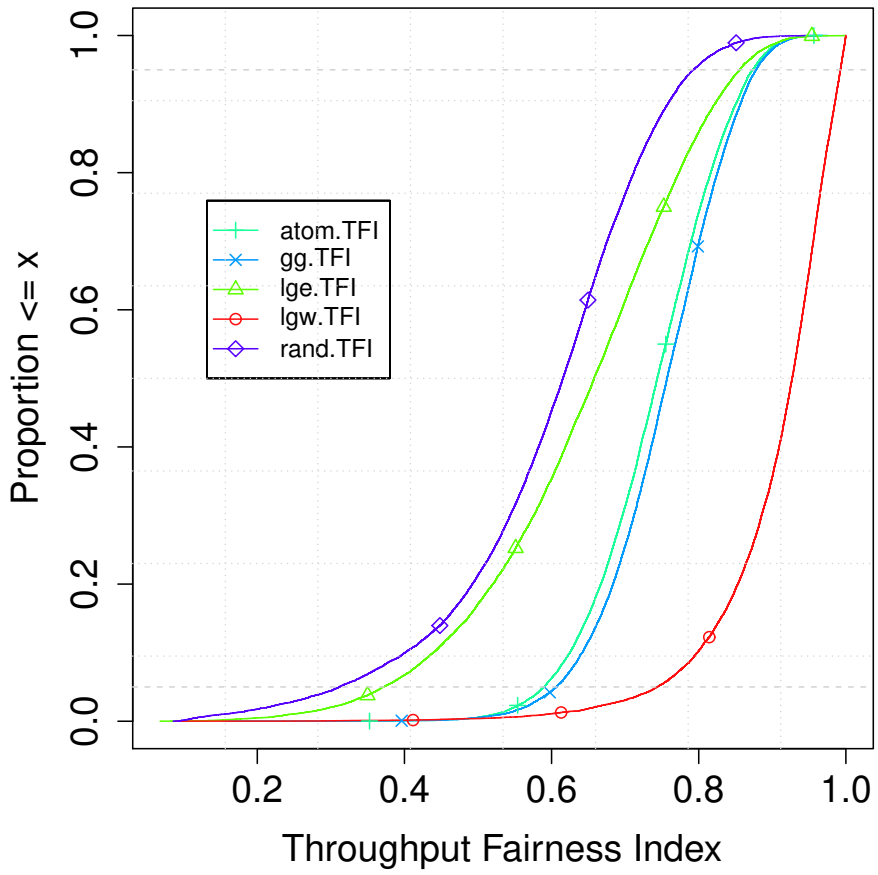

Figure 6.18: CDF of aggregate throughput over multiple runs.

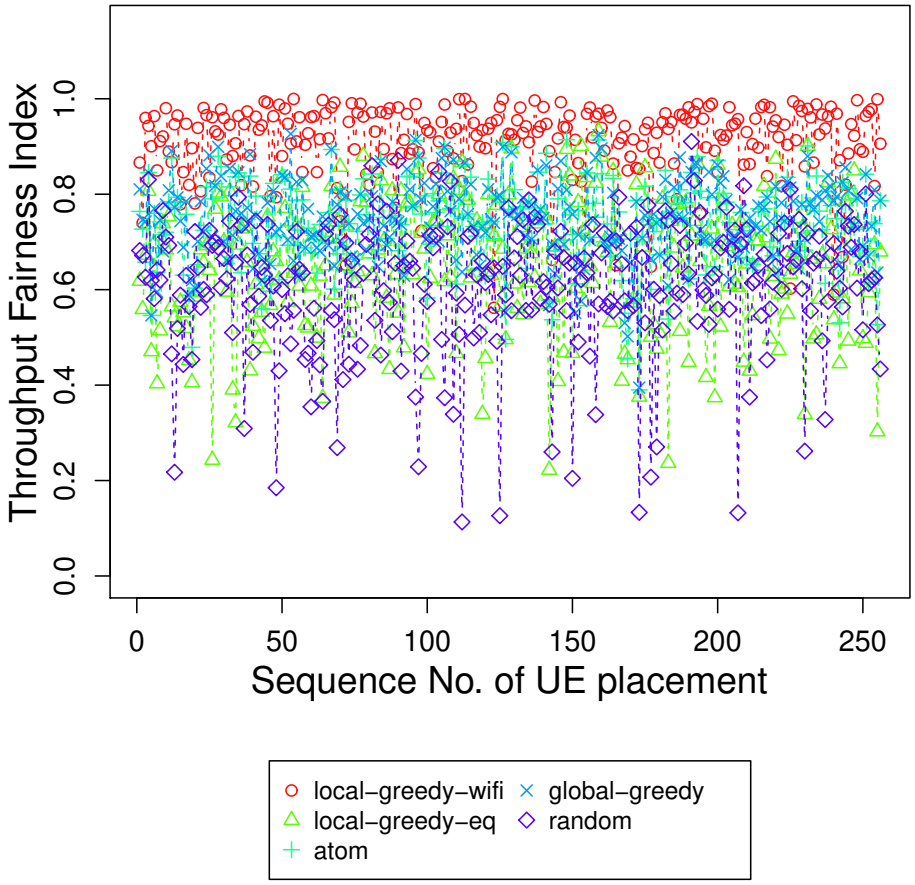

Figure 6.19: Performance of the first 256 runs.

6.2.4 Flow Level Results

Table 6.24 and Table 6.25 show the flow-level results with the 32-flow uniform placement baseline experiment. Interestingly, even though the difference between lge and the algorithms in the GIFMS is very small in the aggregate results, the difference between them in flow-level results is still noticeable in this 32-flow experiment, as compared to the 12-flow experiment in Section 6.1.2. We see the P_{equal} of ATOM and global-greedy is only about 25% now as opposed to around 50% in the 12-flow baseline experiment. $P_{greater}$ and $P_{greater}$ are now about (45%, 30%) compared to (33%, 17%) in the 12-flow baseline. We think this means even though the mean of lge is close to those achieved by the GIFMS algorithms, it owns larger per-flow deviations over all the runs as the number of UEs increases. These per-flow differences can be averaged or smoothed in two aggregation processes. First, it can be smoothed when the aggregate metrics are calculated over all the flows such as GPF value and aggregate throughput. Second, when calculating the means of these metrics over multiple runs, those differences can be further smoothed. Flow level results can convey information that is not available in aggregate results. This is the reason we report the flow-level results. If we look back to the TFI tables in Table 6.10 and Table 6.22, we can find some traces to this difference over runs. We see that even though the means of lge is close to the GIFMS algorithms in both 12-flow and 32-flow results, its standard deviation is noticeably higher than that of the GIFMS algorithms in the 32-flow result.

Table 6.24: Flow Level Comparison over lge

	$P_{greater}$	P_{equal}	P_{less}
lgw	8.01	7.60	84.39
lge	0.00	100.00	0.00
atom	44.36	26.98	28.66
gg	45.63	23.43	30.94
rand	29.96	0.39	69.65

Table 6.25: Flow Level Comparison over lgw

	$P_{greater}$	P_{equal}	P_{less}
lgw	0.00	100.00	0.00
lge	84.39	7.60	8.01
atom	85.00	3.04	11.96
gg	84.01	2.75	13.24
rand	51.66	0.13	48.21

6.3 Sensitivity to AP Power Levels

In this section, we evaluate the sensitivity of the results to AP power levels. We simulate the increase of AP power level by varying the κ value in Eq. (4.1). We use κ values in Table 6.26. The κ values in the table are calculated using Eq. (5.1) with the distance d in the equation equal to the LTE radius in Table 6.26. Since we use the same κ for LTE and WiFi, increasing κ essentially increases the coverage radiuses of all the APs. Since the simulated cell remains the same, it is also equivalent to shifting the UEs to use higher MCS indices.

Table 6.27 shows the change of GPF values as the κ value increases. We can see that all the statistics of GPF values for all the algorithms increase as expected. However, the ranking of algorithms remains the same for all three settings. Table 6.28 to Table 6.30 further show the relative differences of algorithms' GPF performance compared to that of local-greedy-equal-chance. We see that the relative differences of the mean, median and maximum GPF for the GIFMS algorithms compared to local-greedy-equal-chance do not change significantly as the κ value increases. However, the standard deviation of the relative differences does increase slightly. Looking back to the raw values in Table 6.27, we find even though the max values of the three algorithms are similar all the time, the min value for local-greedy-equal-chance does not increase as fast as the other two as the κ value increases. This means it does not handle a few bad cases as well as the other two. This is expected since the increased coverage makes more space for global optimization. The global optimization algorithms using GPF as the objective can show a slight advantage in these extreme cases. However, the average performance of local-greedy-equal-chance is still similar to the GIFMS algorithms with higher κ values.

Table 6.26: Kappa Values

LTE Radius	Kappa Value
1	0.146
1.1	0.187
1.2	0.235

Table 6.31 and Table 6.32 compare the performance of aggregate throughput and throughput fairness index for all the algorithms as the κ value increases for reference. From Table 6.31, we see all the statistics of the aggregate throughput increase similar to those of GPF. In Table 6.31, we can better observe the increase of the standard deviation for local-greedy-equal-chance. We can see even the max values of lge have noticeable increases compared with ATOM and global-greedy. The

Table 6.27: Comparison of the GPF Value with Different Kappa Values

kappa=0.146

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.93	4.25	21.93	6.53	40.30
gg.v	21.74	4.34	21.72	5.01	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

kappa=0.187

	mean	sd	median	min	max
lgw.v	11.33	3.58	11.31	-7.54	27.37
lge.v	25.07	4.40	25.16	6.91	42.36
atom.v	26.63	3.92	26.68	11.96	42.54
gg.v	26.45	3.99	26.51	10.34	42.36
rand.v	3.53	5.57	3.77	-24.83	22.89

kappa=0.235

	mean	sd	median	min	max
lgw.v	15.21	3.45	15.22	0.96	29.04
lge.v	28.95	4.20	29.06	8.38	45.49
atom.v	30.61	3.57	30.64	17.00	45.77
gg.v	30.48	3.64	30.51	16.46	45.81
rand.v	6.89	4.98	7.03	-18.67	26.96

Table 6.28: GPF Value Compared with lge [Kappa](kappa=1.0)

	mean	sd	median	min	max
lgw.v	-0.61	-0.17	-0.61	-2.85	-0.44
atom.v	0.05	-0.04	0.05	0.56	0.00
gg.v	0.04	-0.02	0.04	0.20	0.00
rand.v	-1.05	0.40	-1.04	-7.71	-0.47

Table 6.29: GPF Value Compared with lge [Kappa](kappa=1.1)

	mean	sd	median	min	max
lgw.v	-0.76	-0.20	-0.76	-2.00	-0.43
atom.v	0.06	-0.11	0.06	0.54	0.00
gg.v	0.05	-0.10	0.05	0.40	0.00
rand.v	-1.51	0.24	-1.48	-2.00	-0.60

Table 6.30: GPF Value Compared with lge [Kappa](kappa=1.2)

	mean	sd	median	min	max
lgw.v	-0.62	-0.20	-0.62	-1.59	-0.44
atom.v	0.06	-0.16	0.05	0.68	0.01
gg.v	0.05	-0.14	0.05	0.65	0.01
rand.v	-1.23	0.17	-1.22	-2.00	-0.51

GPF values do not show this trend because of the logarithm function makes the difference smaller at a magnitude around 140.

From Table 6.32, we can see the statistics of TFI for all the algorithms remain the same. This is because TFI is independent of the magnitude of throughput.

Table 6.31: Comparison of the Aggregate Throughput (Mbps) with Different Kappa Values

kappa=0.146

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	76.71	10.01	76.36	46.01	125.29
gg.t	76.46	10.23	76.01	44.52	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

kappa=0.187

	mean	sd	median	min	max
lgw.t	48.03	5.47	47.62	31.35	76.61
lge.t	85.64	11.21	85.12	43.96	140.42
atom.t	86.28	10.15	86.00	51.99	133.42
gg.t	86.22	10.36	85.93	51.36	133.50
rand.t	50.83	8.28	49.76	26.82	94.81

kappa=0.235

	mean	sd	median	min	max
lgw.t	54.09	5.82	53.83	35.76	80.33
lge.t	94.84	11.29	94.41	42.83	147.37
atom.t	95.52	10.21	95.30	61.84	140.96
gg.t	95.55	10.28	95.36	59.93	139.18
rand.t	52.47	8.34	51.38	28.76	97.75

Table 6.32: Comparison of the Throughput Fairness Index with Different Kappa Values

kappa=0.146

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.93	0.09	1.00
lge.TFI	0.64	0.15	0.66	0.07	1.00
atom.TFI	0.74	0.09	0.74	0.15	0.97
gg.TFI	0.75	0.08	0.76	0.15	0.98
rand.TFI	0.59	0.14	0.62	0.09	0.96

kappa=0.187

	mean	sd	median	min	max
lgw.TFI	0.91	0.09	0.94	0.10	1.00
lge.TFI	0.63	0.15	0.65	0.07	0.98
atom.TFI	0.77	0.08	0.77	0.36	0.97
gg.TFI	0.78	0.07	0.78	0.37	0.98
rand.TFI	0.64	0.14	0.66	0.09	0.98

kappa=0.235

	mean	sd	median	min	max
lgw.TFI	0.92	0.08	0.94	0.22	1.00
lge.TFI	0.66	0.15	0.68	0.09	1.00
atom.TFI	0.78	0.08	0.79	0.39	0.98
gg.TFI	0.80	0.07	0.80	0.38	0.99
rand.TFI	0.68	0.14	0.70	0.09	0.97

6.4 Impact of UE Clusters

We show the results of two types of UE clusters as we introduced in Section 5.1.3.1, i.e. circular clusters and rectangular clusters. Circular clusters are used to represent localized hot spots, and rectangular clusters represent UEs distributed along intersecting streets.

6.4.1 Circular Clusters

6.4.1.1 Circular Cluster Baseline Comparison

We first test the scenario with the following circular clusters. We use 3 circular clusters. The center of the first cluster is at (2/3, 0). The centers of the other two clusters are achieved by a rotation of the center of the first cluster about the origin by ± 45 degrees. We use the method described in Section 5.1.3.2 to place UEs into the three clusters. The probabilities of placing UEs into the three clusters are always equal. Each of the clusters has a radius of 0.25. In addition, we refer to the entire simulated cell the base cluster in this context. We denote the fraction of UEs in the base

cluster as Pb. In the following baseline study, we first test the scenario when Pb=0. We will test with various Pb values in the next section. Table 6.33 to Table 6.44 show the performance comparison between this circular cluster scenario with $Pb = 0$ and the uniform UE placement scenario. We use *cCluster* as an abbreviation for *Circular Cluster*. It is used in the captions of the tables. As the convention, we first show the comparison of raw data, and then that for the relative difference rates compared with the local-greedy-equal-chance.

As we can see from the GPF value tables, the GPF values of all the mapping algorithms have decreased significantly. However, lge and lgw have a much larger decrease. We previously showed lge to have a similar mean GPF value of around 21 as ATOM and global-greedy under the uniform distribution scenario. However, in the circular cluster scenario, the mean GPF value of lge has dropped to -8.58, while GIFMS such as ATOM and global-greedy manage to achieve a GPF value around 6. We can also see the GPF value of lgw has a decrease of a similar magnitude as lge.

From the aggregate throughput tables, we can see a similar ranking and trend as the GPF values. Interestingly, the TFI values for lgw are very low under this clustered scenario. Intuitively, if all the UEs connect to the same WiFi AP, they should get the same throughput which leads to high TFI. After further investigating the results, we find that the specific topology can generate very low TFI because of the cases similar to the one shown in Fig. 6.20. As we see, almost all the UEs connect to one WiFi AP while one UE connects to the other one. This will result in an extreme imbalance of throughputs between the two sets of UEs under the two WiFi APs and therefore low TFI.

We also notice the TFI of lge has a slightly smaller decrease than lgw but a much larger decrease than the ATOM and global-greedy.

Table 6.33: GPF Value [cCluster-Baseline]()

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	6.52	3.51	6.65	-10.82	18.35
gg.v	6.82	3.69	7.00	-11.75	18.41
rand.v	-10.83	4.94	-10.78	-34.15	7.65

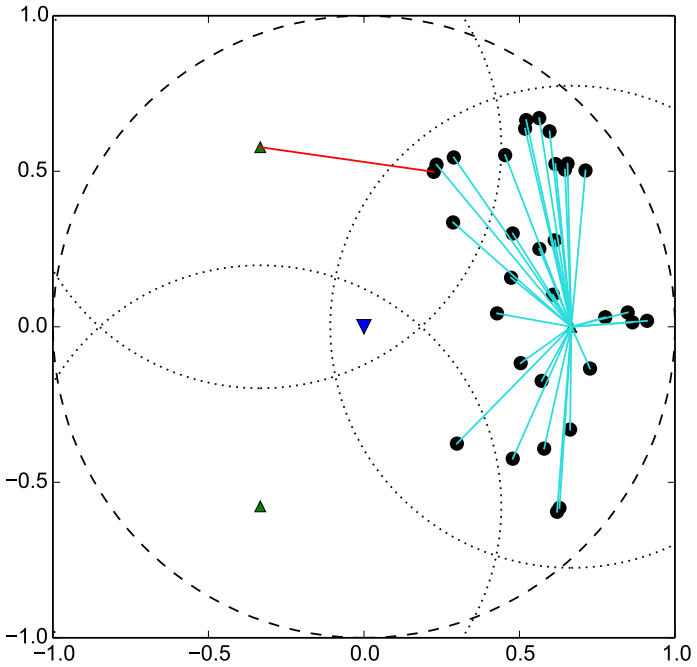

Figure 6.20: The case that lgw achieves the worst TFI when Pb value is low

Table 6.34: GPF Value [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.93	4.25	21.93	6.53	40.30
gg.v	21.74	4.34	21.72	5.01	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

Table 6.35: GPF Value Compared with lge [cCluster-Baseline]()

	mean	sd	median	min	max
lgw.v	-0.73	-0.13	-0.74	-0.24	-2.00
atom.v	2.00	-0.45	2.00	0.93	0.63
gg.v	2.00	-0.40	2.00	0.87	0.63
rand.v	-0.23	-0.12	-0.24	-0.14	-0.23

Table 6.36: GPF Value Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.v	-0.88	-0.19	-0.88	-2.00	-0.56
atom.v	0.05	-0.04	0.05	0.44	0.00
gg.v	0.04	-0.02	0.04	0.18	0.00
rand.v	-2.00	0.33	-2.00	-2.00	-0.61

Table 6.37: Aggregate Throughput (Mbps) [cCluster-Baseline]()

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	45.79	3.77	45.65	24.70	58.25
gg.t	47.16	4.08	47.07	29.21	60.65
rand.t	30.55	4.78	31.19	15.14	49.86

Table 6.38: Aggregate Throughput (Mbps) [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	76.71	10.01	76.36	46.01	125.29
gg.t	76.46	10.23	76.01	44.52	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

Table 6.39: Aggregate Throughput (Mbps) Compared with lge [cCluster-Baseline]()

	mean	sd	median	min	max
lgw.t	-0.34	-0.22	-0.29	-0.25	-0.32
atom.t	0.31	-0.37	0.31	0.64	0.04
gg.t	0.34	-0.29	0.34	0.79	0.08
rand.t	-0.09	-0.14	-0.06	0.18	-0.11

Table 6.40: Aggregate Throughput (Mbps) Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.t	-0.53	-0.70	-0.53	-0.26	-0.60
atom.t	0.02	-0.08	0.02	0.21	-0.06
gg.t	0.01	-0.06	0.02	0.18	-0.07
rand.t	-0.43	-0.27	-0.44	-0.45	-0.36

Table 6.41: Throughput Fairness Index [cCluster-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.44	0.21	0.36	0.12	1.00
lge.TFI	0.46	0.15	0.44	0.10	1.00
atom.TFI	0.75	0.12	0.78	0.35	0.99
gg.TFI	0.78	0.10	0.80	0.37	0.98
rand.TFI	0.52	0.16	0.47	0.22	0.98

Table 6.42: Throughput Fairness Index [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.93	0.09	1.00
lge.TFI	0.64	0.15	0.66	0.07	1.00
atom.TFI	0.74	0.09	0.74	0.15	0.97
gg.TFI	0.75	0.08	0.76	0.15	0.98
rand.TFI	0.59	0.14	0.62	0.09	0.96

Table 6.43: Throughput Fairness Index Compared with lge [cCluster-Baseline]()

	mean	sd	median	min	max
lgw.TFI	-0.05	0.37	-0.20	0.19	0.00
atom.TFI	0.48	-0.24	0.56	1.10	-0.01
gg.TFI	0.52	-0.38	0.59	1.14	-0.02
rand.TFI	0.12	0.07	0.08	0.73	-0.02

Table 6.44: Throughput Fairness Index Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.34	-0.53	0.34	0.27	0.00
atom.TFI	0.14	-0.52	0.12	0.76	-0.03
gg.TFI	0.16	-0.55	0.14	0.78	-0.02
rand.TFI	-0.08	-0.03	-0.07	0.23	-0.04

2. Flow level statistics

Table 6.45 and Table 6.46 show the flow-level statistics under the circular cluster scenario. Comparing with the tables under the uniform baseline in Table 6.24, we can see ATOM and global-greedy win the local greedy algorithms by a larger margin compared to the flow-level results of the uniform UE placement in Section 6.2.4. The $P_{greater}$ increases from 45% to 70%. This is consistent with the aggregate results. It also has the same reason as the aggregate results.

Table 6.45: Flow Level Comparison over lge

	$P_{greater}$	P_{equal}	P_{less}
lgw	6.91	4.82	88.27
lge	0.00	100.00	0.00
atom	72.77	1.93	25.30
gg	69.69	0.70	29.61
rand	52.41	0.59	47.00

Table 6.46: Flow Level Comparison over lgw

	$P_{greater}$	P_{equal}	P_{less}
lgw	0.00	100.00	0.00
lge	88.27	4.82	6.91
atom	81.78	2.13	16.09
gg	78.31	1.32	20.37
rand	65.04	0.89	34.07

6.4.1.2 Scenario When UEs Cluster around WiFi APs

We have also tested with the scenario when two WiFi APs on the left side of the simulated cell are moved to the same locations as the UE clusters. Table 6.47 to Table 6.49 show the results.

Table 6.47: GPF [cCluster-Move-APs]()

	mean	sd	median	min	max
lgw.v	37.32	1.18	37.53	28.62	40.17
lge.v	37.34	1.17	37.53	28.62	40.17
atom.v	33.76	2.64	33.93	22.44	41.45
gg.v	41.58	0.95	41.70	34.32	43.95
rand.v	8.78	4.93	8.81	-16.49	28.15

Table 6.48: Aggregate Throughput (Mbps) [cCluster-Move-APs]()

	mean	sd	median	min	max
lgw.t	106.15	2.00	106.24	97.59	112.96
lge.t	106.14	2.01	106.23	97.59	112.96
atom.t	110.86	5.42	110.98	85.32	126.16
gg.t	120.21	2.63	120.38	105.75	128.03
rand.t	54.54	9.88	52.65	26.45	105.23

Table 6.49: Throughput Fairness Index [cCluster-Move-APs]()

	mean	sd	median	min	max
lgw.TFI	0.93	0.07	0.96	0.40	1.00
lge.TFI	0.94	0.07	0.96	0.23	1.00
atom.TFI	0.64	0.08	0.64	0.34	0.90
gg.TFI	0.95	0.04	0.96	0.55	1.00
rand.TFI	0.68	0.17	0.72	0.09	0.99

From the results, we find that the system performance for all the flow mapping systems increases in this scenario, as opposed to the performance decrease in the baseline circular cluster scenario. This is because the specific topology tends to produce balanced loads to APs. ATOM and global-greedy still have the best GPF performance. The algorithms in LIFMS have GPF performance close to the algorithms in GIFMS. For the performance in terms of aggregate throughput, the

algorithms in LIFMS are also close to the best in GIFMS, and even slightly better than ATOM. This means even under UE clustered scenarios, if the clusters have spatial consistency with the AP locations producing balanced loads to APs, the algorithms in LIFMS will have similar performance as the algorithms in GIFMS in terms of both GPF and aggregate throughput.

Another noticeable difference to both the circular and uniform baseline results is that local-greedy-wifi-preferred has much better performance in this case, which is almost identical to the local-greedy-equal-chance. This is because the UEs are clustered around the three WiFi APs, which makes 1) choosing the closet WiFi AP a good option; 2) the local-greedy-equal-chance always choose WiFi APs which general identical solution as local-greedy-wifi-preferred. Fig. 6.21 shows one example of the association plan generated from local-greedy-equal-chance under this scenario. As we can see, UEs only select the closest WiFi APs since the LTE AP has a lower effective rate compared with that of the WiFi APs as UEs cluster very close to the WiFi APs.

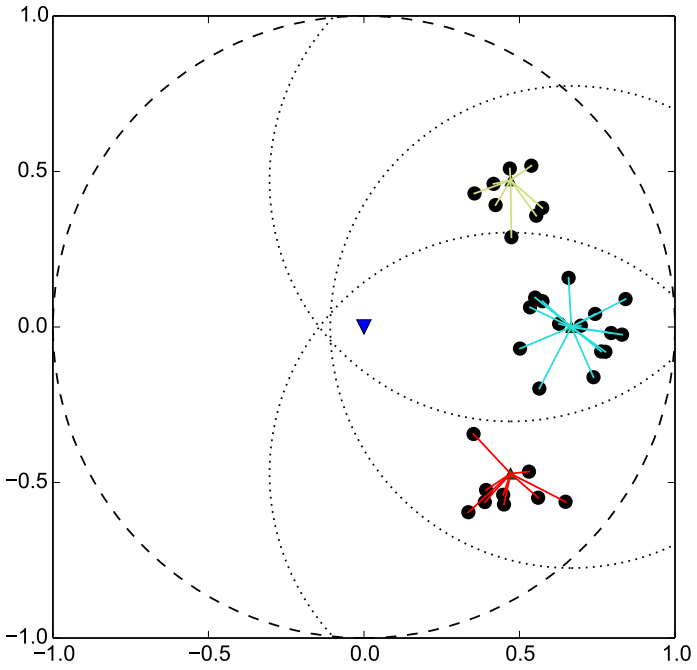

Figure 6.21: Why lgw and lge achieve almost identical results

6.4.1.3 Different Number of UEs in the Base Cluster

In the baseline evaluation, we put 0 UEs in the base cluster. We wonder how the performance will change as we allow more UEs to be placed in the base cluster.

In this evaluation, we test with 11 levels of Pb values from 0 to 1 with an increment step of 0.1. If Pb=0, it is the circular cluster scenario we tested in Section 6.4.1.1. If Pb=1, it is the uniform placement scenario we tested in Section 6.2.

Fig. 6.22 to Fig. 6.24 show the performance results of various mapping algorithms as the Pb increases. We can see lge is very close to ATOM and global-greedy when $Pb = 1$. However, as we put more and more UEs into clusters, the performance of lge degrades much faster than ATOM and global-greedy.

Section A of Appendix shows the detailed statistics, including the standard deviation, min, max and median of the results when using different Pb values.

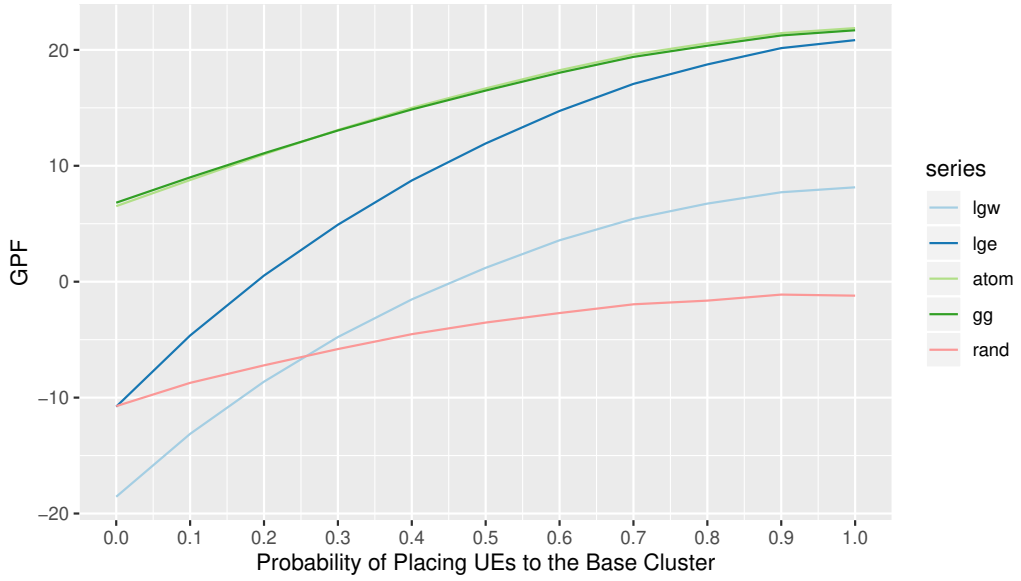

Figure 6.22: Mean GPF as a function of Pb values (circular cluster)

lgw has good TFI performance as it tends to generate solutions with WiFi only. All the WiFi APs in the simulated system use the same MCS table. If a solution further makes loads of WiFi APs similar, all the UEs will have similar throughput. Fig. 6.25 shows this phenomenon when lgw achieves maximum TFI.

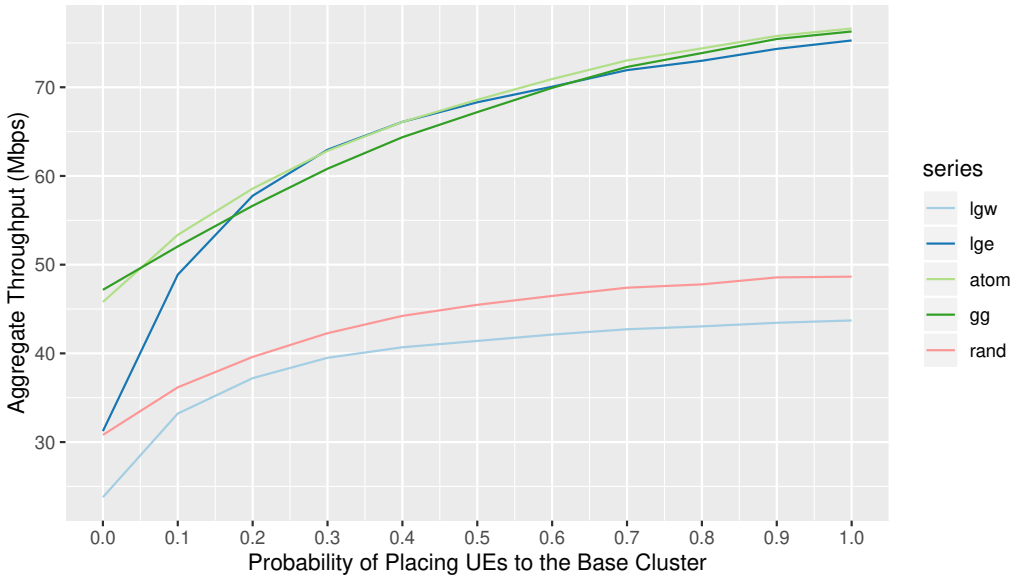

Figure 6.23: Mean aggregate throughput as a function of Pb values (circular cluster)

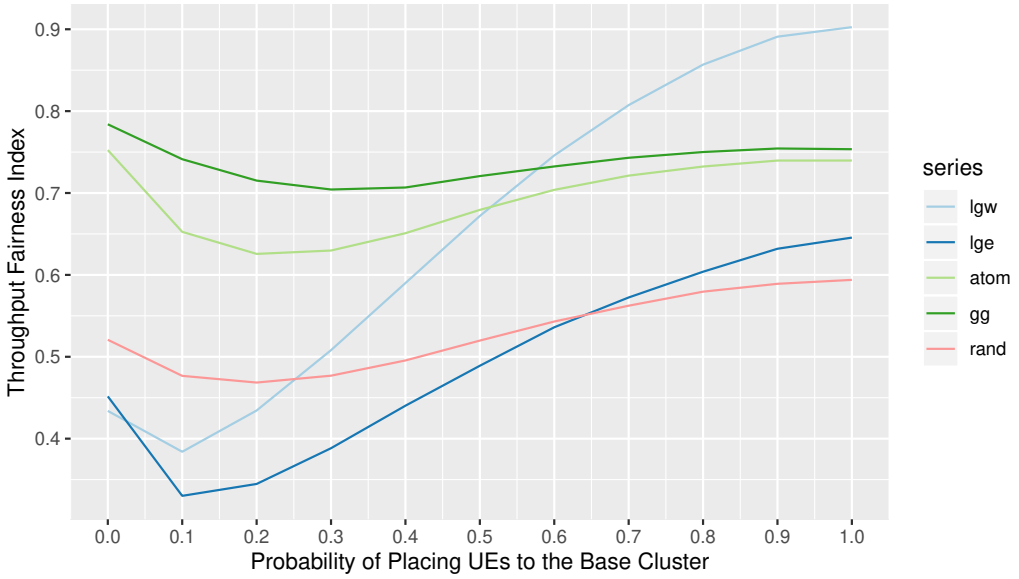

Figure 6.24: Mean TFI as a function of Pb values (circular cluster)

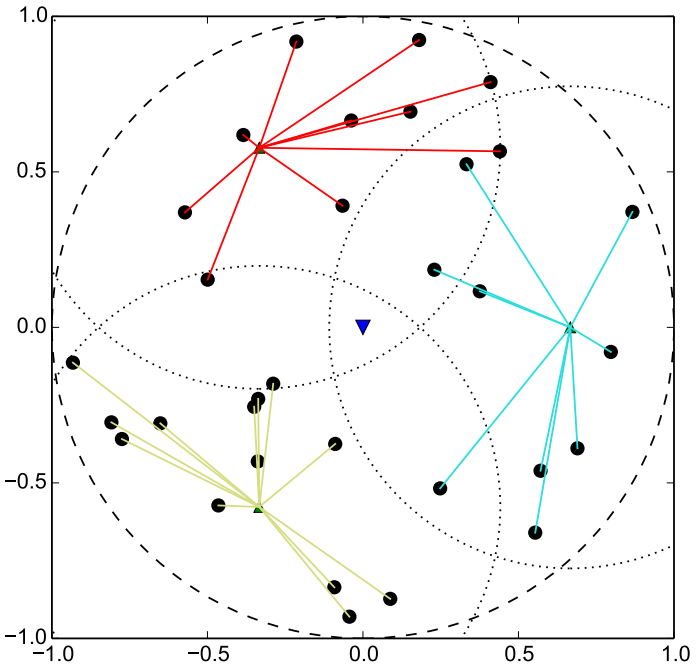

Figure 6.25: The case that lgw achieves the best TFI when Pb value is high

6.4.2 Rectangular Clusters

We use the method described in Section 5.1.3 to form a T-shaped region comprised of two rectangular clusters. The first rectangular cluster is centered at (0, 0) with a width and height of (2, 0.1). The second one is centered at (0, 0.5) with a width and height of (0.1, 1). The probability of UEs being placed into the first cluster is always twice as that of the second one.

6.4.2.1 Rectangular Cluster Baseline Comparison

We first conduct the baseline experiment when the probability to place UEs in the base cluster Pb equals to 0. We compare the baseline results in this clustered UE topology with those with a similar setting under uniform UE placement.

Table 6.50 to Table 6.61 show the detailed comparison results. We use *tCluster* as an abbreviation for *T-shaped Rectangular Cluster*. It is used in the captions of the tables. The first thing we see is that all the flow mapping systems have increased GPF values and aggregate throughput compared with the uniform baseline scenario. From this and the results in the circular cluster case, we know that some clustering types can decrease the overall performance while the other clustering types can increase it. Nevertheless, the GPF values of the GIFMS algorithms increase more than those of the LIFMS algorithms. Both ATOM and global greedy have a GPF value increase around 9 while all the LIFMS algorithms only have an increase of around 5. We think it is because of the natural advantages of more information and using GPF as the objectives in the GIFMS algorithms.

From the aggregate throughput tables, we can more clearly see the performance increase has three levels. ATOM and global-greedy have increased for around 28. lge has increased for around 17. random-assignment and lgw have only increased for around 5. This better reveals the three performance levels when handling this type of clustering scenario from the algorithms.

From the TFI tables, we notice that the TFI of the GIFMS algorithms remains almost the same. However, the lge has a noticeable TFI decrease from the T-shaped cluster scenario to the uniform scenario. From Fig. 6.29, we observe that this decrease only happen starting from around $Pb = 0.7$. After further investigations, we think this is caused by the greedy nature of lge. Fig. 6.26 shows an example when lge achieves the lowest TFI under the uniform placement. We see it is because the greedy selection can sometimes form this extreme case of only one UE is connected to the LTE. This will cause highly unbalanced throughput among UEs and therefore low TFI. However,

under T-shaped cluster, there will be a cluster of UEs around the LTE AP which will greatly reduce the chance of that kind of extreme case.

The TFI value has decreased when T-shaped cluster is used. It is because the uniform placement can help to form more evenly divided UEs to three WiFi APs, which can help lgw to boost TFI value. The T-shaped cluster has higher probability of placing UEs inside the areas that only LTE is available, which will give lgw low TFI.

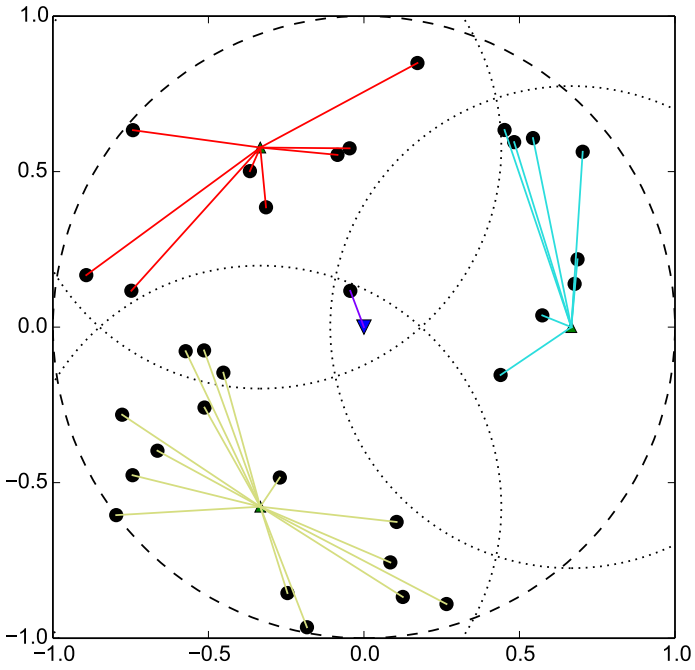

Figure 6.26: The case that lge achieves the worst TFI when Pb value is high

Table 6.50: GPF Value [tCluster-Baseline]()

	mean	sd	median	min	max
lgw.v	3.73	4.06	3.79	-12.84	17.91
lge.v	25.15	5.00	25.41	-2.87	41.46
atom.v	30.59	4.08	30.85	10.33	44.20
gg.v	30.52	4.24	30.80	9.65	44.20
rand.v	4.12	5.42	4.21	-19.65	27.83

Table 6.51: GPF Value [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.93	4.25	21.93	6.53	40.30
gg.v	21.74	4.34	21.72	5.01	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

Table 6.52: GPF Value Compared with lge [tCluster-Baseline]()

	mean	sd	median	min	max
lgw.v	-1.48	-0.21	-1.48	-1.27	-0.79
atom.v	0.19	-0.20	0.19	2.00	0.06
gg.v	0.19	-0.17	0.19	2.00	0.06
rand.v	-1.44	0.08	-1.43	-1.49	-0.39

Table 6.53: GPF Value Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.v	-0.88	-0.19	-0.88	-2.00	-0.56
atom.v	0.05	-0.04	0.05	0.44	0.00
gg.v	0.04	-0.02	0.04	0.18	0.00
rand.v	-2.00	0.33	-2.00	-2.00	-0.61

Table 6.54: Aggregate Throughput (Mbps) [tCluster-Baseline]()

	mean	sd	median	min	max
lgw.t	40.92	5.27	40.33	25.83	61.56
lge.t	92.66	8.20	92.75	57.21	126.27
atom.t	103.39	8.31	103.61	69.15	132.88
gg.t	104.91	8.69	105.14	69.33	133.60
rand.t	51.74	8.91	50.77	27.21	96.05

Table 6.55: Aggregate Throughput (Mbps) [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	76.71	10.01	76.36	46.01	125.29
gg.t	76.46	10.23	76.01	44.52	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

Table 6.56: Aggregate Throughput (Mbps) Compared with lge [tCluster-Baseline]()

	mean	sd	median	min	max
lgw.t	-0.77	-0.44	-0.79	-0.76	-0.69
atom.t	0.11	0.01	0.11	0.19	0.05
gg.t	0.12	0.06	0.13	0.19	0.06
rand.t	-0.57	0.08	-0.58	-0.71	-0.27

Table 6.57: Aggregate Throughput (Mbps) Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.t	-0.53	-0.70	-0.53	-0.26	-0.60
atom.t	0.02	-0.08	0.02	0.21	-0.06
gg.t	0.01	-0.06	0.02	0.18	-0.07
rand.t	-0.43	-0.27	-0.44	-0.45	-0.36

Table 6.58: Throughput Fairness Index [tCluster-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.80	0.13	0.82	0.21	1.00
lge.TFI	0.71	0.11	0.73	0.12	0.94
atom.TFI	0.75	0.07	0.76	0.25	0.96
gg.TFI	0.74	0.08	0.75	0.26	0.96
rand.TFI	0.61	0.11	0.62	0.09	0.95

Table 6.59: Throughput Fairness Index [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.93	0.09	1.00
lge.TFI	0.64	0.15	0.66	0.07	1.00
atom.TFI	0.74	0.09	0.74	0.15	0.97
gg.TFI	0.75	0.08	0.76	0.15	0.98
rand.TFI	0.59	0.14	0.62	0.09	0.96

Table 6.60: Throughput Fairness Index Compared with lge [tCluster-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.12	0.14	0.12	0.57	0.06
atom.TFI	0.06	-0.47	0.04	0.72	0.02
gg.TFI	0.04	-0.38	0.02	0.74	0.03
rand.TFI	-0.15	-0.01	-0.17	-0.32	0.01

Table 6.61: Throughput Fairness Index Compared with lge [Uniform-Baseline]()

	mean	sd	median	min	max
lgw.TFI	0.34	-0.53	0.34	0.27	0.00
atom.TFI	0.14	-0.52	0.12	0.76	-0.03
gg.TFI	0.16	-0.55	0.14	0.78	-0.02
rand.TFI	-0.08	-0.03	-0.07	0.23	-0.04

6.4.2.2 Different Number of UEs in the Base Cluster

Fig. 6.27 to Fig. 6.29 further show the results of the flow mapping algorithms as we vary the *Pb* values the same way as in Section 6.4.1.1. We can see that even if all the algorithms in the leading group (ATOM, gg, lge) have similar GPF values and aggregate throughput, increased UE clustering can make the performance of lge much worse than that of the ATOM and global-greedy. Comparing with the uniform baseline scenario, the performance of lge is still much worse than the other two algorithms in the leading group. This demonstrates that the GIFMS can be useful comparing with local-greedy-equal-chance when the sites have a clear UE-clustering characteristic.

From Fig. 6.27, We notice lgw has a different trend compared to the algorithms in the leading group as *Pb* value increases. It is because of the cases as shown in Fig. 6.30. Due to the "WiFi preferred" nature and the T-shaped cluster, lgw can produce association plans that congest only one WiFi AP while leaving the other APs nearly idle. This can produce low GPF and aggregate throughput. The other algorithms in the leading group avoid this by putting more flows to the LTE.

Section B of the Appendix shows the detailed statistics including the standard deviation, min, max and median of the results when using different *Pb* values. We can see the standard deviation of all the metrics remains stable as the *Pb* increases.

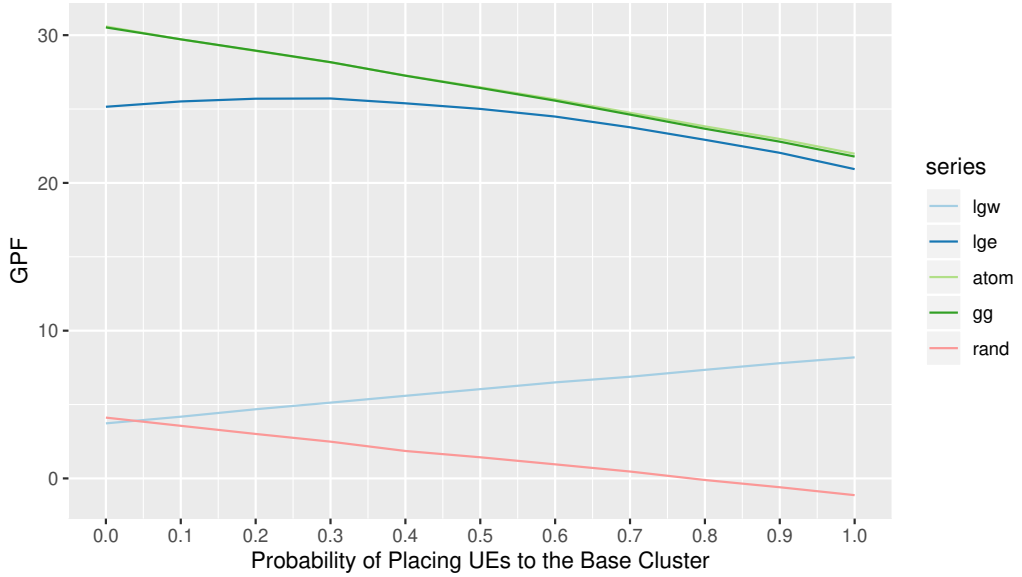

Figure 6.27: Mean GPF as a function of Pb values (rectangular cluster)

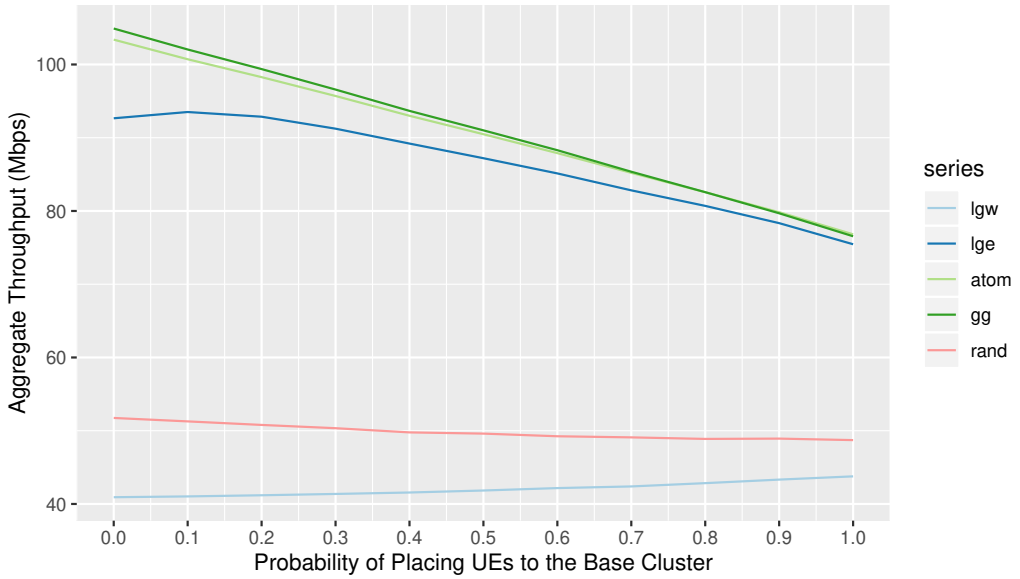

Figure 6.28: Mean aggregate throughput as a function of Pb values (rectangular cluster)

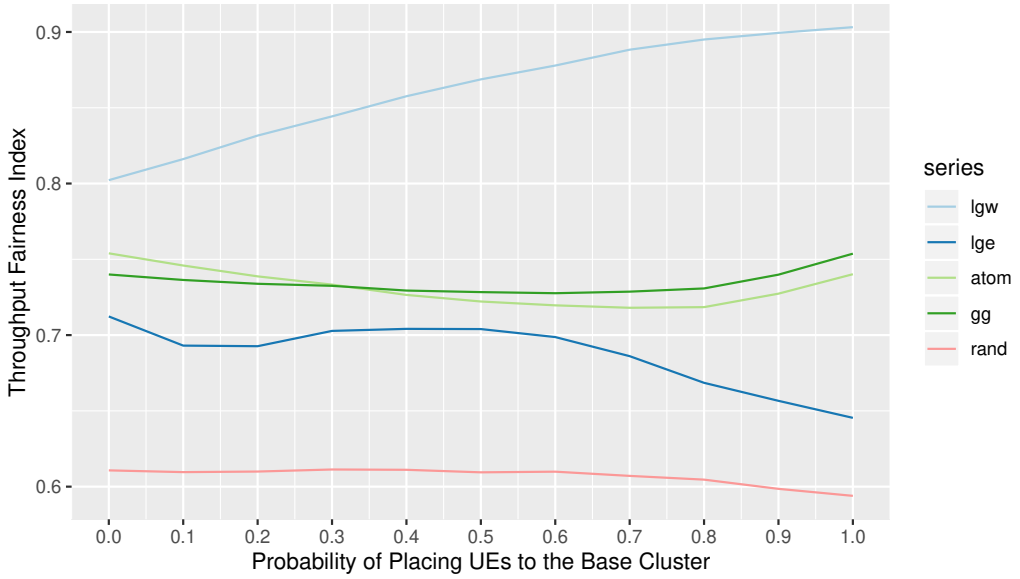

Figure 6.29: Mean TFI as a function of Pb values (rectangular cluster)

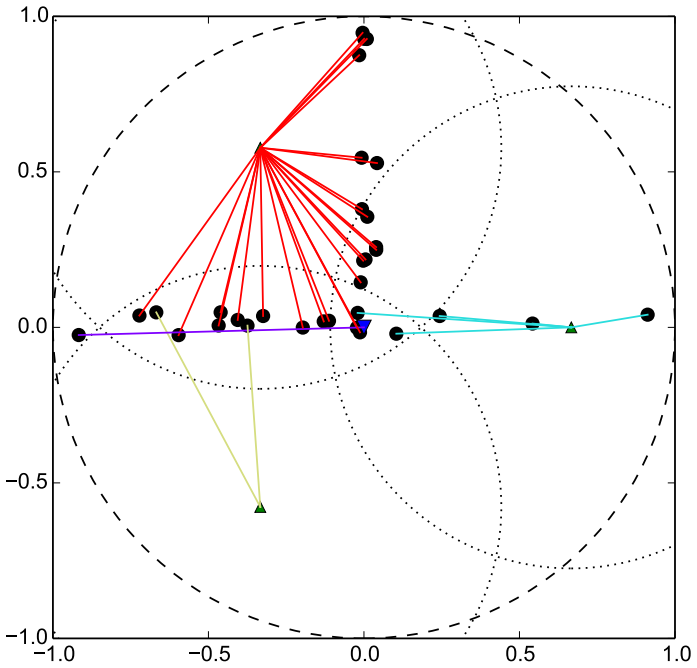

Figure 6.30: Why lgw has worse performance when T-shaped clusters are used

6.5 Impact of Changing WiFi APs to PF Scheduled

In this section, we evaluate the impact of changing WiFi APs to use PF scheduling for the performance of flow mapping systems. It is evaluated under both the uniform UE topology and circular cluster UE topology.

6.5.1 Uniform UE Topology

We first test with uniform UE topology similar to the uniform baseline in Section 6.2. Table 6.62 to Table 6.64 show the results of GPF, aggregate throughput and TFI respectively. As expected, the performance of both GPF and aggregate throughput for all the mapping algorithms has increased while that of TFI has decreased. This is because proportional fairness trades for higher spectrum efficiency with a larger distance to an equal throughput allocation. We also note that the relative difference and ranking of all the mapping algorithms do not change.

Table 6.62: GPF [WiFi-PF]()

	mean	sd	median	min	max
lgw.v	13.04	3.51	13.09	-1.97	25.45
lge.v	24.91	4.10	24.97	9.13	41.39
atom.v	25.42	3.91	25.44	9.96	41.91
gg.v	25.38	3.93	25.43	10.10	41.88
rand.v	1.55	6.66	1.90	-33.85	23.06

Table 6.63: Aggregate Throughput (Mbps) [WiFi-PF]()

	mean	sd	median	min	max
lgw.t	58.25	5.85	58.14	36.80	82.39
lge.t	91.01	10.75	90.54	52.88	146.71
atom.t	89.04	9.94	88.65	54.14	131.96
gg.t	89.71	10.30	89.27	54.32	130.41
rand.t	62.50	9.24	62.13	27.12	109.93

Table 6.64: Throughput Fairness Index [WiFi-PF]()

	mean	sd	median	min	max
lgw.TFI	0.73	0.07	0.75	0.11	0.89
lge.TFI	0.65	0.11	0.67	0.09	0.89
atom.TFI	0.70	0.07	0.71	0.37	0.89
gg.TFI	0.69	0.07	0.70	0.18	0.89
rand.TFI	0.47	0.10	0.47	0.09	0.80

6.5.2 Circular Cluster UE Topology

We then evaluate with a similar setting as the circular cluster baseline in Section 6.4.1.1. Table 6.65 to Table 6.67 show the results of GPF, aggregate throughput and TFI respectively. We observe similar system performance boost as in the uniform topology case. The ranking of all the flow mapping algorithms does not change either in this case.

Table 6.65: GPF [WiFi-PF]()

	mean	sd	median	min	max
lgw.v	-12.41	4.50	-12.31	-32.18	3.83
lge.v	-3.39	5.03	-3.25	-23.98	12.44
atom.v	8.10	3.26	8.26	-8.23	18.03
gg.v	8.72	3.25	8.90	-7.86	18.71
rand.v	-7.97	5.09	-7.84	-32.64	8.64

Table 6.66: Aggregate Throughput (Mbps) [WiFi-PF]()

	mean	sd	median	min	max
lgw.t	30.59	4.48	31.10	14.77	46.14
lge.t	40.15	5.57	40.18	18.79	59.74
atom.t	48.83	3.18	48.98	29.21	59.23
gg.t	49.03	3.39	49.22	32.52	59.76
rand.t	36.75	4.99	37.16	19.13	51.45

Table 6.67: Throughput Fairness Index [WiFi-PF]()

	mean	sd	median	min	max
lgw.TFI	0.51	0.14	0.46	0.19	0.88
lge.TFI	0.54	0.13	0.53	0.14	0.90
atom.TFI	0.75	0.10	0.77	0.40	0.95
gg.TFI	0.78	0.09	0.81	0.40	0.96
rand.TFI	0.52	0.10	0.50	0.25	0.86

6.6 Impact of Non-Participants

If the service provider of a GIFMS can only incrementally enroll users to participate in the system, there will be non-participants in the system. In this dissertation, we assume the following type of non-participant. The non-participants do not report scheduling information as required for the GIFMS as we detailed in Chapter 8. They will neither receive nor comply with the commands from the mapping system. They use certain policy-based LIFMS. For the results reported here, the non-participants were assumed to use local-greedy-equal-chance. Because this type of non-

participants do not report the necessary information such as its effective rates from all the APs to the GIFMS, it is impossible to optimize the GPF of the whole system including the non-participants. In this case, the optimization objective of the mapping system can only be optimizing the performance metrics (in our case the Generalized Proportional Fairness) for all the participants. However, when evaluating system performance, we need to include both participants and non-participants. Therefore in the following evaluations, we optimize the GPF of participants while measuring the GPF of the whole system including both participants and non-participants.

Concerning the system performance in terms of GPF, there are two major impacts from the non-participants.

1. The final association plan in a scenario with non-participants can be seen as a concatenation of the solution of the participants using a GIFMS solution and that of the non-participants using an LIFMS solution. As we have seen in the evaluations in Chapter 6, the performance of LIFMS can have various performance distances to the GIFMS under different system parameters.

2. Additionally, having a part of users use LIFMS while another part of users use GIFMS is similar to have two brains in a system giving commands to two parts of users. The commands can conflict if no appropriate communications between the two brains.

   For example, the non-participants can produce "hidden traffic" that is not known by the flow mapping system. The hidden traffic competes for resources on the APs in the system with participants. This renders the calculation process in Section 3.5 inaccurate when estimating the apportioned throughput of participants.

The combined effect of the two factors above is unknown. It is the reason that we evaluate the impact of the non-participants to the flow mapping systems using the following experiments.

6.6.1 Impact of Non-Participants under Clustered UE Topology

We experiment with the same system parameters as the baseline circular cluster experiment in Section 6.4.1.1, while varying the number of non-participants as {0, 4, 8, 12, 16, 20, 24, 28, 32}. We call the ratio between the number participants and the total number of UEs in the system as *deployment ratio* (dRatio). The corresponding deployment ratios are {1, 0.875, 0.75, 0.625, 0.5, 0.375, 0.25, 0.125, 0}.

Fig. 6.31 to Fig. 6.33 show the performance metrics of GPF, aggregate throughput and TFI respectively. The x-axes of the figures are the deployment ratio. We can see, as the deployment ratio increases, the performance of local greedy algorithms do not change as expected. However, the performance of both GIFMS algorithms have a clear increasing trend as more UEs participate. We can see the increase is close to linear from dRaio=0 to dRatio=0.75. This shows the performance of GIFMS is superior to that of LIFMS under certain clustered scenarios, while the improvement ratio is nearly linear to the participating ratio.

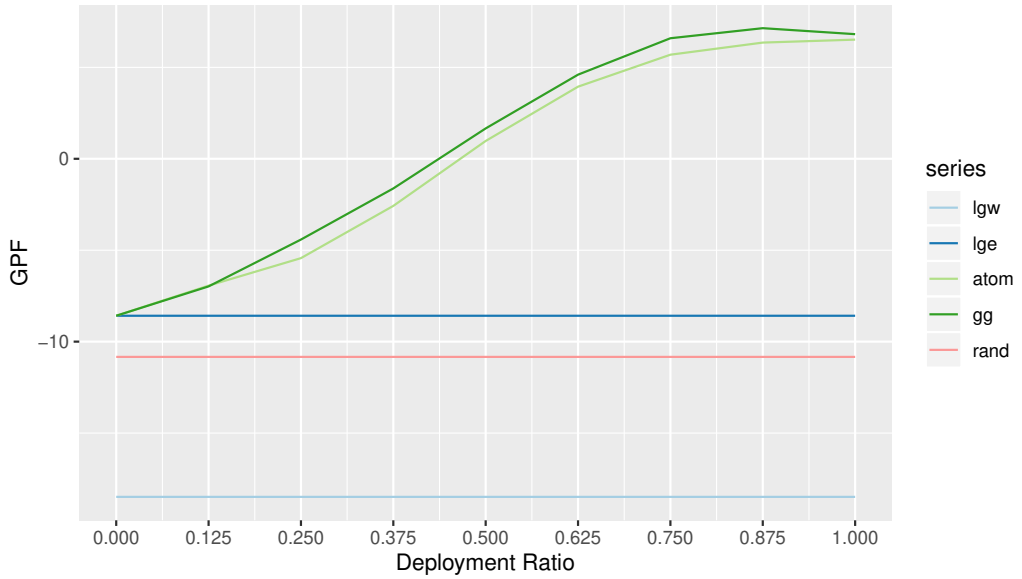

Figure 6.31: GPF of the flow mapping algorithms with various deployment ratios (circular cluster)

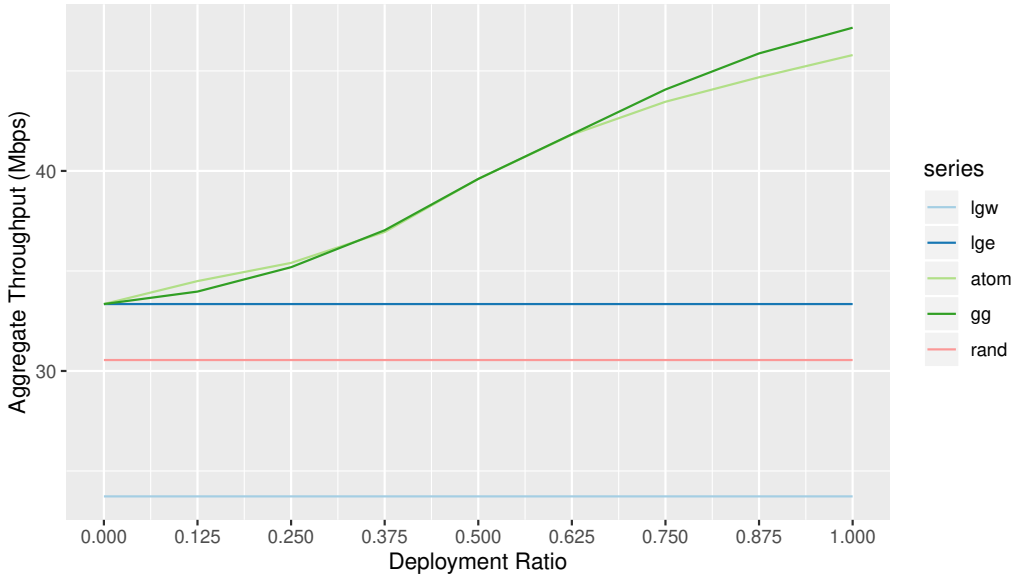

Figure 6.32: Aggregate throughput of the flow mapping algorithms with various deployment ratios (circular cluster)

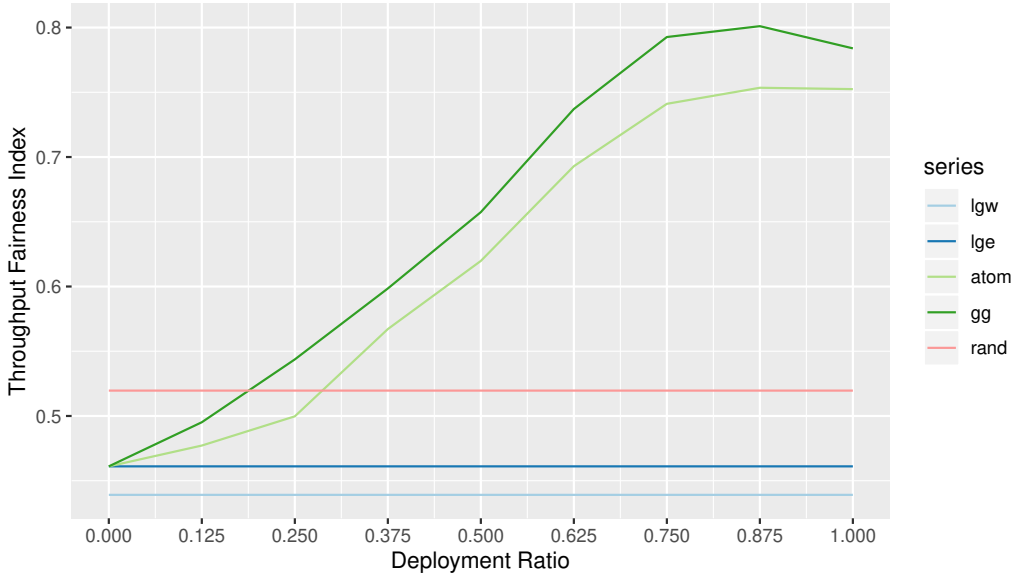

Figure 6.33: TFI of the flow mapping algorithms with various deployment ratios (circular cluster)

6.6.2 Impact of Non-Participants under Uniform UE Topology

Fig. 6.34 to Fig. 6.36 show the performance metrics similar to the ones in Section 6.6.1, but with a uniform UE placement. As we can see, the performance of GIFMS barely increases as the deployment ratio increases. It is because the performance difference between the lge and

the GIFMS algorithms is small under the uniform UE placement. We can see both the GPF and aggregate throughput have even decreased slightly when the dRatio is small (e.g. dRatio=0.125). The performance of adding few participants can be worse than all of them using LIFMS. We believe this shows the possible performance degradation from the concatenation of two solutions or conflict of the commands from two "brains".

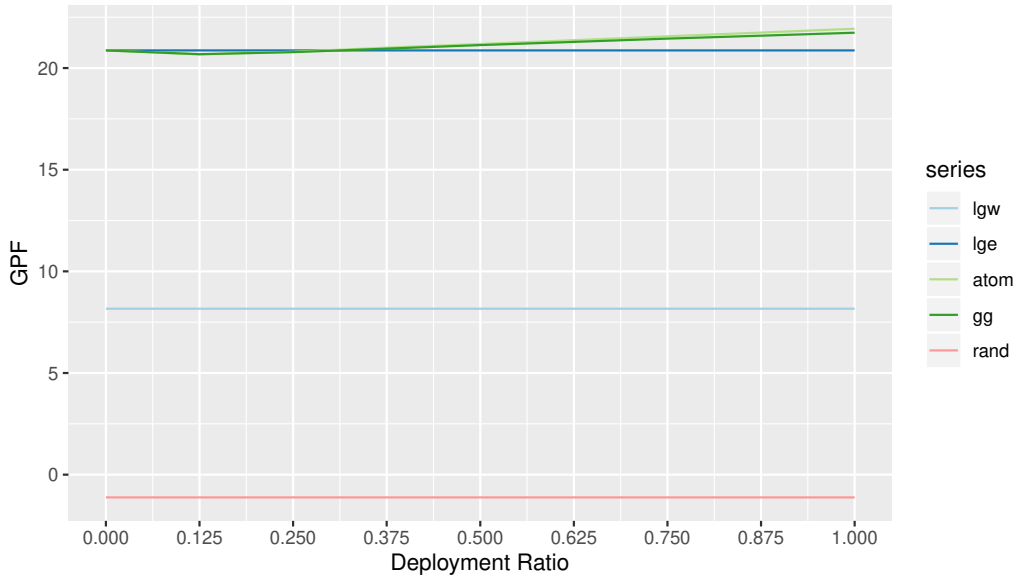

Figure 6.34: GPF of the flow mapping algorithms with various deployment ratios (uniform)

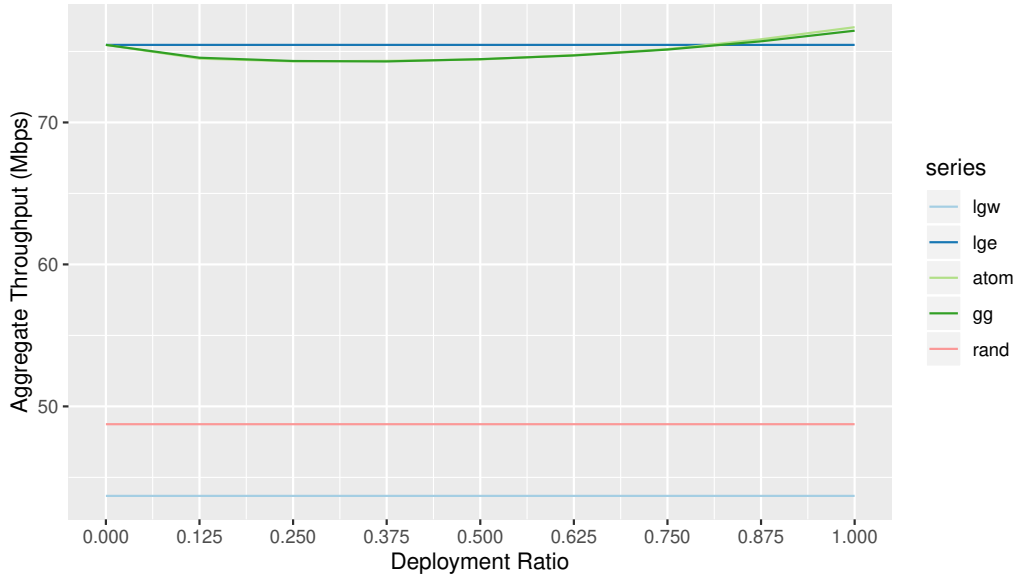

Figure 6.35: Aggregate throughput of the flow mapping algorithms with various deployment ratios (uniform)

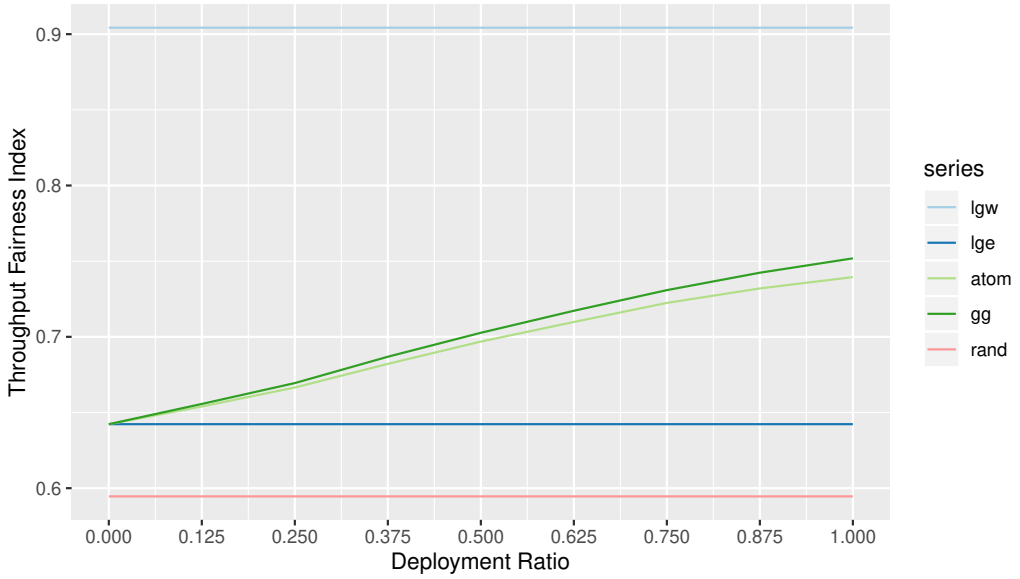

Figure 6.36: TFI of the flow mapping algorithms with various deployment ratios (uniform)

6.6.3 Possible Improvement to the Throughput Estimation Accuracy of Participants

The following improvements to the throughput estimation of the participating UEs can be conducted for GIFMS by inferring the hidden loads generated by the non-participants. We assume the mapping system knows the following information of each participating UE from its reporting as detailed in Section 8.3.

1. Real throughput from AP_j to UE_i, denoted as T'_{ij}
2. The effective peak rate from AP_j to UE_i, denoted as r_{ij}

Intuitively, if there was hidden traffic under one AP, the reported T'_{ij} will be smaller than the apportioned throughput T_{ij} calculated using Eq. (3.6) and Eq. (3.8) based on r_{ij}. The idea is to use this difference to infer certain measure of the non-participants, and then use that information to correct the throughput estimation. We call this technique *throughput correction*.

We derive the general formulas for throughput corrections for UE under both the proportional and max-min fair scheduled APs. We also show the inference procedure by way of example. In the example, we assume there are 5 flows with r_{ij} (6, 3, 9, *18*, *24*). Only the last two flows are hidden flows.

6.6.3.1 Proportional Fair APs

According to how we model the bandwidth sharing effect of a proportional fair AP in Eq. (3.6), if there was any hidden traffic on it, all it takes to correct the throughput of non-participants is to know the total number of non-participants connected to it (N_{non}) and adds it to the number of flows in Eq. (3.6) as follows,

$$T_{ij}^{*} = \frac{r_{ij}}{N + N_{non}} \tag{6.2}$$

where T_{ij}^{*} is the throughput of AP_j to UE_i after correction. N is the total number of flows can be seen by the mapping system.

If the AP has a PF scheduler, the real throughput T'_{ij} that the three participants measure will be (6/5, 3/5, 9/5) respectively according to Eq. (3.6). They report to the GIFMS with the following r_{ij} and T'_{ij} pairs, $< (6, 6/5),\ (3, 3/5),\ (9, 9/5) >$. The mapping system only sees three flows. It calculates the T_{ij} using Eq. (3.6). The resulting T_{ij} for the three participating flows will be (6/3, 3/3, 9/3), or (2, 1, 3). We can see the reported T'_{ij} = (6/5, 3/5, 9/5) are smaller than the calculated T_{ij} = (2, 1, 3). Note using the pair of throughput information from any participating flow, we can infer the number of non-participants on this AP. Using the first flow as an example, we know $\frac{6}{3+N_{non}} = 6/5$. Therefore, N_{non} =2, which means there are 2 hidden flows on this AP.

We can then use the Eq. (6.2) to correct the throughput estimations of all the participants on that AP.

Note to speed up the mapping algorithms, we use the incremental GPF function evaluation technique described in Eqs. (3.9) and (3.10) to calculate the objective function value change instead of doing an O(MN^2) evaluation of the throughput of each UE every time. Those calculations can also be corrected with the information inferred about the non-participants.

For example, if we try to add a flow to a PF AP, Eq. (3.9) can be corrected as follows,

$$\Delta U^{*} = K * log(\frac{K + N_{non}}{K + N_{non} + 1}) + log(\frac{r_{qj}}{K + N_{non} + 1}) \tag{6.3}$$

Note, the first K in Eq. (6.3) does not need to be corrected, as the objective function only considers the participants.

6.6.3.2 Max-Min Fair APs

For a max-min fair AP, according to the bandwidth sharing equation in Eq. (3.8), the only information needed to correct the throughput estimation for participants is the round time of non-participants on it (R_{non}). For k non-participants with effective rates $\hat{r_{kj}}$, $R_{non} = \sum_k \frac{1}{r_{kj}}$. Let the round time of participants be R while that for non-participants be R_{non}. The throughput estimation can be corrected as

$$T_{ij}^* = \frac{1}{R + R_{non} + \frac{1}{r_{qj}}} \tag{6.4}$$

We use the same 5-flow example in Section 6.6.3.1 to illustrate this process. The real throughput the three participants measured will be 1.412 as the round time is around 0.708. However, from the reported effective rates of participants, the mapping system gets a round time of 0.55. Therefore, we know the round time contributed by the non-participants (R_{non}) is around 0.708 - 0.55 $\approx$ 0.158. Then we can use Eq. (6.4) to correct the throughput of participants.

Similar to the PF AP, we also use the incremental evaluation technique in Eqs. (3.11) and (3.12) to speed up the mapping algorithms. The calculation procedures can also be corrected using the inferred R_{non}. For example, if we are going to add one flow to a max-min fair AP, Eq. (3.11) can be corrected as the following,

$$\Delta U^* = (K+1) \times log(\frac{1}{R + R_{non} + \frac{1}{r_{qj}}}) - K \times log(\frac{1}{R + R_{non}}) \tag{6.5}$$

6.6.3.3 Summary

The scheme described in this section can be considered as adding a one-direction communication channel from the LIFMS used by non-participants to GIFMS used by participants, or creating a one-direction information flow between the two "brains" in the system. Because the GIFMS now considers the effects from the LIFMS while LIFMS does not consider that from GIFMS. However, this technique can only improve the accuracy of the throughput estimation of the participants. As for the system performance, it is still unpredictable because the combined effect of this improvement and the other factors such as the possible performance degradation from the LGFMS solution is still unknown.

6.6.4 Results with Throughput Correction for Participants

6.6.4.1 Clustered UE Topology

Section C.1 shows the detailed comparison between the results with and without throughput correction for participants under a circular cluster UE topology the same as that in Section 6.4.1. As we can see, the throughput correction can help to improve performance in some cases. However, it does not always the case. For example, for ATOM, both the GPF and aggregate throughput can get slightly worse after the throughput correction.

6.6.4.2 Uniform UE Topology

Section C.2 shows the detailed comparison between the results with and without throughput correction for participants under a uniform UE topology the same as that in Section 6.2. We can see a similar trend as in the cluster case. However, since the difference of lge to the GIFMS algorithms is much smaller in this case, it is more difficult to tell the combined effects in this case.

6.7 Discussion on How to Model Inelastic Traffic

Even though this dissertation only evaluates the scenarios when all the UEs have elastic traffic, it is still possible to model the inelastic traffic when the demands of traffic are known. If there was no hidden traffic from non-participants, the mapping system can estimate the demand information of UEs by collecting UEs' throughput measurements. The scenarios with both non-participants and elastic traffic is more complicated, which we will brief at the end of this section.

If a flow is inelastic, let us denote its demand as D and its apportioned rate if connected to AP_j with elastic traffic as T_{ij}, which can be calculated using Eq. (3.6) or Eq. (3.8). There are two cases to consider,

1. $D \geq T_{ij}$

In this case, the flow can be modeled the same as an elastic flow. There is no impact of this kind of elastic flows to the throughput of other flows on AP_j.

2. $D < T_{ij}$

We call the inelastic flow in this case a *low-demand inelastic flow*. Since it does not use all of its proportional or max-main fair share of the resources, the other flows will get more resources

and therefore higher throughput. Note the throughput of each low-demand inelastic flow is exactly its demand. We will detail how to conduct the throughput estimation corrections for the other flows which is not a low-demand inelastic flow on the same AP for proportional and max-min fair scheduled APs respectively.

6.7.1 Throughput Estimation Correction for Flows under PF Scheduled APs

Under a PF scheduled AP, if the number of active UEs connected is K, the time share used by UE_i with inelastic traffic on AP_j (τ^*_{ij}) can be expressed as,

$$\tau^*_{ij} = \tau_{ij} \times \frac{T'_{ij}}{T_{ij}} \tag{6.6}$$

where T'_{ij} is the real/measured throughput AP_j to UE_i while T_{ij} is the apportioned throughput. From the analysis for Eq. (3.5), we know that the time share of a UE under a PF scheduled AP (τ_{ij}) is $1/K$. Therefore,

$$\tau^*_{ij} = \frac{1}{K} \times \frac{T'_{ij}}{T_{ij}}$$

The part of time share not used by that UE (u_{ij}) can be expressed as,

$$u_{ij} = \tau_{ij} \times (1 - \frac{T'_{ij}}{T_{ij}}) = \frac{1}{K} \times (1 - \frac{T'_{ij}}{T_{ij}})$$

Let us denote the set of inelastic traffic UEs on AP_j as $\mathbb{I}_j$ while that of elastic traffic UEs as the $\mathbb{E}_j$. The total of unused fractional time share (U_j) can be calculated as,

$$U_j = \sum_{i \in \mathbb{I}_j} u_{ij} = \sum_{i \in \mathbb{I}_j} (\frac{1}{K} \times (1 - \frac{T'_{ij}}{T_{ij}}))$$

The fractional time share taken by the flows that are not low-demand inelastic flows (t_{ni}) can be calculated as,

$$t_{ni} = U_j + \sum_{i \in \mathbb{E}_j} \tau_{ij} \tag{6.7}$$

or

$$t_{ni} = 1 - \sum_{i \in \mathbb{I}_j} \tau^*_{ij} \tag{6.8}$$

Let us denote the number of UEs with inelastic traffic on AP_j ($|\mathbb{I}_j|$) as y. The number of UEs with elastic flows $x = K - y$. The corrected throughputs of the other flows T^*_{ij} can then be calculated as,

$$T^*_{ij} = f(t_{ni}) \times r_{ij} \tag{6.9}$$

where f is the bandwidth sharing function. For PF scheduled AP, f is only related to the number of actively connected UEs. Therefore,

$$T^*_{ij} = \frac{t_{ni} \times r_{ij}}{x} \tag{6.10}$$

6.7.2 Throughput Estimation Correction for Flows under Max-Min Fair Scheduled APs

We can use a similar procedure to conduct the throughput estimation corrections for the flows that are not a low-demand elastic flow on an AP.

We use the same way to calculate the time share used by UE_i with elastic traffic as in Eq. (6.6) except that the τ_{ij} is based on the time sharing formula for max-min fair scheduled APs in Eq. (3.7) here.

$$\tau^*_{ij} = \frac{\frac{1}{r_{ij}}}{\sum_{i \in A_j} \frac{1}{r_{ij}}} \times \frac{T'_{ij}}{T_{ij}}$$

where A_j is the set of UEs connected to AP_j. u_{ij} can be calculated as,

$$u_{ij} = \frac{\frac{1}{r_{ij}}}{\sum_{i \in A_j} \frac{1}{r_{ij}}} \times (1 - \frac{T'_{ij}}{T_{ij}})$$

and,

$$U_j = \sum_{i \in \mathbb{I}_j} u_{ij} = \sum_{i \in \mathbb{I}_j} (\frac{\frac{1}{r_{ij}}}{\sum_{i \in A_j} \frac{1}{r_{ij}}} \times (1 - \frac{T'_{ij}}{T_{ij}}))$$

Then, we can calculate t_{ni} using either Eq. (6.7) or Eq. (6.8). The corrected throughputs of the other flows T^*_{ij} can also be calculated using Eq. (6.9) except the bandwidth sharing function is based on Eq. (3.8). Therefore,

$$\begin{aligned} T^*_{ij} &= f(t_{ni}) \times r_{ij} \\ &= \frac{\frac{t_{ni}}{r_{ij}}}{\sum_{i \in \mathbb{E}_j} \frac{1}{r_{ij}}} \times r_{ij} \\ &= \frac{t_{ni}}{\sum_{i \in \mathbb{E}_j} \frac{1}{r_{ij}}} \end{aligned} \tag{6.11}$$

6.7.3 Solution If Low-Demand Elastic Traffic and Non-Participants Co-Exist in the System

If there were both non-participants and low-demand elastic traffic in the system, UEs may report their demands directly. Then, the mapping systems can do the throughput correction for low-demand elastic traffic followed by the throughput correction for non-participants.

Chapter 7

Evaluations using Dynamic Simulations

In this chapter, we consider potentially more realistic scenarios in which UEs have dynamic behavior. We use the methodology in Section 5.2.2 to conduct event-driven simulations, which simulate both *on-off* session dynamics and UE mobility dynamics. The mapping algorithms tested in the dynamic simulations include:

1. ATOM (atom)
2. Global greedy (gg)
3. Local greedy - equal chance (lge)
4. Local greedy - WiFi preferred (lgw)
5. Load-aware local greedy - equal chance (llg)

Note the most important difference from the list of the algorithms we tested in the static simulation is that we can simulate the llg algorithm now, which requires the simulation of *on/off* events.

We use exponential distribution as the model of session lengths. Both the session lengths of *on* and *off* events follow this distribution with a mean value of 1 nominal time unit. In our evaluation, we run the dynamic simulations for multiple runs with different initial UE placements to

get statistically stabilized mean performance results. The initial placements use the same placement strategies as the strategies used in the static simulations in Section 5.1.3.1, i.e. the uniform and clustered placements. The results shown in this chapter uses 1250 runs for each experiment. Each run has 32768 *on/off* events. As described in Section 5.2.2, we calculate the weighted average of performance metrics for each run. Then we report the statistics of the 1250 weighted averages across the runs.

We show the results when the movement probability (P_m) is equal to 1/4. We have also tested with other probabilities of movement in [0, 1] with a step size of 0.1. The results are basically identical. Therefore, we will only show the results of $P_m = 1/4$ here.

We still use 4 APs in the dynamic simulations, as in the static simulations. Since the *on-off* dynamics reduces the number of online UEs that participates, we increase the total number of UEs to 64. As the means of the duration of the *on* and *off* states are identical, the mean number of UEs in the *on* state is still 32. It is the same as the number of UEs in all the static simulations without the optimal solution (Section 6.2 to Section 6.6). Since the state transition process is $i.i.d$ distributed across all UEs, it is common for short bursts of predominantly *on* transitions and short bursts of predominantly *off* transitions to occur. It has been observed that for approximately 95% of the simulated time the number of UEs in the *on* state is between 24 and 40. But occasionally as few as 16 and as many as 48 UEs have been observed in the *on* state. Since there are 64 UEs, each run consumes a mean simulated time of 32768 / 64 = 512 nominal time units.

In the dynamics simulations, the GIFMS mapping systems perform remapping only at each *on* events. We have also evaluated time-driven remapping for GIFMS. Performance degrades significantly as the rate approaches 1 remapping per nominal time unit. The S-GIFMS mapping system also reschedules at *on* event. However, it only decides and changes the association of the flows that are changing to the *on* state. The LIFMS mapping systems do not change their mapping solutions unless there is a UE movement.

7.1 Dynamic Simulation Results under Uniform UE Topology

Table 7.1 to Table 7.6 show the results of dynamic simulation under the uniform UE topology. For each metric, we always show the table with $P_m = 0$ first followed by that with $P_m = 0.25$. We

have three observations from the tables.

1. The results from the dynamic simulations are very similar to those from the static simulations in terms of all three metrics in Section 6.2. For example, the GPF values of all the algorithms only differ from the values in Table 6.18 by no more than 0.4. The ranking of mapping algorithms remains the same. As the dynamic simulations conduct more remappings, we see the standard deviation decreases as expected.

2. The performance of the load-aware local greedy algorithm (llg) is very close to the performance of the GIFMS algorithms. We see from Tables 7.1 and 7.2, in terms of mean GPF, llg can achieve about 99% of the highest GPF value achieved by the GIFMS algorithms.

3. The mobility does not have a major impact to the results, as the results with and without mobility are statistically the same.

Table 7.1: GPF of Dynamic Simulation (Uniform, Pm=0)

	mean	sd	median	min	max
lgw.v	7.83	2.54	7.82	0.64	16.62
lge.v	20.51	2.98	20.57	11.95	29.19
atom.v	21.54	2.90	21.57	13.04	30.39
gg.v	21.36	2.96	21.40	12.76	30.29
llg.v	21.13	2.91	21.13	12.48	29.95

Table 7.2: GPF of Dynamic Simulation (Uniform, Pm=0.25)

	mean	sd	median	min	max
lgw.v	7.79	2.54	7.81	0.48	16.77
lge.v	20.51	2.98	20.56	11.71	29.27
atom.v	21.54	2.89	21.57	12.98	30.48
gg.v	21.36	2.96	21.39	12.71	30.38
llg.v	21.13	2.91	21.14	12.46	30.03

Table 7.3: Aggregate Throughput (Mbps) of Dynamic Simulation (Uniform, Pm=0)

	mean	sd	median	min	max
lgw.t	43.72	3.47	43.51	34.48	57.67
lge.t	75.32	7.00	75.42	53.73	96.65
atom.t	76.62	6.56	76.81	56.70	97.37
gg.t	76.31	6.69	76.46	55.86	97.13
llg.t	75.10	6.78	75.11	54.72	96.95

Table 7.4: Aggregate Throughput (Mbps) of Dynamic Simulation (Uniform, Pm=0.25)

	mean	sd	median	min	max
lgw.t	43.71	3.47	43.48	34.39	57.81
lge.t	75.32	6.99	75.25	53.64	96.57
atom.t	76.63	6.54	76.84	56.57	97.51
gg.t	76.32	6.67	76.48	55.66	97.68
llg.t	75.11	6.76	75.12	54.31	96.95

Table 7.5: Throughput Fairness Index of Dynamic Simulation (Uniform, Pm=0)

	mean	sd	median	min	max
lgw.TFI	0.90	0.04	0.91	0.63	0.96
lge.TFI	0.64	0.09	0.65	0.33	0.86
atom.TFI	0.74	0.05	0.74	0.57	0.86
gg.TFI	0.75	0.05	0.75	0.60	0.87
llg.TFI	0.73	0.05	0.73	0.56	0.87

Table 7.6: Throughput Fairness Index of Dynamic Simulation (Uniform, Pm=0.25)

	mean	sd	median	min	max
lgw.TFI	0.90	0.04	0.91	0.64	0.96
lge.TFI	0.64	0.09	0.65	0.33	0.87
atom.TFI	0.74	0.05	0.74	0.56	0.86
gg.TFI	0.75	0.05	0.75	0.60	0.86
llg.TFI	0.73	0.05	0.73	0.56	0.87

7.2 Dynamic Simulation Results under Clustered UE Topology

Table 7.7 to Table 7.12 show the results of dynamic simulation under the clustered UE topology. As can be seen in the tables, the values of all three metrics are very similar to those obtained in the static simulations and reported in Section 6.4. Interestingly, we see that llg even obtains sightly better GPF performance compared with ATOM and gg in this scenario, although it achieves marginally lower aggregate throughput. We think it is because llg uses the AP-level proportional fair objective and its changes to the flow mapping each time is limited to the flow that has an *off-on* transition. This provides less opportunity for the heavy flows to take up more resources as compared with the GIFMS using a global GPF objective and a global rescheduling. This helps to boost the fairness metrics, even though slightly decreasing the aggregate throughput. We also see that lgw still has much worse performance compared to the other flow mapping algorithms. It confirms that, in the scenarios with dynamics, the currently-used policy based flow mapping algorithm still has poor performance with much room for improvement.

Table 7.7: GPF of Dynamic Simulation (Cluster, Pm=0)

	mean	sd	median	min	max
lgw.v	-21.71	3.29	-21.78	-30.33	-11.44
lge.v	-11.83	3.73	-11.80	-23.34	0.39
atom.v	6.07	2.41	6.22	-1.80	14.83
gg.v	6.37	2.56	6.54	-2.69	15.07
llg.v	7.32	2.23	7.49	-0.09	15.09

Table 7.8: GPF of Dynamic Simulation (Cluster, Pm=0.25)

	mean	sd	median	min	max
lgw.v	-21.99	3.27	-22.04	-30.79	-11.88
lge.v	-11.88	3.73	-11.82	-23.41	0.18
atom.v	6.02	2.42	6.13	-1.78	14.72
gg.v	6.32	2.56	6.54	-2.44	14.95
llg.v	7.27	2.24	7.41	-0.03	14.97

Table 7.9: Aggregate Throughput (Mbps) of Dynamic Simulation (Cluster, Pm=0)

	mean	sd	median	min	max
lgw.t	21.78	2.83	22.22	12.86	28.57
lge.t	30.41	3.43	30.34	21.11	39.84
atom.t	45.65	2.42	45.70	38.08	54.79
gg.t	47.10	2.60	47.10	37.82	56.10
llg.t	45.58	2.57	45.61	37.55	55.03

Table 7.10: Aggregate Throughput (Mbps) of Dynamic Simulation (Cluster, Pm=0.25)

	mean	sd	median	min	max
lgw.t	21.61	2.80	22.00	12.84	28.61
lge.t	30.44	3.43	30.47	21.13	39.86
atom.t	45.61	2.42	45.65	38.10	54.74
gg.t	47.06	2.60	47.09	37.96	56.08
llg.t	45.55	2.58	45.59	37.54	54.93

Table 7.11: Throughput Fairness Index of Dynamic Simulation (Cluster, Pm=0)

	mean	sd	median	min	max
lgw.TFI	0.44	0.11	0.41	0.29	1.00
lge.TFI	0.46	0.09	0.45	0.24	0.86
atom.TFI	0.75	0.07	0.76	0.54	0.93
gg.TFI	0.78	0.06	0.79	0.58	0.93
llg.TFI	0.84	0.04	0.84	0.66	0.93

Table 7.12: Throughput Fairness Index of Dynamic Simulation (Cluster, Pm=0.25)

	mean	sd	median	min	max
lgw.TFI	0.44	0.11	0.41	0.29	1.00
lge.TFI	0.46	0.09	0.45	0.25	0.86
atom.TFI	0.75	0.07	0.76	0.53	0.92
gg.TFI	0.78	0.06	0.79	0.58	0.93
llg.TFI	0.84	0.04	0.84	0.65	0.92

Chapter 8

Discussion on Possible Implementations for the Mapping Systems

In this chapter, we first discuss common modules needed for all three types of flow mapping systems (LIFMS, GIFMS, and S-GIFMS). We then propose options to implement each type of system. The trade-offs among different types of systems and among the options of each system are also discussed.

8.1 Common Modules

8.1.1 Handover Module

A handover module assists UEs in switching their interfaces for existing flows. It makes sure the existing flows are not disrupted when mapping systems command to switch connections. This reduces the cost of handovers.

All the flow mapping systems in GIFMS require a handover module, as they remap all the UEs periodically and can enforce existing flows to change their connections. LIFMS and S-GIFMS do not require a handover module if they are only triggered by *on* events. However, if they are also

triggered by mobility events, they require a functional handover module too.

We think there are at least three network layers in which the flow mapping systems can implement the handover module.

1. Application Layer

   A scheme similar to [33] can be used. Basically, an HTTP proxy can be added to handle flow handovers.

2. Transport Layer

   The flow mapping systems can use Multipath TCP, which has proved to support handover between LTE and WiFi [36].

3. IP Layer

   We can also use Mobile IP based solution similar to the solution in the 3GPP standard such as the solution in IP Flow Mobility [39].

The trade-offs in the implementation selection of the handover module are as follows. The application layer solution requires neither changes to the network infrastructure nor mobile devices. It only relies on a proxy server which can be set up by a 3rd-party provider other than network providers, and proxy server configurations on mobile devices. The transport layer solution does not require any changes to the network infrastructure. However, it requires network stack upgrades at both the mobile devices and corresponding nodes. The IP layer solution involves changes to the network infrastructure. Depending on the implementation, it may also entail changes to the network stacks on mobile devices.

8.1.2 Information Collection Module

In all three types of flow mapping systems, each UE needs to locally collect information needed as the inputs of the flow mapping algorithms (For the following discussion, when we say all the mapping systems, we will not consider random-assignment. It is because, for random-assignment, each UE only needs to know the total number of accessible APs locally.).

For all the mapping algorithms, there are two types of information that must be collected locally,

1. Information of all the available connections and its radio access technology (RAT) type (LTE or WiFi, and their versions);

2. Information about the connection status of UEs to all the APs in the system. "Not accessible" is one of the statuses. If an AP is reachable by a UE, either received signal strength or MCS index works for this purpose.

There is other information that can be optionally collected such as the measured throughput on the connected interface, as we will mention in Section 8.3.

8.2 Implementation Options for LIFMS

With connection status information locally collected and converted into effective rates of UEs, the local-greedy-equal-chance is easy to implement (local-greedy-wifi-preferred is already in use, even though most current systems do not support handover very well). It only needs to run the algorithms we described in Section 5.3. If any UE's connection needs to be changed, it can enforce them without breaking the existing connections using the schemes in Section 8.1.1.

8.3 Implementation Options for GIFMS

To make GIFMS work, the global information on the connection status between all the UEs and all the APs in the system needs to be aggregated at network components at a certain network level. For example, the information can be aggregated at a top-level server, a set of APs, or all the UEs in the system. The flow mapping algorithms will then be run at the same set of network components. That is why we call it a conceptually centralized approach. However, as the information aggregated into lower levels, more new protocol designs are needed to share and synchronize scheduling information. This is not the focus of this dissertation. Therefore, we assume a centralized flow mapping server in the following description.

UEs or APs must periodically send to the server a message that includes the following fields, (UE_ID, AP_ID, RAT_TYPE_ID, CONNECTION_STATUS_METRICS). UE_ID and AP_ID are unique IDs for UEs and APs respectively. The RAT_TYPE_ID and CONNECTION_STATUS_METRICS correspond to the two types of locally collected information we described in Section 8.1.2. The server

can convert the CONNECTION_STATUS_METRICS for each RAT type to an effective rate using the method we described in Section 4.6.

After collecting information from the received messages, the server runs the flow mapping algorithms and sends the association plans back to UEs. UEs then enforce the association plan using the handover module.

8.3.1 Some Optional Optimizations

Usually, more information can be achieved from one UE's current connection compared to a potential connection. We think at least the following two types of information can be obtained and used for some optional optimizations to the flow mapping systems.

1. Measured throughputs of the current connections

   This information can be helpful for the optimization when there are non-participants in the system as described in Section 6.6.3.

2. MCS indices used by the current connections

   The SINR-to-MCS-Index mapping can be inaccurate for a specific site since the mapping tables in the deployment can be different from the mapping tables derived from fading models. If MCS indices of the current connections can be collected and reported together with the SINRs of those connections, a site-specific SINR-to-MCS-Index mapping can be learned from the reported information, which can alleviate the impact of the error aforementioned.

8.4 Implementation Options for S-GIFMS

We have already discussed that certain AP level information should be monitored and broadcast to UEs for the S-GIFMS to work in Section 5.3. Since an S-GIFMS does not require UEs to send report messages and receive association plans, it has less control overhead compared with GIFMS. Additionally, as analyzed in the description of llg in Section 5.3, it does not have any enforced handover as GIFMS if only triggered by *on* events. Therefore, it can work without the implementation of a handover module.

We discuss how the information can be monitored and broadcast for PF scheduled APs and max-min scheduled APs with various RATs.

8.4.1 Information Monitoring

It is trivial for APs to monitor the number of UEs connected to it. For the round time required by max-min fair scheduled APs, the APs can record the timestamps at the beginning and end of each round. The difference between the two timestamps will be the round time.

8.4.2 Information Broadcasting

For an 802.11 AP, it only requires adding AP level information to the broadcast beacons of the AP. Those beacons can be received by UEs even if they have been associated with some other 802.11 APs.

For cellular APs, if the system contains only APs from one provider, the APs can broadcast the information via either Master Information Block (MIB) or System Information Block (SIB) messages [7]. If there were multiple providers, the schemes described in the Section 7.2 of [16] can be used.

Alternatively, APs can report this information to an ANDSF [1] server which can then send the required information to UEs. However, this will make the S-GIFMS have the same overhead for sending association plan messages to UEs as GIFMS, while only saving the control messages of UEs' reporting.

Chapter 9

Conclusion and Future Work

9.1 Conclusion

We make the following conclusions from the conducted evaluations:

1. Local-greedy-wifi-preferred is far less effective than all the other mapping systems in terms of both generalized proportional fairness and aggregate throughput, in most environments tested. Local-greedy-equal-chance, which provides equal opportunity to LTE and WiFi, has a significant performance improvement compared to the WiFi-preferred version. This means, for both network providers and individual users, if the information about local connection status and AP scheduling types is properly utilized, even local greedy flow mapping systems giving LTE and WiFi equal opportunity can greatly improve system performance. This improvement requires neither changes to the network infrastructure nor additional scheduling servers. It incurs the least control overhead. However, its performance is not as consistent as GIFMS and S-GIFMS under various circumstances.

2. Local-greedy-equal-chance has performance close to that of GIFMS and S-GIFMS when UEs are distributed more evenly relative to AP coverages. However, its performance in terms of both GPF and aggregate throughput is significantly inferior to that of GIFMS and S-GIFMS under scenarios when there are imbalanced loads among APs. The imbalanced loads can be introduced by UEs clustered around hot spots.

3. GIFMS have more consistent GPF and aggregate throughput improvement with lower standard

deviations under all the environments tested compared with the local greedy mapping systems. The performance of ATOM and global-greedy is very close to the optimal solution in terms of both GPF and aggregate throughput. However, they require an additional conceptually centralized server and impose more control and handover overhead.

4. S-GIFMS can, on average, achieve similar performance as the GIFMS under all the settings in terms of both GPF value and aggregate throughput. It also does not have any enforced handover and requires less overhead. However, some minor modifications to the broadcasted beacons for all the APs are needed to implement it. We have identified the minimum information needed for both proportional and max-min fair scheduled APs. The suggested minimum information can serve as a guide for the next generation HetNets.

Executive Summary: From the evaluations, we observe the following trade-offs a provider should consider when deploying a flow mapping system for a HetNet. Both the flow mapping systems using local and global information can work with the current wireless systems. The local information based ones require fewer system changes and overhead. However, they suffer from inconsistent system performance improvement under various scenarios. The AP-level information based flow mapping system with the implementation we propose, even though not applicable to the current wireless systems, is ideal for the next generation HetNets in which APs can be modified to monitor and broadcast certain scheduling information related to optimization objectives. The shared information can help mobile devices to make better AP association decisions incorporating the AP load information, when compared with purely local information based decisions. It can achieve performance close to the flow mapping systems using global information while having low overhead and deployment cost comparing to the global information based ones. The identified minimum scheduling information that must be monitored and shared by APs provide important guidance for the minimum information sharing in the next generation HetNets.

9.2 Future Work

Several interesting future directions of this research include:

1. To use other types of fairness as the objective;

2. To evaluate non-elastic traffic;

3. To evaluate the scenarios when flow splitting is allowed;

4. To evaluate the impact of an increasing level of upstream traffic;

5. To evaluate other types of session dynamics and mobility patterns;

6. To evaluate the impact of not-fully-synced information data as the mapping systems become more distributed;

7. To evaluate the impact of various *on-off* session characteristics to various rescheduling policies for S-GIFMS;

8. To define and evaluate the security and authentication mechanisms required for new control messages for GIFMS and S- GIFMS.

Appendices

Appendix A Circular Cluster Detailed Test Result Summary

Pb=0

	mean	sd	median	min	max
lgw.v	-18.55	5.25	-18.49	-37.57	-0.20
lge.v	-10.78	5.44	-10.73	-32.49	9.22
atom.v	6.52	3.51	6.65	-10.82	18.35
gg.v	6.82	3.69	7.00	-11.75	18.41
rand.v	-10.74	5.09	-10.59	-33.28	6.13

Pb=0.1

	mean	sd	median	min	max
lgw.v	-13.13	5.61	-13.01	-34.27	12.78
lge.v	-4.64	6.16	-4.51	-30.95	20.85
atom.v	8.77	3.68	8.83	-6.12	24.83
gg.v	9.00	3.79	9.14	-6.24	24.73
rand.v	-8.73	5.34	-8.57	-32.67	12.09

Pb=0.2

	mean	sd	median	min	max
lgw.v	-8.62	5.62	-8.44	-35.00	10.36
lge.v	0.53	6.21	0.70	-24.89	24.97
atom.v	10.99	3.92	11.03	-6.71	28.25
gg.v	11.08	3.96	11.20	-5.99	27.66
rand.v	-7.22	5.60	-6.97	-29.62	15.87

Pb=0.3

	mean	sd	median	min	max
lgw.v	-4.78	5.41	-4.64	-33.61	14.22
lge.v	4.92	6.14	5.15	-21.92	25.96
atom.v	13.09	4.08	13.10	-4.82	28.68
gg.v	13.05	4.10	13.10	-4.82	28.80
rand.v	-5.81	5.66	-5.63	-27.78	13.13

Pb=0.4

	mean	sd	median	min	max
lgw.v	-1.53	5.19	-1.33	-26.07	16.57
lge.v	8.74	5.81	8.92	-15.95	29.50
atom.v	15.00	4.18	15.03	-2.06	30.11
gg.v	14.86	4.20	14.92	-3.03	29.51
rand.v	-4.53	5.81	-4.31	-32.68	17.22

Pb=0.5

	mean	sd	median	min	max
lgw.v	1.20	4.88	1.31	-20.35	19.59
lge.v	11.93	5.50	12.05	-15.96	37.92
atom.v	16.68	4.23	16.67	-0.98	38.20
gg.v	16.49	4.25	16.48	-2.08	38.20
rand.v	-3.53	5.91	-3.22	-27.87	17.10

Table 1: Comparison of the GPF Value with Different Pb (I)

Pb=0.6

	mean	sd	median	min	max
lgw.v	3.57	4.54	3.62	-15.45	20.54
lge.v	14.73	5.21	14.87	-10.63	32.29
atom.v	18.25	4.26	18.27	-0.28	33.71
gg.v	18.03	4.30	18.06	-0.31	33.52
rand.v	-2.70	5.97	-2.46	-32.21	15.72

Pb=0.7

	mean	sd	median	min	max
lgw.v	5.43	4.28	5.51	-11.86	21.00
lge.v	17.07	4.95	17.25	-6.07	37.30
atom.v	19.61	4.31	19.63	3.10	37.65
gg.v	19.40	4.37	19.43	2.52	37.65
rand.v	-1.95	6.08	-1.71	-35.45	19.76

Pb=0.8

	mean	sd	median	min	max
lgw.v	6.74	4.00	6.74	-10.05	22.52
lge.v	18.74	4.68	18.85	-1.94	37.65
atom.v	20.57	4.26	20.62	4.49	38.14
gg.v	20.36	4.34	20.40	3.24	37.95
rand.v	-1.63	6.07	-1.40	-30.76	20.02

Pb=0.9

	mean	sd	median	min	max
lgw.v	7.72	3.80	7.65	-7.43	24.95
lge.v	20.15	4.53	20.15	2.70	35.75
atom.v	21.45	4.27	21.42	5.72	36.85
gg.v	21.25	4.36	21.24	4.94	36.69
rand.v	-1.11	6.20	-0.83	-35.54	18.80

Pb=1

	mean	sd	median	min	max
lgw.v	8.14	3.70	8.11	-5.73	22.66
lge.v	20.84	4.44	20.83	4.51	38.54
atom.v	21.88	4.26	21.84	5.50	38.86
gg.v	21.69	4.34	21.67	5.40	38.73
rand.v	-1.20	6.23	-0.92	-31.47	18.81

Table 2: Comparison of the GPF Value with Different Pb (II)

Pb=0

	mean	sd	median	min	max
lgw.t	23.79	4.58	24.79	9.89	37.74
lge.t	31.24	5.53	31.17	11.59	50.16
atom.t	45.79	3.77	45.65	24.70	58.25
gg.t	47.16	4.08	47.07	29.21	60.65
rand.t	30.81	4.91	31.38	14.57	52.52

Pb=0.1

	mean	sd	median	min	max
lgw.t	33.21	9.61	31.35	10.97	94.85
lge.t	48.87	15.18	46.00	12.16	135.84
atom.t	53.36	9.50	50.88	32.08	108.70
gg.t	52.05	7.25	50.93	33.24	103.21
rand.t	36.17	8.81	34.83	15.12	94.94

Pb=0.2

	mean	sd	median	min	max
lgw.t	37.21	8.91	35.40	10.72	96.46
lge.t	57.77	15.18	55.79	15.77	144.73
atom.t	58.58	10.22	56.64	34.84	108.64
gg.t	56.64	8.82	54.94	35.02	108.49
rand.t	39.60	9.71	37.76	15.03	101.20

Pb=0.3

	mean	sd	median	min	max
lgw.t	39.50	8.03	37.99	11.19	88.06
lge.t	62.96	14.47	61.55	20.94	132.70
atom.t	62.83	10.41	61.64	37.40	120.83
gg.t	60.82	9.59	59.21	35.85	107.24
rand.t	42.27	9.84	40.32	14.94	99.83

Pb=0.4

	mean	sd	median	min	max
lgw.t	40.69	7.15	39.52	17.25	89.69
lge.t	66.10	13.52	64.79	26.50	144.83
atom.t	66.09	10.28	65.38	36.28	111.13
gg.t	64.37	9.97	63.11	36.21	112.13
rand.t	44.23	9.74	42.30	16.45	95.85

Pb=0.5

	mean	sd	median	min	max
lgw.t	41.40	6.45	40.46	20.81	79.00
lge.t	68.30	12.74	67.18	24.84	136.27
atom.t	68.59	10.09	68.00	38.05	119.65
gg.t	67.20	10.02	66.21	37.56	119.65
rand.t	45.48	9.41	43.77	17.26	101.07

Table 3: Comparison of the Aggregate Throughput (Mbps) with Different Pb (I)

Pb=0.6

	mean	sd	median	min	max
lgw.t	42.12	6.01	41.30	23.64	78.70
lge.t	70.07	12.16	69.16	32.62	138.16
atom.t	70.93	10.01	70.43	40.32	117.88
gg.t	69.93	10.10	69.20	40.10	114.37
rand.t	46.47	9.02	44.87	23.21	95.93

Pb=0.7

	mean	sd	median	min	max
lgw.t	42.72	5.68	42.14	23.15	79.14
lge.t	71.93	11.66	71.02	33.10	131.14
atom.t	73.02	10.09	72.52	41.99	120.41
gg.t	72.29	10.27	71.79	43.56	120.36
rand.t	47.41	8.73	45.99	20.03	101.10

Pb=0.8

	mean	sd	median	min	max
lgw.t	43.04	5.46	42.51	27.81	79.79
lge.t	72.99	11.27	72.29	36.31	133.05
atom.t	74.39	10.02	74.02	44.89	116.31
gg.t	73.86	10.20	73.41	44.61	119.78
rand.t	47.78	8.41	46.53	21.63	99.67

Pb=0.9

	mean	sd	median	min	max
lgw.t	43.45	5.34	42.99	27.89	71.67
lge.t	74.32	11.00	73.70	40.98	136.65
atom.t	75.78	10.12	75.42	42.75	116.45
gg.t	75.44	10.31	74.94	41.77	120.78
rand.t	48.57	8.35	47.43	23.87	91.03

Pb=1

	mean	sd	median	min	max
lgw.t	43.71	5.22	43.30	29.66	69.80
lge.t	75.28	10.78	74.70	41.67	127.37
atom.t	76.62	9.99	76.32	41.81	116.83
gg.t	76.29	10.22	75.86	40.41	121.63
rand.t	48.65	8.23	47.56	22.28	94.31

Table 4: Comparison of the Aggregate Throughput (Mbps) with Different Pb (II)

Pb=0

	mean	sd	median	min	max
lgw.TFI	0.43	0.21	0.36	0.12	1.00
lge.TFI	0.45	0.16	0.42	0.10	1.00
atom.TFI	0.75	0.12	0.78	0.35	0.99
gg.TFI	0.78	0.10	0.80	0.37	0.98
rand.TFI	0.52	0.16	0.48	0.20	0.99

Pb=0.1

	mean	sd	median	min	max
lgw.TFI	0.38	0.18	0.37	0.05	1.00
lge.TFI	0.33	0.16	0.32	0.05	1.00
atom.TFI	0.65	0.21	0.71	0.09	0.98
gg.TFI	0.74	0.16	0.78	0.11	0.99
rand.TFI	0.48	0.18	0.47	0.06	0.98

Pb=0.2

	mean	sd	median	min	max
lgw.TFI	0.43	0.18	0.44	0.05	1.00
lge.TFI	0.34	0.15	0.33	0.05	1.00
atom.TFI	0.63	0.19	0.66	0.10	0.97
gg.TFI	0.72	0.16	0.75	0.11	0.99
rand.TFI	0.47	0.18	0.48	0.06	0.95

Pb=0.3

	mean	sd	median	min	max
lgw.TFI	0.51	0.19	0.52	0.06	1.00
lge.TFI	0.39	0.16	0.38	0.06	0.94
atom.TFI	0.63	0.17	0.65	0.11	0.99
gg.TFI	0.70	0.16	0.73	0.11	0.99
rand.TFI	0.48	0.19	0.49	0.07	0.94

Pb=0.4

	mean	sd	median	min	max
lgw.TFI	0.59	0.19	0.61	0.07	1.00
lge.TFI	0.44	0.16	0.44	0.06	0.98
atom.TFI	0.65	0.15	0.67	0.12	0.98
gg.TFI	0.71	0.14	0.73	0.12	0.98
rand.TFI	0.50	0.18	0.52	0.07	0.96

Pb=0.5

	mean	sd	median	min	max
lgw.TFI	0.67	0.18	0.69	0.07	1.00
lge.TFI	0.49	0.16	0.49	0.07	0.97
atom.TFI	0.68	0.13	0.69	0.11	0.97
gg.TFI	0.72	0.12	0.74	0.13	0.98
rand.TFI	0.52	0.18	0.54	0.07	0.98

Table 5: Comparison of the Throughput Fairness Index with Different Pb (I)

Pb=0.6

	mean	sd	median	min	max
lgw.TFI	0.75	0.16	0.77	0.08	1.00
lge.TFI	0.54	0.16	0.54	0.06	0.99
atom.TFI	0.70	0.11	0.71	0.14	0.98
gg.TFI	0.73	0.11	0.74	0.14	0.97
rand.TFI	0.54	0.17	0.57	0.08	0.95

Pb=0.7

	mean	sd	median	min	max
lgw.TFI	0.81	0.14	0.84	0.09	1.00
lge.TFI	0.57	0.16	0.58	0.07	0.98
atom.TFI	0.72	0.10	0.73	0.14	0.97
gg.TFI	0.74	0.10	0.75	0.16	0.97
rand.TFI	0.56	0.16	0.59	0.07	0.96

Pb=0.8

	mean	sd	median	min	max
lgw.TFI	0.86	0.12	0.89	0.10	1.00
lge.TFI	0.60	0.16	0.62	0.07	1.00
atom.TFI	0.73	0.09	0.74	0.19	0.97
gg.TFI	0.75	0.09	0.76	0.19	0.96
rand.TFI	0.58	0.15	0.60	0.08	0.96

Pb=0.9

	mean	sd	median	min	max
lgw.TFI	0.89	0.10	0.92	0.10	1.00
lge.TFI	0.63	0.15	0.65	0.07	1.00
atom.TFI	0.74	0.09	0.74	0.28	0.96
gg.TFI	0.75	0.08	0.76	0.15	0.97
rand.TFI	0.59	0.15	0.61	0.08	0.95

Pb=1

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.93	0.20	1.00
lge.TFI	0.65	0.15	0.66	0.07	1.00
atom.TFI	0.74	0.09	0.74	0.30	0.96
gg.TFI	0.75	0.08	0.76	0.31	0.96
rand.TFI	0.59	0.14	0.61	0.08	0.94

Table 6: Comparison of the Throughput Fairness Index with Different Pb (II)

Appendix B T-Shaped Rectangular Cluster Detailed Test Result Summary

Pb=0

	mean	sd	median	min	max
lgw.v	3.73	4.06	3.79	-12.84	17.91
lge.v	25.15	5.00	25.41	-2.87	41.46
atom.v	30.59	4.08	30.85	10.33	44.20
gg.v	30.52	4.24	30.80	9.65	44.20
rand.v	4.12	5.42	4.21	-19.65	27.83

Pb=0.1

	mean	sd	median	min	max
lgw.v	4.18	3.96	4.23	-14.51	18.16
lge.v	25.51	5.00	25.71	3.88	41.07
atom.v	29.72	4.24	29.93	10.82	42.36
gg.v	29.72	4.36	29.92	9.42	42.43
rand.v	3.56	5.43	3.62	-21.55	23.38

Pb=0.2

	mean	sd	median	min	max
lgw.v	4.68	3.93	4.70	-10.55	19.19
lge.v	25.70	4.99	25.96	6.01	41.91
atom.v	28.93	4.35	29.11	10.36	42.35
gg.v	28.95	4.45	29.16	10.41	42.35
rand.v	3.01	5.54	3.11	-27.25	21.97

Pb=0.3

	mean	sd	median	min	max
lgw.v	5.13	3.86	5.14	-12.65	20.90
lge.v	25.71	4.87	25.90	2.17	43.38
atom.v	28.16	4.39	28.32	10.18	43.77
gg.v	28.17	4.46	28.37	9.32	43.81
rand.v	2.49	5.64	2.66	-21.37	22.70

Pb=0.4

	mean	sd	median	min	max
lgw.v	5.59	3.84	5.59	-10.05	19.73
lge.v	25.39	4.79	25.49	3.20	41.09
atom.v	27.27	4.45	27.38	6.94	41.87
gg.v	27.26	4.52	27.36	5.84	41.87
rand.v	1.86	5.76	2.02	-29.40	21.67

Pb=0.5

	mean	sd	median	min	max
lgw.v	6.04	3.80	6.04	-8.79	22.29
lge.v	25.01	4.69	25.13	5.55	40.97
atom.v	26.47	4.44	26.58	9.03	41.06
gg.v	26.42	4.51	26.53	6.45	41.06
rand.v	1.43	5.77	1.58	-24.83	23.24

Table 7: Comparison of the GPF with Different Pb (I)

Pb=0.6

	mean	sd	median	min	max
lgw.v	6.50	3.74	6.47	-8.37	21.26
lge.v	24.49	4.61	24.55	1.75	40.54
atom.v	25.65	4.44	25.68	6.40	41.18
gg.v	25.57	4.52	25.60	4.89	41.14
rand.v	0.95	5.85	1.16	-27.28	21.57

Pb=0.7

	mean	sd	median	min	max
lgw.v	6.88	3.76	6.82	-8.29	22.56
lge.v	23.76	4.57	23.86	1.73	40.74
atom.v	24.75	4.45	24.84	7.65	41.66
gg.v	24.62	4.53	24.72	7.04	41.66
rand.v	0.47	5.94	0.67	-26.66	21.58

Pb=0.8

	mean	sd	median	min	max
lgw.v	7.35	3.73	7.29	-6.96	22.70
lge.v	22.92	4.49	22.95	4.07	41.01
atom.v	23.83	4.37	23.85	7.42	41.01
gg.v	23.67	4.45	23.69	5.11	41.01
rand.v	-0.11	6.01	0.16	-25.23	20.41

Pb=0.9

	mean	sd	median	min	max
lgw.v	7.80	3.70	7.76	-9.68	21.64
lge.v	22.04	4.47	22.08	4.65	40.27
atom.v	22.97	4.32	23.01	6.38	40.27
gg.v	22.79	4.41	22.86	6.07	40.27
rand.v	-0.60	6.10	-0.29	-35.79	21.09

Pb=1

	mean	sd	median	min	max
lgw.v	8.19	3.70	8.13	-8.51	24.44
lge.v	20.93	4.46	20.99	2.07	37.85
atom.v	21.97	4.27	22.05	6.04	38.09
gg.v	21.78	4.36	21.87	4.83	38.09
rand.v	-1.13	6.19	-0.88	-26.79	18.37

Table 8: Comparison of the GPF with Different Pb (II)

Pb=0

	mean	sd	median	min	max
lgw.t	40.92	5.27	40.33	25.83	61.56
lge.t	92.66	8.20	92.75	57.21	126.27
atom.t	103.39	8.31	103.61	69.15	132.88
gg.t	104.91	8.69	105.14	69.33	133.60
rand.t	51.74	8.91	50.77	27.21	96.05

Pb=0.1

	mean	sd	median	min	max
lgw.t	41.03	5.31	40.50	24.55	74.14
lge.t	93.51	11.76	92.93	52.58	150.60
atom.t	100.71	9.10	100.75	57.88	133.05
gg.t	102.05	9.39	102.17	58.15	135.14
rand.t	51.27	8.86	50.22	23.60	101.09

Pb=0.2

	mean	sd	median	min	max
lgw.t	41.19	5.29	40.69	26.14	75.76
lge.t	92.87	12.51	92.34	52.48	146.73
atom.t	98.27	9.82	98.38	59.58	136.80
gg.t	99.37	10.12	99.56	59.86	135.52
rand.t	50.79	8.82	49.67	26.14	103.58

Pb=0.3

	mean	sd	median	min	max
lgw.t	41.37	5.27	40.77	26.50	71.42
lge.t	91.24	12.12	90.88	51.28	143.80
atom.t	95.70	10.23	95.72	57.64	140.96
gg.t	96.58	10.49	96.61	56.00	145.14
rand.t	50.34	8.77	49.19	23.19	103.62

Pb=0.4

	mean	sd	median	min	max
lgw.t	41.57	5.25	41.02	26.73	73.18
lge.t	89.20	12.01	88.57	50.19	139.09
atom.t	93.00	10.62	92.84	54.19	134.88
gg.t	93.67	10.86	93.53	54.67	134.88
rand.t	49.77	8.63	48.70	26.47	105.99

Pb=0.5

	mean	sd	median	min	max
lgw.t	41.84	5.25	41.31	26.99	75.85
lge.t	87.20	11.68	86.65	46.10	144.44
atom.t	90.48	10.80	90.42	52.76	135.74
gg.t	91.01	10.99	90.99	52.51	137.82
rand.t	49.61	8.61	48.56	21.12	102.99

Table 9: Comparison of the Aggregate Throughput (Mbps) with Different Pb (I)

Pb=0.6

	mean	sd	median	min	max
lgw.t	42.17	5.19	41.71	24.64	67.70
lge.t	85.13	11.39	84.60	43.98	128.68
atom.t	87.90	10.84	87.73	48.51	130.28
gg.t	88.30	10.99	88.09	48.32	130.28
rand.t	49.24	8.35	48.14	26.54	92.25

Pb=0.7

	mean	sd	median	min	max
lgw.t	42.39	5.21	41.90	24.85	70.53
lge.t	82.82	11.17	82.38	45.76	132.47
atom.t	85.20	10.84	85.08	46.05	132.33
gg.t	85.36	11.01	85.20	46.10	132.33
rand.t	49.09	8.40	48.04	27.24	92.65

Pb=0.8

	mean	sd	median	min	max
lgw.t	42.84	5.21	42.36	26.74	72.04
lge.t	80.70	10.99	80.30	37.98	129.87
atom.t	82.56	10.66	82.32	46.22	129.87
gg.t	82.57	10.82	82.37	47.52	129.87
rand.t	48.88	8.29	47.85	24.68	94.62

Pb=0.9

	mean	sd	median	min	max
lgw.t	43.33	5.19	42.93	29.03	68.78
lge.t	78.34	10.90	77.87	40.73	132.21
atom.t	79.83	10.44	79.55	44.19	126.76
gg.t	79.68	10.62	79.39	44.72	125.74
rand.t	48.92	8.31	47.81	24.66	93.21

Pb=1

	mean	sd	median	min	max
lgw.t	43.77	5.21	43.37	28.70	72.59
lge.t	75.46	10.74	74.93	36.43	129.88
atom.t	76.82	10.08	76.52	42.45	119.76
gg.t	76.55	10.27	76.19	42.67	123.74
rand.t	48.72	8.28	47.51	23.78	94.38

Table 10: Comparison of the Aggregate Throughput (Mbps) with Different Pb (II)

Pb=0

	mean	sd	median	min	max
lgw.TFI	0.80	0.13	0.82	0.21	1.00
lge.TFI	0.71	0.11	0.73	0.12	0.94
atom.TFI	0.75	0.07	0.76	0.25	0.96
gg.TFI	0.74	0.08	0.75	0.26	0.96
rand.TFI	0.61	0.11	0.62	0.09	0.95

Pb=0.1

	mean	sd	median	min	max
lgw.TFI	0.82	0.13	0.84	0.08	1.00
lge.TFI	0.69	0.14	0.73	0.15	0.94
atom.TFI	0.75	0.07	0.75	0.24	0.94
gg.TFI	0.74	0.07	0.74	0.25	0.96
rand.TFI	0.61	0.11	0.62	0.09	0.94

Pb=0.2

	mean	sd	median	min	max
lgw.TFI	0.83	0.12	0.86	0.11	1.00
lge.TFI	0.69	0.14	0.73	0.16	0.94
atom.TFI	0.74	0.07	0.74	0.24	0.95
gg.TFI	0.73	0.07	0.74	0.22	0.95
rand.TFI	0.61	0.12	0.62	0.07	0.97

Pb=0.3

	mean	sd	median	min	max
lgw.TFI	0.84	0.12	0.87	0.07	1.00
lge.TFI	0.70	0.13	0.73	0.17	0.94
atom.TFI	0.73	0.07	0.74	0.25	0.96
gg.TFI	0.73	0.07	0.74	0.26	0.95
rand.TFI	0.61	0.12	0.63	0.08	0.97

Pb=0.4

	mean	sd	median	min	max
lgw.TFI	0.86	0.11	0.88	0.10	1.00
lge.TFI	0.70	0.12	0.73	0.15	0.94
atom.TFI	0.73	0.07	0.73	0.20	0.94
gg.TFI	0.73	0.07	0.74	0.20	0.94
rand.TFI	0.61	0.12	0.63	0.08	0.97

Pb=0.5

	mean	sd	median	min	max
lgw.TFI	0.87	0.11	0.90	0.08	1.00
lge.TFI	0.70	0.11	0.72	0.16	0.95
atom.TFI	0.72	0.07	0.73	0.34	0.95
gg.TFI	0.73	0.07	0.73	0.29	0.94
rand.TFI	0.61	0.13	0.63	0.08	0.96

Table 11: Comparison of the Throughput Fairness Index with Different Pb (I)

Pb=0.6

	mean	sd	median	min	max
lgw.TFI	0.88	0.10	0.90	0.10	1.00
lge.TFI	0.70	0.11	0.72	0.07	0.96
atom.TFI	0.72	0.07	0.72	0.18	0.94
gg.TFI	0.73	0.07	0.73	0.17	0.96
rand.TFI	0.61	0.13	0.63	0.09	0.95

Pb=0.7

	mean	sd	median	min	max
lgw.TFI	0.89	0.10	0.91	0.17	1.00
lge.TFI	0.69	0.11	0.70	0.12	0.97
atom.TFI	0.72	0.08	0.72	0.34	0.95
gg.TFI	0.73	0.07	0.73	0.30	0.95
rand.TFI	0.61	0.13	0.62	0.09	0.98

Pb=0.8

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.92	0.16	1.00
lge.TFI	0.67	0.12	0.68	0.07	0.97
atom.TFI	0.72	0.08	0.72	0.30	0.96
gg.TFI	0.73	0.08	0.74	0.27	0.97
rand.TFI	0.60	0.13	0.62	0.09	0.95

Pb=0.9

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.92	0.19	1.00
lge.TFI	0.66	0.13	0.67	0.07	0.97
atom.TFI	0.73	0.08	0.73	0.29	0.96
gg.TFI	0.74	0.08	0.74	0.29	0.97
rand.TFI	0.60	0.14	0.62	0.09	0.96

Pb=1

	mean	sd	median	min	max
lgw.TFI	0.90	0.09	0.93	0.09	1.00
lge.TFI	0.65	0.15	0.66	0.07	0.99
atom.TFI	0.74	0.09	0.74	0.32	0.96
gg.TFI	0.75	0.08	0.76	0.30	0.97
rand.TFI	0.59	0.14	0.62	0.08	0.94

Table 12: Comparison of the Throughput Fairness Index with Different Pb (II)

Appendix C Detailed Comparison Results of the Impacts from Non-Participants

C.1 Clustered UE Topology

We list the detailed results of the impact of non-participants under the cluster topology here for reference.

dRatio=0.0

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-8.58	5.55	-8.44	-29.67	9.59
gg.v	-8.58	5.55	-8.44	-29.67	9.59
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.125

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-5.51	4.75	-5.52	-24.47	11.08
gg.v	-5.44	4.75	-5.44	-24.47	11.06
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.25

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-4.42	4.65	-4.42	-24.41	12.65
gg.v	-4.03	4.54	-4.01	-23.33	12.65
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.375

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-2.94	4.66	-2.90	-23.22	13.08
gg.v	-2.06	4.48	-2.02	-22.81	13.18
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.5

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-0.96	4.52	-0.88	-19.68	15.06
gg.v	0.62	4.35	0.70	-17.78	15.67
rand.v	-10.83	4.94	-10.78	-34.15	7.65

Table 13: Comparison of the GPF with Different dRatio using Throughput Correction (I)

dRatio=0.0

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-8.58	5.55	-8.44	-29.67	9.59
gg.v	-8.58	5.55	-8.44	-29.67	9.59
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.125

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-6.94	5.01	-6.89	-26.95	10.37
gg.v	-6.98	5.16	-6.89	-26.95	10.37
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.25

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-5.43	4.64	-5.41	-23.91	10.37
gg.v	-4.42	4.74	-4.39	-24.27	12.13
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.375

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	-2.57	4.33	-2.55	-21.62	13.08
gg.v	-1.62	4.70	-1.61	-22.15	14.32
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.5

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	0.97	4.29	1.10	-18.67	14.49
gg.v	1.66	4.62	1.77	-17.70	17.82
rand.v	-10.83	4.94	-10.78	-34.15	7.65

Table 14: Comparison of the GPF with Different dRatio (I)

dRatio=0.625

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	1.05	4.24	1.15	-19.45	17.82
gg.v	3.32	4.18	3.41	-17.01	17.65
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.75

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	3.45	4.13	3.56	-15.24	16.67
gg.v	5.80	4.00	6.03	-12.84	18.38
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.875

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	5.60	3.62	5.87	-12.61	17.45
gg.v	7.25	3.58	7.52	-11.54	18.50
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=1.0

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	6.52	3.51	6.65	-10.82	18.35
gg.v	6.82	3.69	7.00	-11.75	18.41
rand.v	-10.83	4.94	-10.78	-34.15	7.65

Table 15: Comparison of the GPF with Different dRatio using Throughput Correction (II)

dRatio=0.625

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	3.94	4.08	4.20	-17.01	16.17
gg.v	4.60	4.39	4.79	-17.01	18.02
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.75

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	5.69	3.66	5.92	-12.84	17.45
gg.v	6.59	3.90	6.93	-12.84	18.38
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=0.875

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	6.36	3.52	6.50	-11.16	17.45
gg.v	7.14	3.60	7.35	-11.71	18.55
rand.v	-10.83	4.94	-10.78	-34.15	7.65

dRatio=1.0

	mean	sd	median	min	max
lgw.v	-18.49	4.86	-18.44	-37.57	-1.08
lge.v	-8.58	5.55	-8.44	-29.67	9.59
atom.v	6.52	3.51	6.65	-10.82	18.35
gg.v	6.82	3.69	7.00	-11.75	18.41
rand.v	-10.83	4.94	-10.78	-34.15	7.65

Table 16: Comparison of the GPF with Different dRatio (II)

dRatio=0.0

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	33.35	5.49	33.26	12.70	55.84
gg.t	33.35	5.49	33.26	12.70	55.84
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.125

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	34.38	4.89	34.15	16.86	57.22
gg.t	34.25	4.87	34.02	16.86	57.22
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.25

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	35.52	4.71	35.33	19.08	53.34
gg.t	35.24	4.69	35.06	18.34	53.34
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.375

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	37.47	4.72	37.87	18.56	55.03
gg.t	37.09	4.74	37.45	17.91	53.51
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.5

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	39.73	4.16	39.96	20.47	55.08
gg.t	39.27	4.23	39.45	20.33	55.55
rand.t	30.55	4.78	31.19	15.14	49.86

Table 17: Comparison of the Aggregate Throughput (Mbps) with Different dRatio using Throughput Correction (I)

dRatio=0.0

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	33.35	5.49	33.26	12.70	55.84
gg.t	33.35	5.49	33.26	12.70	55.84
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.125

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	34.49	4.99	34.26	17.04	56.61
gg.t	33.97	5.28	33.80	17.04	56.61
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.25

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	35.40	4.69	35.19	18.23	55.03
gg.t	35.19	4.89	35.00	17.07	57.22
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.375

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	36.95	4.76	37.24	17.73	55.03
gg.t	37.04	4.92	37.35	17.54	54.46
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.5

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	39.60	4.29	39.79	19.07	54.98
gg.t	39.60	4.62	39.79	20.18	57.07
rand.t	30.55	4.78	31.19	15.14	49.86

Table 18: Comparison of the Aggregate Throughput (Mbps) with Different dRatio (I)

dRatio=0.625

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	40.95	3.67	40.93	21.00	57.07
gg.t	40.72	3.96	40.58	21.15	56.81
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.75

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	41.63	3.69	41.54	23.96	55.37
gg.t	42.46	4.23	42.37	25.73	58.23
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.875

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	42.87	3.62	43.03	25.22	56.69
gg.t	44.90	4.03	45.06	26.30	58.93
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=1.0

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	45.79	3.77	45.65	24.70	58.25
gg.t	47.16	4.08	47.07	29.21	60.65
rand.t	30.55	4.78	31.19	15.14	49.86

Table 19: Comparison of the Aggregate Throughput (Mbps) with Different dRatio using Throughput Correction (II)

dRatio=0.625

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	41.81	3.91	41.90	21.79	55.15
gg.t	41.84	4.44	41.74	21.86	58.23
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.75

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	43.46	3.66	43.65	24.07	56.69
gg.t	44.07	4.38	44.33	25.73	58.93
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=0.875

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	44.68	3.57	44.62	24.07	57.21
gg.t	45.88	4.08	45.87	26.19	59.55
rand.t	30.55	4.78	31.19	15.14	49.86

dRatio=1.0

	mean	sd	median	min	max
lgw.t	23.73	4.40	24.79	9.89	40.47
lge.t	33.35	5.49	33.26	12.70	55.84
atom.t	45.79	3.77	45.65	24.70	58.25
gg.t	47.16	4.08	47.07	29.21	60.65
rand.t	30.55	4.78	31.19	15.14	49.86

Table 20: Comparison of the Aggregate Throughput (Mbps) with Different dRatio (II)

C.2 Uniform UE Topology

We list the detailed results of the impact of non-participants under the uniform UE topology.

dRatio=0.0

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.87	4.44	20.89	5.06	40.18
gg.v	20.87	4.44	20.89	5.06	40.18
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=0.125

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.80	4.37	20.79	4.58	40.03
gg.v	20.83	4.37	20.81	4.58	40.03
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=0.25

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.74	4.33	20.71	4.58	40.03
gg.v	20.82	4.33	20.79	4.58	40.03
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=0.375

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.70	4.31	20.69	5.77	39.91
gg.v	20.83	4.30	20.82	5.73	39.91
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=0.5

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.71	4.29	20.70	5.01	39.59
gg.v	20.89	4.28	20.86	5.73	39.59
rand.v	-1.17	6.24	-0.87	-34.22	20.05

Table 21: Comparison of the GPF with Different dRatio using Throughput Correction (I)

dRatio=0.0

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	20.87	4.44	20.89	4.17	40.18
gg.v	20.87	4.44	20.89	4.17	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=0.125

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	20.67	4.39	20.69	4.17	40.03
gg.v	20.68	4.38	20.69	4.17	38.03
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=0.25

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	20.80	4.36	20.80	4.98	40.18
gg.v	20.78	4.37	20.76	5.07	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=0.375

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.00	4.33	20.99	5.29	40.16
gg.v	20.95	4.34	20.94	5.07	40.16
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=0.5

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.18	4.31	21.18	5.32	40.05
gg.v	21.13	4.33	21.12	4.35	40.05
rand.v	-1.12	6.22	-0.90	-28.00	21.40

Table 22: Comparison of the GPF with Different dRatio (I)

dRatio=0.625

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.77	4.29	20.75	5.47	39.59
gg.v	21.00	4.27	20.99	5.79	39.59
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=0.75

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	20.92	4.29	20.93	5.95	40.05
gg.v	21.21	4.27	21.18	6.42	40.16
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=0.875

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	21.19	4.29	21.18	5.75	39.59
gg.v	21.49	4.28	21.46	6.28	40.17
rand.v	-1.17	6.24	-0.87	-34.22	20.05

dRatio=1.0

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.11	-7.03	22.56
lge.v	20.87	4.44	20.89	5.06	40.18
atom.v	21.93	4.25	21.93	6.53	40.30
gg.v	21.74	4.34	21.72	5.01	40.18
rand.v	-1.17	6.24	-0.87	-34.22	20.05

Table 23: Comparison of the GPF with Different dRatio using Throughput Correction (II)

dRatio=0.625

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.37	4.29	21.35	6.21	40.18
gg.v	21.29	4.33	21.28	5.78	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=0.75

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.56	4.27	21.55	6.42	40.06
gg.v	21.45	4.32	21.45	5.63	40.06
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=0.875

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.75	4.26	21.73	6.53	40.18
gg.v	21.59	4.33	21.60	5.76	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

dRatio=1.0

	mean	sd	median	min	max
lgw.v	8.16	3.68	8.13	-7.73	22.56
lge.v	20.87	4.44	20.89	4.17	40.18
atom.v	21.93	4.25	21.93	6.53	40.30
gg.v	21.74	4.34	21.72	5.01	40.18
rand.v	-1.12	6.22	-0.90	-28.00	21.40

Table 24: Comparison of the GPF with Different dRatio (II)

dRatio=0.0

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	75.47	10.83	74.77	38.36	133.51
gg.t	75.47	10.83	74.77	38.36	133.51
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=0.125

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	74.44	10.60	73.81	40.61	124.57
gg.t	74.43	10.57	73.79	39.65	124.57
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=0.25

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	74.35	10.62	73.71	42.65	124.57
gg.t	74.52	10.63	73.91	42.95	124.57
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=0.375

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	74.57	10.67	73.92	42.66	120.28
gg.t	74.87	10.70	74.24	43.09	120.28
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=0.5

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	74.98	10.78	74.28	42.33	123.14
gg.t	75.29	10.80	74.77	44.13	122.70
rand.t	48.81	8.36	47.61	23.88	96.92

Table 25: Comparison of the Aggregate Throughput (Mbps) with Different dRatio using Throughput Correction (I)

dRatio=0.0

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	75.46	10.83	74.80	37.33	133.51
gg.t	75.46	10.83	74.80	37.33	133.51
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=0.125

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	74.49	10.62	73.90	42.05	133.51
gg.t	74.56	10.62	73.96	41.46	133.51
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=0.25

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	74.30	10.46	73.74	41.64	123.92
gg.t	74.33	10.53	73.73	42.75	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=0.375

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	74.34	10.36	73.82	43.48	119.12
gg.t	74.30	10.45	73.73	44.04	119.12
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=0.5

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	74.44	10.29	73.93	43.93	120.62
gg.t	74.46	10.42	73.87	43.45	120.62
rand.t	48.75	8.25	47.67	23.58	93.06

Table 26: Comparison of the Aggregate Throughput (Mbps) with Different dRatio (I)

dRatio=0.625

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	75.44	10.85	74.88	42.33	125.79
gg.t	75.64	10.83	75.08	43.12	122.70
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=0.75

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	75.93	10.87	75.43	42.89	123.14
gg.t	75.82	10.75	75.35	44.13	119.12
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=0.875

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	76.36	10.80	75.97	42.89	122.70
gg.t	76.02	10.55	75.55	43.59	124.88
rand.t	48.81	8.36	47.61	23.88	96.92

dRatio=1.0

	mean	sd	median	min	max
lgw.t	43.70	5.20	43.23	28.96	68.25
lge.t	75.47	10.83	74.77	38.36	133.51
atom.t	76.71	10.01	76.36	46.01	125.29
gg.t	76.46	10.23	76.01	44.52	123.92
rand.t	48.81	8.36	47.61	23.88	96.92

Table 27: Comparison of the Aggregate Throughput (Mbps) with Different dRatio using Throughput Correction (II)

dRatio=0.625

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	74.74	10.24	74.14	43.93	123.92
gg.t	74.72	10.37	74.18	43.87	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=0.75

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	75.17	10.18	74.61	44.95	120.27
gg.t	75.13	10.32	74.60	44.35	120.27
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=0.875

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	75.86	10.11	75.39	44.63	123.92
gg.t	75.71	10.27	75.21	44.35	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

dRatio=1.0

	mean	sd	median	min	max
lgw.t	43.70	5.21	43.24	28.78	71.92
lge.t	75.46	10.83	74.80	37.33	133.51
atom.t	76.71	10.01	76.36	46.01	125.29
gg.t	76.46	10.23	76.01	44.52	123.92
rand.t	48.75	8.25	47.67	23.58	93.06

Table 28: Comparison of the Aggregate Throughput (Mbps) with Different dRatio (II)

Bibliography

[1] Access network discovery and selection function (ANDSF) management object (MO). In *3GPP TS 24.312 Rel 12 Version 12.9.0*.

[2] 802.11n standard. `https://standards.ieee.org/standard/802_11n-2009.html`, accessed 2019.

[3] About Wi-Fi assist. `https://support.apple.com/en-us/HT205296`, accessed 2019.

[4] Evolved universal terrestrial radio access (E-UTRA); physical layer procedures. `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2427`, accessed 2019.

[5] How iOS decides which wireless network to auto-join. `https://support.apple.com/en-us/HT202831`, accessed 2019.

[6] 3GPP. Multiple access PDN connectivity. http://www.3gpp.org/DynaReport/WiCr–430035.htm, 2016.

[7] 3GPP. 3GPP TS 38.331 - NR; radio resource control (RRC); protocol specification. `http://www.3gpp.org/ftp//Specs/archive/38_series/38.331/38331-f70.zip`, 2019.

[8] R. Amin, J. Martin, J. Deaton, L.A. DaSilva, A. Hussien, and A. Eltawil. Balancing spectral efficiency, energy consumption, and fairness in future heterogeneous wireless systems with reconfigurable devices. *Selected Areas in Communications, IEEE Journal on*, 31(5):969–980, May 2013.

[9] Ehsan Aryafar, Alireza Keshavarz-Haddad, Michael Wang, and Mung Chiang. RAT selection games in HetNets. In *Proc. of INFOCOM*, 2013.

[10] Y. Bejerano, Seung-Jae Han, and Li Li. Fairness and load balancing in wireless LANs using association control. *Networking, IEEE/ACM Transactions on*, 15(3):560–573, June 2007.

[11] Paul Bender, Peter Black, Matthew Grob, Roberto Padovani, Nagabhushana Sindhushyana, and Andrew Viterbi. CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users. *Communications Magazine, IEEE*, 38(7):70–77, 2000.

[12] Dimitris Bertsimas, Vivek F Farias, and Nikolaos Trichakis. The price of fairness. *Operations research*, 59(1):17–31, 2011.

[13] T. Bu, Li Li, and R. Ramjee. Generalized proportional fair scheduling in third generation wireless data networks. In *Proc. of INFOCOM*, pages 1–12, April 2006.

[14] Etienne F. Chaponniere, Peter J. Black, Jack M. Holtzman, and David Ngar Ching Tse. Transmitter directed, multiple receiver system using path diversity to equitably maximize throughput, 10 2002. US Patent 6449490.

[15] P. Coucheney, C. Touati, and B. Gaujal. Fair and efficient user-network association algorithm for multi-technology wireless networks. In *INFOCOM 2009, IEEE*, pages 2811–2815, April 2009.

[16] Supratim Deb, Kanthi Nagaraj, and Vikram Srinivasan. MOTA: Engineering an operator agnostic mobile service. In *Proc. of MobiCom*, pages 133–144, New York, NY, USA, 2011. ACM.

[17] Shuo Deng, Anirudh Sivaraman, and Hari Balakrishnan. All your network are belong to us: A transport framework for mobile network selection. In *Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*, page 19. ACM, 2014.

[18] N. A. Elmosilhy, M. M. Elmesalawy, and A. M. Abd Elhaleem. User association with mode selection in LWA-based multi-RAT HetNet. *IEEE Access*, 7:158623–158633, 2019.

[19] ETSI. Base station (BS) radio transmission and reception, 3GPP TS 36.104 release 14, 2017.

[20] D. Fooladivanda and C. Rosenberg. Joint resource allocation and user association for heterogeneous wireless cellular networks. *Wireless Communications, IEEE Transactions on*, 12(1):248–257, January 2013.

[21] Amitabha Ghosh, Nitin Mangalvedhe, Rapeepat Ratasuk, Bishwarup Mondal, Mark Cudak, Eugene Visotsky, Timothy A Thomas, Jeffrey G Andrews, Ping Xia, Han Shin Jo, et al. Heterogeneous cellular networks: From theory to practice. *IEEE communications magazine*, 50(6):54–64, 2012.

[22] Raj Jain, Dah-Ming Chiu, and William R Hawe. *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*, volume 38. 1984.

[23] B. Jin, S. Kim, D. Yun, H. Lee, W. Kim, and Y. Yi. Aggregating LTE and Wi-Fi: Toward intra-cell fairness and high tcp performance. *IEEE Transactions on Wireless Communications*, 16(10):6295–6308, Oct 2017.

[24] B. H. Jung, N. Song, and D. K. Sung. A network-assisted user-centric wifi-offloading model for maximizing per-user throughput in a heterogeneous network. *IEEE Transactions on Vehicular Technology*, 63(4):1940–1945, May 2014.

[25] Jean-Marc Kelif, Stephane Senecal, Marceau Coupechoux, and Constant Bridon. Analytical performance model for poisson wireless networks with pathloss and shadowing propagation. In *2014 IEEE Globecom Workshops (GC Wkshps)*, pages 1528–1532. IEEE, 2014.

[26] Frank P Kelly, Aman K Maulloo, and David KH Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research society*, pages 237–252, 1998.

[27] A. Keshavarz-Haddad, E. Aryafar, M. Wang, and M. Chiang. HetNets selection by clients: Convergence, efficiency, and practicality. *IEEE/ACM Transactions on Networking*, 25(1):406–419, Feb 2017.

[28] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi. A tutorial on IEEE 802.11ax high efficiency WLANs. *IEEE Communications Surveys Tutorials*, 21(1):197–216, Firstquarter 2019.

[29] J Nicholas Laneman, David NC Tse, and Gregory W Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *Information Theory, IEEE Transactions on*, 50(12):3062–3080, 2004.

[30] Jin Seong Lee and Jaiyong Lee. Multipath TCP performance improvement in mobile network. In *2015 Seventh International Conference on Ubiquitous and Future Networks*, pages 710–714. IEEE, 2015.

[31] Y. Lin, W. Bao, W. Yu, and B. Liang. Optimizing user association and spectrum allocation in HetNets: A utility perspective. *IEEE Journal on Selected Areas in Communications*, 33(6):1025–1039, June 2015.

[32] R. Liou, Y. Lin, and S. Tsai. An investigation on LTE mobility management. *IEEE Transactions on Mobile Computing*, 12(1):166–176, Jan 2013.

[33] Rajesh Mahindra, Hari Viswanathan, Karthik Sundaresan, Mustafa Y. Arslan, and Sampath Rangarajan. A practical traffic management system for integrated LTE-WiFi networks. In *Proc. of MobiCom*, pages 189–200. ACM, 2014.

[34] Motorola. BS and MCS signaling and tables. https://www.3gpp.org/DynaReport/TDocExMtg–R1-52b–26911.htm.

[35] Pavan Nuggehalli. LTE-WLAN aggregation [industry perspectives]. *IEEE Wireless Communications*, 23(4):4–6, 2016.

[36] Christoph Paasch, Gregory Detal, Fabien Duchene, Costin Raiciu, and Olivier Bonaventure. Exploring mobile/WiFi handover with multipath tcp. In *Proc. of CellNet*, pages 31–36, New York, NY, USA, 2012. ACM.

[37] M. Peng, G. He, L. Wang, and C. Kai. AP selection scheme based on achievable throughputs in SDN-enabled WLANs. *IEEE Access*, 7:4763–4772, 2019.

[38] Allen L Ramaboli, Olabisi E Falowo, and Anthony H Chan. Bandwidth aggregation in heterogeneous wireless networks: A survey of current approaches and issues. *Journal of Network and Computer Applications*, 35(6):1674–1690, 2012.

[39] CB Sankaran. Data offloading techniques in 3GPP Rel-10 networks: A tutorial. *Communications Magazine, IEEE*, 2012.

[40] A. Sridharan, R.K. Sinha, R. Jana, Bo Han, K.K. Ramakrishnan, N.K. Shankaranarayanan, and I. Broustis. Multi-path TCP: Boosting fairness in cellular networks. In *Proc. of ICNP*, pages 275–280, Oct 2014.

[41] Wei Wang, Xin Liu, J. Vicente, and P. Mohapatra. Integration gain of heterogeneous WiFi/WiMAX networks. *Mobile Computing, IEEE Transactions on*, 10(8):1131–1143, Aug 2011.

[42] Dionysis Xenakis, Nikos Passas, Lazaros Merakos, and Christos Verikoukis. Mobility management for femtocells in LTE-advanced: Key aspects and survey of handover decision algorithms. *IEEE Communications surveys & tutorials*, 16(1):64–91, 2013.

[43] Qiaoyang Ye, Beiyu Rong, Yudong Chen, M. Al-Shalash, C. Caramanis, and J.G. Andrews. User association for load balancing in heterogeneous cellular networks. *Wireless Communications, IEEE Transactions on*, 12(6):2706–2716, June 2013.