

Article

An Adjusted Free-Market-Inspired Approach to Mitigate Free-Riding Behavior in Peer-to-Peer Fog Computing

Heba Kurdi ^{1,2,*} , Alhanoof Althnian ³ , Mohammad Abdulghani ⁴  and Sarah Alkharji ¹ 

¹ Computer Science Department, College of Computer and Information Sciences, King Saud University, Riyadh 11451, Saudi Arabia; 435920197@student.ksu.edu.sa

² Mechanical Engineering Department, Massachusetts Institute of Technology (MIT), Cambridge, MA 02142-1308, USA

³ Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh 11451, Saudi Arabia; aalthnian@ksu.edu.sa

⁴ Mechanical Engineering Department, College of Engineering, King Saud University, Riyadh 11451, Saudi Arabia; 433106773@student.ksu.edu.sa

* Correspondence: hkurdi@ksu.edu.sa or hakurdi@mit.edu

Received: 8 November 2020; Accepted: 25 November 2020; Published: 30 November 2020



Abstract: Peer-to-peer (P2P) architecture is increasingly gaining attention as a potential solution for the scalability problem facing the Internet of Things (IoT). It can be adopted for the fog computing layer to sustain the massive flow of data from constrained IoT nodes to the cloud. The success of a P2P-based system is entirely dependent on the continuity of resource sharing among individual nodes. Free riding is a severe problem that contradicts this main principle of P2P systems. It is understood that peers tend to consume resources from other peers without offering any in return. This free riding behavior can decrease system scalability and content availability, resulting in a decline in performance. Significant efforts have been made to hinder this behavior and to encourage cooperation amongst peers. To this end, we propose AFMIA, an Adjusted Free-Market-Inspired Approach that considers resources as goods that have dynamic prices based on the amount of supply and demand. Peers have wealth that can be increased by providing resources and spent by consuming them. The experimental results indicate that the proposed algorithm can successfully improve fairness without compromising on success rates.

Keywords: fog computing; peer-to-peer; free riding; free market; economy; cloud computing

1. Introduction

Meeting the challenges presented by Internet of Things (IoT) mandates an architecture that can cater for the massive flow of data from constrained edge devices to fog gateways to cloud services. Peer-to-peer (P2P) architecture can be exploited in fog computing to achieve the required scalability and availability, particularly at the edges of the Internet [1]. The P2P architecture has recently become a popular paradigm for building distributed systems in different domains such as file-sharing networks, cloud computing and more recently fog computing. The P2P architecture provides many advantages over the traditional client server architecture, such as high scalability, fault-tolerance, resilience and effectiveness in coping with dynamics and heterogeneity. However, the success of a P2P based system depends on the voluntary participation of its peers (nodes), which assumes that all peers will be active and share their resources [2]. In reality, most peers act in a selfish manner by trying to maximize their own benefit without considering the interests of others. The reason for such behavior is that offering a resource has no direct benefit to the peer which leads to a serious problem known as free riding [3,4].

Free riders are peers that exploit the resources donated by others without offering anything in return [5]. This results in a decrease in the diversity of resources and subsequently the system collapses [6–8]. In addition, free riding increases the response time for a user request because a small number of altruistic peers provide resources to a large number of peers. Therefore, a wide range of research efforts have proposed different approaches to detect and prevent free riding behavior [9–11]. However, the problem is still demanding and no approach proved to be successful in providing fairness among all peers [10]. The purpose of this paper is to mitigate free-riding behavior in P2P systems and ensure fairness for all peers by introducing AFMIA, a free-market-inspired approach.

An idealized free market economy is an economic system in which pricing for goods and services are determined merely by the forces of demand and supply [12]. Demand defines the amount of products that consumers are willing to buy at a particular price. The law of demand suggests that as the price of particular goods increases, the amount of demand decreases (inversely correlated). Supply defines the amount of products available in the market. The law of supply states that as the price of a particular good increases, the supply amount increases [13]. When the amount of supply and demand is equal, the market is considered to have reached equilibrium, and the corresponding price is called the equilibrium price or market-clearing price. AFMIA considers each resource as a product with a dynamic price that changes over time based on the amount of supply and demand. Also, each peer has wealth based on the credits that the peer owns, which can be earned (or spent) when providing (or consuming) resources.

The main contributions of this paper can be summarized as follows:

1. A novel free-market-inspired approach to overcome free-riding behavior in P2P Fog Computing environments.
2. A free riding behavior simulator, built on top of the renowned QTM [14] simulator, that enables modeling of the free riding behavior in P2P systems.
3. An extensive evaluation framework that assesses the performance of freeriding mitigation algorithms with respect to both fairness, success rate and execution time considering different system scales and loads.

The novelty of this work lies in the pricing model. The pricing model is dynamic, it changes with the supply and demand of a resource. Dynamically assigned value ascribes higher incentives to suppliers who provide demanding peers with the most requested least available resources. This allows the most requested resources to garner more suppliers, least requested resources will have fewer suppliers.

The remainder of this paper is organized as follows. Section 2 reviews related work in combating free riding behavior. Section 3 introduces the AFMIA design. Section 4 describes the evaluation methodology, and in Section 5 we present and discuss the experimental results. Finally, a summary with concluding remarks is provided in Section 6.

2. Literature Review

The related works that address free-riding behavior in P2P systems can be broadly classified into two main categories: (1) Reputation-based approaches, and (2) Economy-based approaches. Reputation-based approaches aggregate feedback from other peers in order to build trust among peers. A large number of contributions have adopted this approach, for instance [6,9,15–24]. They mainly differ in how trust is computed for a peer and the degree of accuracy and complexity involved [25]. Reputation-based approaches share multiple disadvantages such as centralization and communication overhead as well as whitewashing [26] and collusion [27,28]. Further, reputation systems are implemented upon the assumption that each peer would report their interactions with others honestly, which is not always the case [29].

Economy-based approaches can be further classified into three categories, namely, monetary-based approaches, reciprocity-based approaches, and credits (or points)-based approaches. AFMIA is considered a credit-based approach but has a dynamic file value based on current supply and demand.

Therefore, we briefly review the monetary-based and reciprocity-based approaches and focus on the related work of the credit-based approach. The first approach operates based on charging peers for resource consumption. A payment scheme was proposed in [30] which mainly consists of two components; an accounting module to store virtual currency for each peer and a settlement module, maintained by a single authority, to exchange virtual currency for services. This solution is impractical when applied to P2P systems since it requires an external centralized authority and suffers from large communication overhead and a single point of failure [26,31]. In addition, the majority of the monetary-based approaches require permanent user identification, which is difficult to implement due to the decentralized nature of P2P systems and peer anonymity [32].

In reciprocity-based approaches, peers receive services based on their level of contribution, also known as the barter system. The reciprocity can be either direct, which is based on the peer's own interactions, or indirect, where peers consider the collective interactions of all peers. A major disadvantage of the former is that it requires frequent interactions between each pair of peers, while the latter can be easily deceived by malicious peers [33]. This approach was implemented in BitTorrent using the tit-for-tat (TFT) strategy [26,27,33]. In [29], the authors proposed demerit point strategy that enhances optimistic unchoking algorithm in BitTorrent. In addition, some works, such as [34], used game theory to measure peers' contributions.

In credits-based approaches, peers receive credits by providing services to other peers, and spend credits by requesting services from others. The work in [35] used a points-based incentive mechanism, namely Global Contribution (GC), to maintain fairness in P2P systems. The proposed GC algorithm calculates a global score for each peer based on its bandwidth contribution to the entire network, which is then used to determine if the peer can download files. The GC increases when the peer uploads more than it downloads, uploads more to a high-contributing peer than to a low-contributing peer (free rider), or downloads more from a low-contributing peer than a busy high-contributing peer. In [36] an enhancement for this mechanism is proposed by assigning a Simplified Biased Contribution Index (SBCI) to each peer to balance the number of uploads and downloads. Index managers update the SBCI at a specified time interval for each peer based on their contribution during the previous period. A credit-based approach to overcome free riding in P2P networks is introduced in [37]. Peers are initially provided with an equal number of points which are increased by one when a peer uploads and decrease by one when a peer downloads. If the points balance for a peer reaches a certain amount, the peer is not allowed to download files and is given a grace period, during which the peer can pay off the balance. If the balance is not paid off, the peer is charged interest. The concept of grace period was also used in [38]. In their work, however, the grace period begins once a peer downloads a file. If the peer does not upload a file to the network during the period, the peers' future requests are rejected. An Extended Point-Based Incentive Mechanism (EPBIM) is presented in [39] for solving the free-riding problem. The system awards active peers who share their resources an incentive in the form of extra points which can be used to download files.

3. System Design

3.1. Free-Market Inspiration

In economics, a market is defined as a group of buyers and sellers of a specific product [12]. Buyers are individuals who buy the product; hence as a group they determine the demand for the product. Sellers are individuals who sell the product, so they determine the supply of the product. In short, the behavior of buyers and sellers is what is referred to in economics as demand and supply, respectively.

In a competitive market, there are a large number of sellers and buyers. Each seller knows that similar products are offered by other sellers, and each buyer is aware that there are many sellers to buy from. Therefore, a single seller or buyer has no influence over the quantity or price of a product. Rather, this is determined by the collective interaction between all sellers and buyers as they exchange products,

services, and money. A free market is a market where pricing for goods and services are determined merely by the forces of demand and supply, with no government intervention. The opposite of a competitive market is a monopoly market, where there is a single seller of a particular product who sets the price.

Demand captures buyer behavior and is defined as the rate at which buyers want to purchase a product [13]. A demand curve is a graphical representation of the relationship between price and quantity in demand in a given time period (see Figure 1a). Quantity of a particular product in demand is the amount of product that buyers are “willing to” and “can” purchase [12]. Willingness captures the first factor that affects demand which is taste. The second factor is the ability to purchase a product at a particular price, which means that one must have sufficient wealth. Both demand factors are affected by the market price. The law of demand states that as the price P of a particular good increases, the quantity in demand Q_d decreases, as shown in Equation (1) below and depicted in the demand curve in Figure 1. There are other factors that affect demand such as income, number of buyers, and prices of related products. However, economics usually assume that these factors are constant since they act over a much larger time frame [13].

$$\uparrow P \rightarrow \downarrow Q_d \tag{1}$$

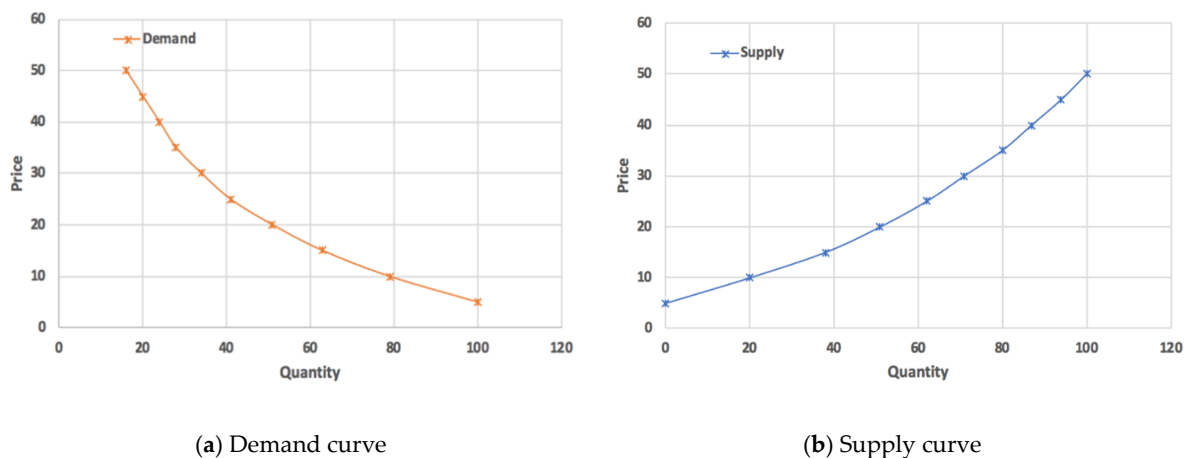


Figure 1. Demand and supply curves.

Supply captures seller behavior, which is determined by willingness and ability to supply goods. Figure 1b shows the supply curve, which represents the relationship between quantity supplied and price. The quantity supplied is defined as the amount of product that sellers are willing and able to sell [12]. The curve slopes upward, which indicates that at higher prices, a higher quantity of the product will be available to consumers since suppliers are able to maintain a profit regardless of the costs of production [13]. The law of supply states that as the price P of a particular good increases, the quantity supplied Q_s increases [12], this is shown in Equation (2) below.

$$\uparrow P \rightarrow \uparrow Q_s \tag{2}$$

Figure 2 depicts both supply and demand curves intersecting at the market’s equilibrium, where the price is called the equilibrium price or market clearing price. At the equilibrium price, sellers have sold all they want and buyers have purchased all they require [12]. If the market price increases above the equilibrium price, the quantity supplied is greater than the quantity demanded, which results in a surplus. In this case, sellers reduce prices to attract buyers, which results in an increase in quantity demanded and a decrease in quantity supplied. Sellers continue to cut prices until the equilibrium price is reached. If the market price decreases below the equilibrium price, the quantity

supplied is less than the quantity demanded, which causes a shortage. In this case, sellers can increase prices. When the market price increases, the quantity demanded decreases and the quantity supplied increases until equilibrium is reached.

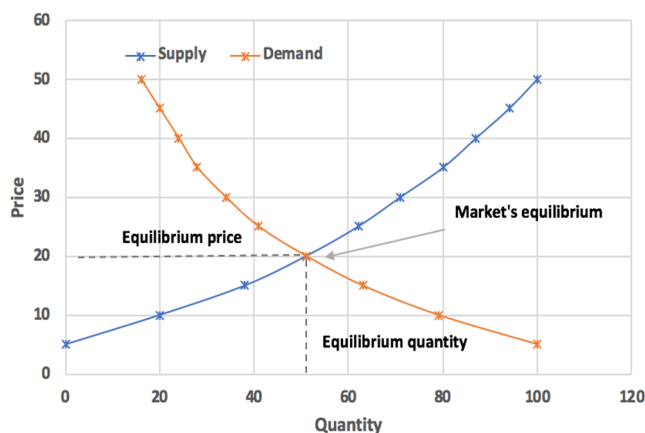


Figure 2. Equilibrium of supply and demand.

In our model, resources' values, much like in free markets, are a manifestation of peers' (fog nodes') behavior. Demand is represented by the total number of peers who request a resource, while supply is the number of peers that own a resource and are willing to share it with other peers in the network. As the demand for a resource increases, the value of the resource increases, and suppliers are incentivized to increase their supply of said resource. When the supply reaches a critical point that exceeds the demand, the value of the resource begins to decline proportionately. In addition, each peer has wealth based on credits that the peer owns and must have enough wealth equal or more to the desired resource value. Once a peer consumes a resource, the resource value is deducted from the peer's wealth. Also, the resource value is added to the provider peer's wealth.

3.2. System Architecture

As shown in Figure 3, AFMIA consists of two main components; the system component and the peer component. The system component consists of three main sub-components: the system registry, the resources tracker, and the peers tracker. The system registry keeps track of the available resources and peers in the system. It includes two public lists: the resources list and the peers list. The resources list contains information about all available resources in the system, as shown in Table 1. This acts as a resource management service that is dynamically updated after each transaction. The peers list contains information about all peers in the system. This includes the peer's ID, IP address, and wealth. The resources tracker keeps track of the value and growth rate of each resource available in the system and then updates them in the resources list. The peers tracker computes the wealth of each peer and incentives given to suppliers after each transaction. It also updates these values in the peer list.

The peer component consists of three main sub-components: the local resource list, the resource manager, and the credit manager. The local resource list is a private list for each peer that stores a list of all resources that the peer owns. The content of this list is updated whenever the peer available resources change. The resources manager includes the wealth tracker and the resource provider/requester. The former is responsible for checking whether a peer has enough wealth (credit) to request a resource. If so, the resource provider/requester allows the transaction to proceed. Once the transaction is complete, the resource provider/requester informs the system to update the information of the corresponding resource in the resources list and also peer's wealth in the peers list. In the case of not enough wealth, the wealth tracker asks the peer to earn credits. The credit manager is activated when a peer needs to earn credit. This includes a resource selector and a strategy selector. The resource selector is responsible for selecting a source (local/global) from which a resource will

be selected for sharing, while the strategy selector selects a resource based on the selected wealth accumulation strategy, either time-based or credit-based.

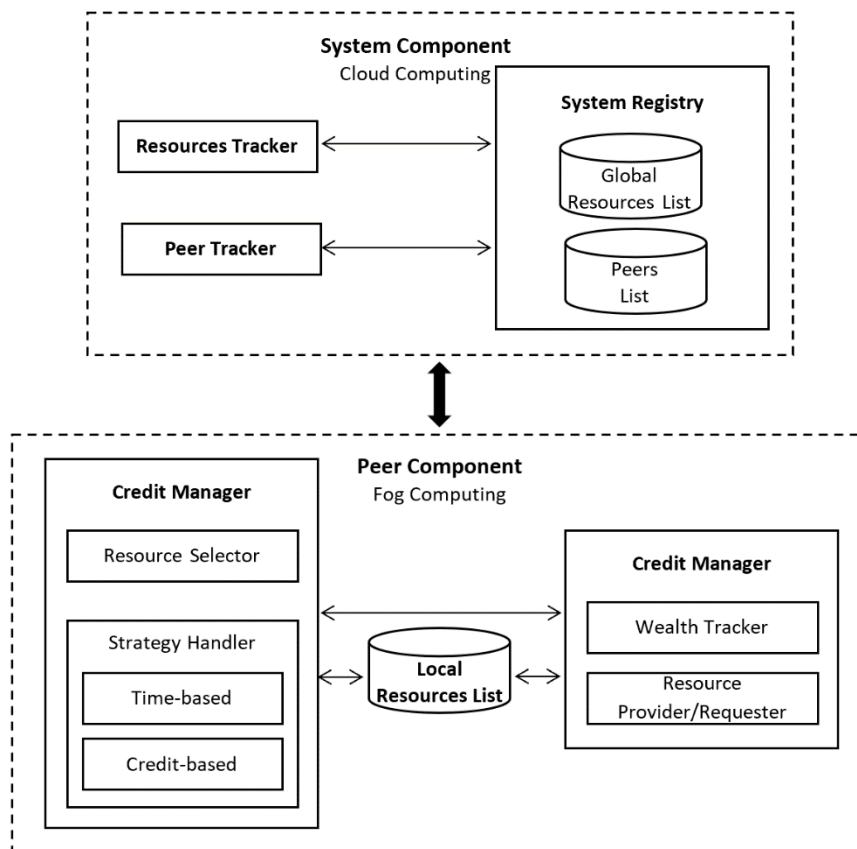


Figure 3. System architecture.

Table 1. Main fields in the resources list.

| Field | Description |
|----------------------|---|
| RID | A unique identifier for the resource |
| Type | Type of the resource |
| Resource description | A short description of the resource |
| Availability | Data related to when the resources is available |
| Last transaction | Information about the last transaction when this resources was involved |
| Number of suppliers | Current number of peers who have this resource |
| Number of demanders | Number of peers currently requesting this resource |
| Value | Value of the resource in credit |
| Resources size | Size of the resources |
| Growth rate | Growth rate of the resource |
| Supplier | List of suppliers of the resource, their IP addresses, and IDs |

3.3. Definitions

AFMIA uses the following terms. For simplicity, we present the notations used for these terms in Table 2.

Table 2. AFMIA notations.

| Notation | Description |
|------------|---|
| S_i | Number of suppliers of resource i |
| D_i | number of demanders of resource i |
| V_i | Value of resource i in credits |
| I_i | Incentives of resource i paid in credits to all suppliers |
| G_i | Growth rate of the value of resource i |
| Z_i | Size of the resource i in |
| $V_i(t_1)$ | Value of resource i at time t_1 |
| $V_i(t_2)$ | Value of resource i at time t_2 |
| Δt | Elapsed time between two time steps (from t_1 to t_2) |
| W_j | Wealth of the peer j in credit |
| FR-RoS | Free riders' rate of sharing |
| RoS | Rate of sharing |

- **Suppliers:** Peers that own a resource and are willing to share it with other peers in the network. We refer to the number of suppliers of a resource i by S_i . The minimum value of S_i for any resource i is one.
- **Demanders:** Peers who request resources from suppliers. We use D_i to refer to the number of demanders of a resource i . Supplier and demander are dynamic roles. A peer can be a supplier of a certain resource and a demander of another.
- **Resource Value:** The value of a resource i is measured in credits and denoted by V_i . This is a dynamic value updated after each transaction. We refer to the value of a resource i at time t_1 by $V_i(t_1)$.
- **Incentive:** The amount of credits that all suppliers of a resource except the actual provider receive once the demander uses the resource. We use I_i to refer to the incentives of a resource i .
- **Growth Rate:** The amount of change in a resource i 's value during a certain period of time, denoted by G_i . A resource has a high growth rate if the resource value increases over a short period of time, which indicates that there is currently a high demand for it.
- **Resource size:** Size of a resource i in the resource unit, referred by Z_i . It is normalized in the range (2–10).
- **Delta time (Δt):** Elapsed time between two time steps (from t_1 to t_2).
- **Wealth:** The amount of credits a peer owns. We refer to the wealth of a peer j using W_j . Peers can earn credit when they provide a resource.
- **Global resources:** Resources available to all peers in the system through the cloud. Information about these resources is stored in the resources list in the system component.
- **Local resources:** Resources owned by a peer.
- **Rate of sharing:** This represents how often a free rider responds to resource sharing requests. A free rider's rate of sharing of 0% (FR-RoS = 0%) indicates that the free rider does not respond at all, while 20% indicates that the free rider responds to only 20% of the requests.

3.4. AFMIA Algorithm Explained

Figure 4 presents the AFMIA algorithm flowchart. At system initialization, all resources values and growth rates are set to zero. As the number of demanders D_i of a resource i increases, the value of the resource V_i increases according to Equation (3). The growth rate G_i of the resource can be computed by Equation (4).

$$V_i = \frac{D_i}{S_i \times Z_i} \quad (3)$$

$$G_i = \frac{V(t_1) - V(t_2)}{\Delta t} \quad (4)$$

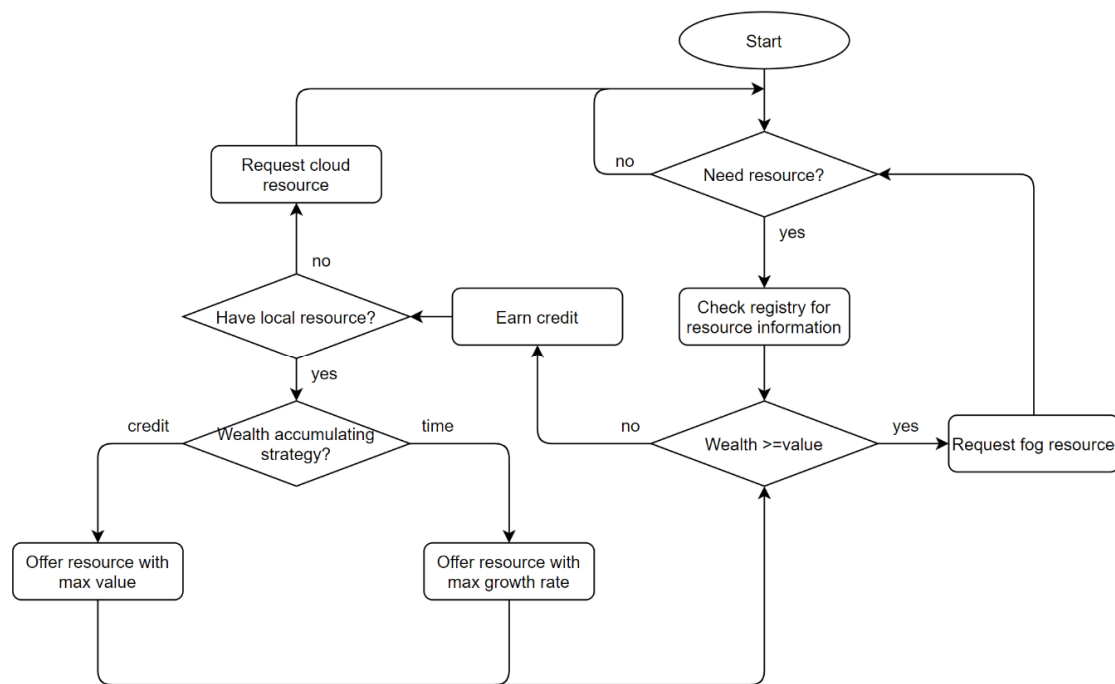


Figure 4. AFMIA algorithm.

This should incentivize other suppliers of the resource. Therefore, once a demander uses a resource, the system deposits credits to the supplier who provides the resource that are equal to the resource value. Additionally, the system incentivizes all other suppliers of the resource by depositing credit, calculated according to the Equation (5)

$$I_i = \frac{V_i}{S_i} \tag{5}$$

As a result, the number of demanders D_i decreases. At the point of equilibrium (where $D_i = S_i$), the value of the resource V_i and incentives I_i starts to decline as S_i exceeds D_i . The system continues to behave in this manner. Each peer starts out with a wealth equals to its available resources. Then, if a peer j needs a resource R_j , the peer checks the system registry for information about the resource and its suppliers. After selecting a resource, if the peer has enough credits to afford the resource value (i.e., $W_j \geq V_i$), the access request is approved. In the event that the value of the resource is greater than the peer’s wealth, but the peer still wants this resource, the wealth tracker instructs the resources selector to share some resources to earn credits. In this case, the resources selector checks if the peer has available local resources. If so, the peer chooses one according the wealth accumulation strategy (time or credit as explained below) and adds it to local resource list which means that it is allowed for sharing and added to the system registry. Otherwise, the peer selects a global resource (available through the cloud).

There are two strategies that can be used for selecting a resource; credit-based and time-based wealth accumulation strategies. The adoption of one strategy over the other depends on the preference or priority of the peers when they seek to grow their wealth by earning credit to be paid later as a price for using a resource. The credit-based strategy assumes that the peer’s goal is to increase his wealth regardless of the number of resources that he may need to share. Therefore, if the peer follows this strategy, the following step is performed, depending on whether the peer is offering a local or global resource:

- Local resource: The peer selects a resource with the maximum value V_i to share.

- Global resource: The peer selects a resource with the maximum value V_i to share, such that $V_i \leq W_j$.

The time strategy assumes that the peer's primary objective is to minimize the number of resources that need to be shared in order to be allowed access to his needed resource in the shortest time possible. This is achieved by selecting a resource with an increasing value. Therefore, if the peer follows this strategy, the following step is performed, depending on whether the peer is sharing a local or global resource:

- Local resource: The peer selects a resource with the maximum growth rate G_i to share.
- Global resource: The peer selects a resource with the maximum growth rate G_i to share, such that $V_i \leq W_j$.

The peer continues to offer resources for sharing until the peer's wealth W_j is greater than or equal to V_i , where i is the resource that the peer wants to use. Once the peer has enough wealth, the access request is approved and the peer's wealth is updated by deducting credits equal to the resource value, while the suppliers' wealth is increased. In addition, the system updates several data points in the resources list such as the resource value, last transaction, number of suppliers, number of demanders, and the resource's growth rate.

A Pareto distribution usually results from a perfectly free market, that is a minority of peers will be the only ones able to acquire high demand low supply resources by having the majority of network credits. Since the goal of AFMIA is to ensure resources are readily available when they are needed, it is necessary to ensure that credits do not concentrate in a minority of peers. Therefore, all suppliers of a particular resource are rewarded with credit in order to avoid a Pareto distribution of wealth from emerging.

4. Evaluation Methodology

For conducting experiments, we built a free-riding behavior simulator on top of the Quantitative Trust Management (QTM) [14] renowned simulator. We modified the simulator to introduce a free rider user model, which responds to some (or no) files requests according to the free rider's rate of sharing. QTM is a Java-based open source simulation framework for file sharing in P2P networks to evaluate trust management systems.

For simplicity and without compromising generality we considered a file sharing (as a special case of resource sharing) network. We evaluated two versions of the AFMIA algorithm based on the adopted wealth accumulation strategy; time-based strategy and credit-based strategy. We also present the performance of the system when no anti-free riding algorithm (none algorithm) is used, in order to establish a baseline performance to compare with AFMIA. We conducted simulations of a network of 100 peers, 300 files, and a total of 3000 transactions. These are the maximum values possible before the machine crashes or suffers from a considerable performance degradation. The proposed approach was evaluated with different percentages of free riders in the network. Three cases are considered (1) the majority of peers are free riders, (2) the majority of peers are non-free riders, and (3) the number of free riders and non-free riders are equal. We also considered three types of free riders according to the rate of sharing, which represents how often a peer responds to file sharing requests; (1) complete free riders who contribute nothing (0% rate of sharing), (2) partial free riders (20% rate of sharing), and (3) partial free riders (50% rate of sharing). A combination of the aforementioned cases resulted in nine scenarios, which are summarized in Table 3.

Table 3. Experimental setting: In all scenarios, the system was operated with 100 peers, 300 files, and 3000 transactions. (FR = free rider, NFR = non-free rider, FR-RoS = free riders' rate of sharing).

| Peers | FR | NFR | FR-RoS |
|----------|----|-----|--------|
| FR < NFR | 20 | 80 | 0% |
| | | | 20% |
| | | | 50% |
| FR = NFR | 50 | 50 | 0% |
| | | | 20% |
| | | | 50% |
| FR > NFR | 80 | 20 | 0% |
| | | | 20% |
| | | | 50% |

We evaluated the AFMIA algorithm with respect to three performance metrics. These metrics have been selected based on their previous usage by the research community of free riding in P2P systems [9,15].

- **Fairness:** This metric suggests that peers should not receive/download more files than they send/upload to the network. It is computed as the total number of files sent over the total number of files received. Unfairness can occur when a free rider is receiving more files than sending (fairness less than one), or when a non-free rider is sending more files than receiving (fairness over one). In this work, we measure fairness for free riders (FR) and non-free riders (NFR) according to Equation (6) and Equation (7), respectively.

$$\text{Fairness}_{\text{FR}} = \frac{\text{Number of files received by FR}}{\text{Number of files sent by FR}} \quad (6)$$

$$\text{Fairness}_{\text{NFR}} = \frac{\text{Number of files received by NFR}}{\text{Number of files sent by NFR}} \quad (7)$$

- **Success Rate:** The success rate is calculated as the total number of files received over the total number of files requested. Similar to fairness, we measure the success rate for free riders and non-free riders according to Equation (8) and Equation (9), respectively.

$$\text{Success Rate}_{\text{FR}} = \frac{\text{Number of files received by FR}}{\text{Number of files requested by FR}} \quad (8)$$

$$\text{Success Rate}_{\text{NFR}} = \frac{\text{Number of files received by NFR}}{\text{Number of files requested by NFR}} \quad (9)$$

- **Execution time:** This metric measures the simulation running time that the system took to perform a specific number of transactions using the AFMIA time-based strategy, the AFMIA credit-based strategy, and none algorithm.

5. Results and Discussion

In this section, the experimental results are presented for fairness, success rate, and execution time to evaluate the performance of the proposed approach versus the none algorithm. All experiments have been carried out on a personal computer equipped with Windows 10, i7 processor, and 8 GB of RAM. Each scenario was repeated ten times and the mean outcomes were calculated to improve accuracy and reduce possible variability resulting from randomizations.

5.1. Fairness

The performance of the proposed algorithm with respect to fairness is shown in Figure 5. Each sub-figure presents the fairness values for both free riders and non-free riders over time measured at an increasing number of transactions performed. Each sub-figure also shows the performance of peers with various free-riders' rates of sharing (FR-RoSs). As expected, the figures suggest that, in all scenarios, the system is unfair at system initialization as free riders have a fairness value of less than one and non-free riders have a fairness value that is greater than one. The former observation goes in line with the known definition of free riding behavior, and as a result, non-free riders take the full workload of sending files requested by other peers. Yet with the none algorithm (Figure 5g–i), as no anti-free riding algorithm is used, free riders continue to consume the system resources over time with little or no contribution (depending on FR-RoS). Therefore, the system continues to be unfair since non-free riders have an almost constant greater-than-one fairness value and free riders have less-than-one fairness value over time.

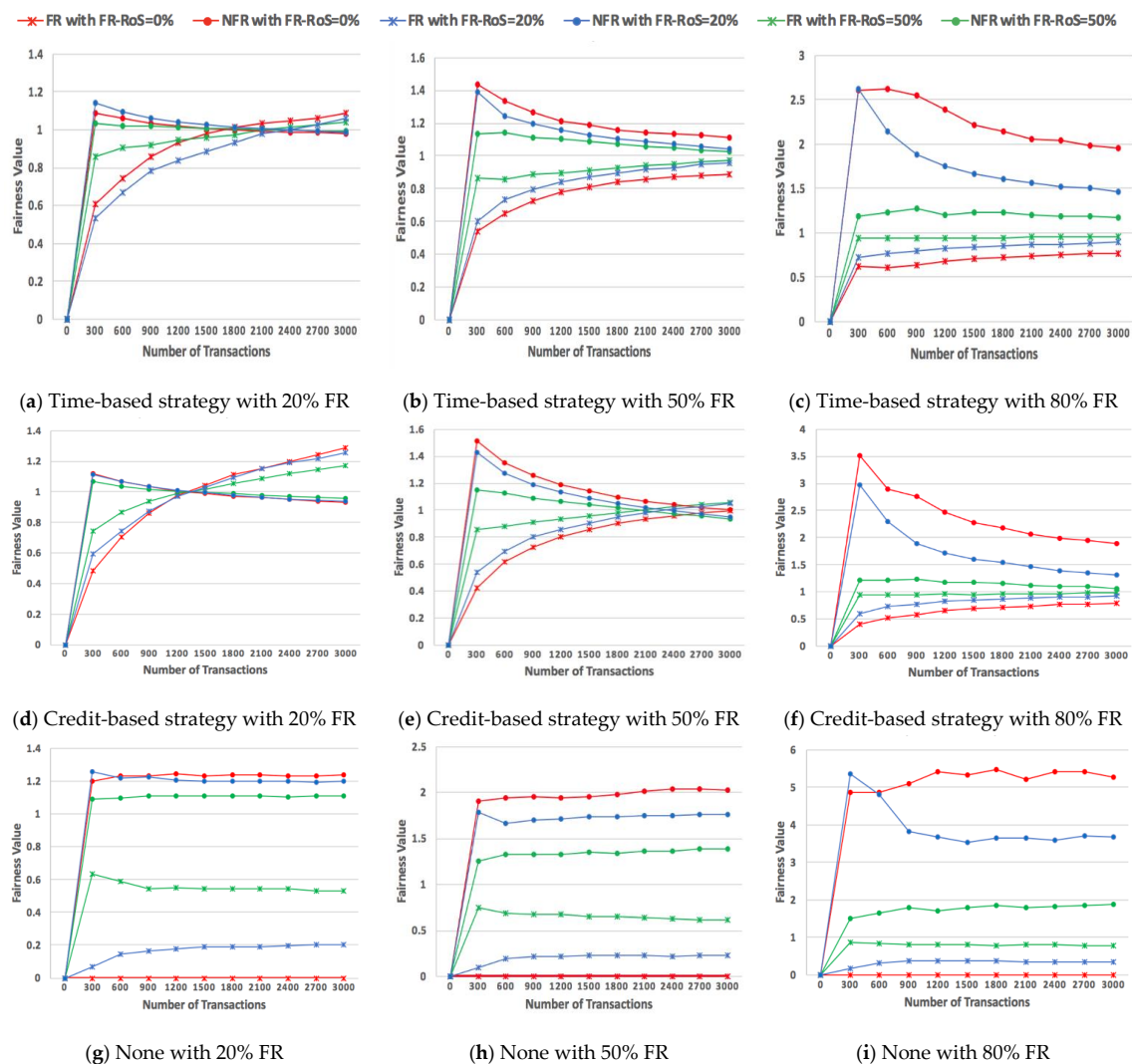


Figure 5. Fairness performance of each strategy.

When AFMIA is adopted, however, using the time-based (Figure 5a–c) and credit-based (Figure 5d–f) strategies, it successfully increases fairness for free riders over time to reach one since free riders are obliged to send files. Consequently, the fairness value of non-free riders decreases over time to reach one too since the number of files that they need to send decreases, which renders a

fair system for both free riders and non-free riders. In scenarios with a small percentage of free riders, adopting a credit-based strategy can help achieve fairness faster for both free riders and non-free riders, but may cause free riders' fairness to slightly exceed one (see Figure 5d), compared to their counterparts for the time-based strategy in Figure 5a. When a credit-based strategy is followed, free riders aim to maximize their wealth regardless of how many files that they need to upload. Therefore, free riders select files with maximum value, and continue to upload files until they earn enough credit. This can cause a higher increase in the number of files uploaded by free riders, compared to those downloaded, and hence fairness is slightly over one.

The results in Figure 5 are quite revealing in several ways. It can be observed in the credit-based strategy in Figure 5d that once fairness is achieved (fairness = 1), free riders with a low rate of sharing (FR-RoS = 0% and FR-RoS = 20%) have a higher fairness value than those with FR-RoS = 50%. The reason for this is that free riders with a low rate of sharing are used to share nothing or few files (depending on FR-RoS), hence they have no or low wealth, respectively. Free riders with a low rate of sharing need to upload more files than free riders with high rate of sharing. In contrast, peers using a time-based strategy focus on minimizing the number of files that need to be uploaded in order to receive the file they require in the shortest time possible. Peers select files with a maximum growth rate since their values are likely to continue growing. In this case, free riders upload fewer files and receive their files faster, but for the same reason, they require more time to achieve fairness compared to the credit-based strategy. Therefore, we can conclude that the AFMIA algorithm, by adopting the free market economy, can successfully adjust the behavior of free riders to improve fairness for both free riding and non-free riding peers.

Several observations can also be made relating to the effect of the percentage of free riders present in the system and their FR-RoS on the fairness of free riders and non-free riders. The value FR-RoS has a direct relationship with free riders' fairness, and an inverse relationship with non-free riders' fairness. As the value of FR-RoS increases, fairness of free riders increases since they send more files, and fairness for non-free riders decreases since they send less files. This is evident by the none algorithm shown in Figure 5g–i. Similar observations can be made from Figure 5a–f, where the proposed algorithm is used. In these scenarios, the effect of FR-RoS on fairness reduces over time as fairness reaches one since the behavior of free riders is adjusted. Further, systems with a high percentage of free riders take a longer time to reach fairness (Figure 5c,f), compared with systems with a smaller percentages of free riders as shown in Figure 5a,b,d,e.

5.2. Success Rate

We present the performance of the AFMIA algorithm with respect to success rate in Figure 6. Each sub-figure presents the success rates for both free riders and non-free riders over time, measured when a specific amount of transactions are performed. Each sub-figure also shows the performance of peers with various FR-RoSs.

Success rate is measured by dividing the total number of files received over the total number of files requested. In other words, it measures the percentage of successfully fulfilled files requests. A low success rate suggests that the majority of the files requested are not being received. Although the AFMIA algorithm is not designed to increase success rate, we compare the algorithm performance with the baseline scenario, i.e., none algorithm, to show the effect of adopting AFMIA on success rate since many related efforts have adopted it as a performance measure.

We can see in Figure 6 that all algorithms, including time-based AFMIA, credit-based AFMIA, and none, maintain nearly the same success rate for a given percentage of free riders. In scenarios with the none algorithm (Figure 6g–i), a peer's behavior does not affect whether the peer receives requested files since no anti-free riding algorithm is used. When the AFMIA algorithm is adopted using a time-based strategy (Figure 6a–c) and a credit-based strategy (Figure 6d–f), although peers do not receive requested files unless they own enough wealth (earned by sending files), both free riders and non-free riders effectively maintain similar success rates with the corresponding none algorithm

scenarios. This reveals that AFMIA can successfully improve fairness for both free riders and non-free riders over time, as shown in Section 5.1, while simultaneously not compromising on success rate by maintaining a similar rate as the baseline scenario, i.e., the none algorithm.

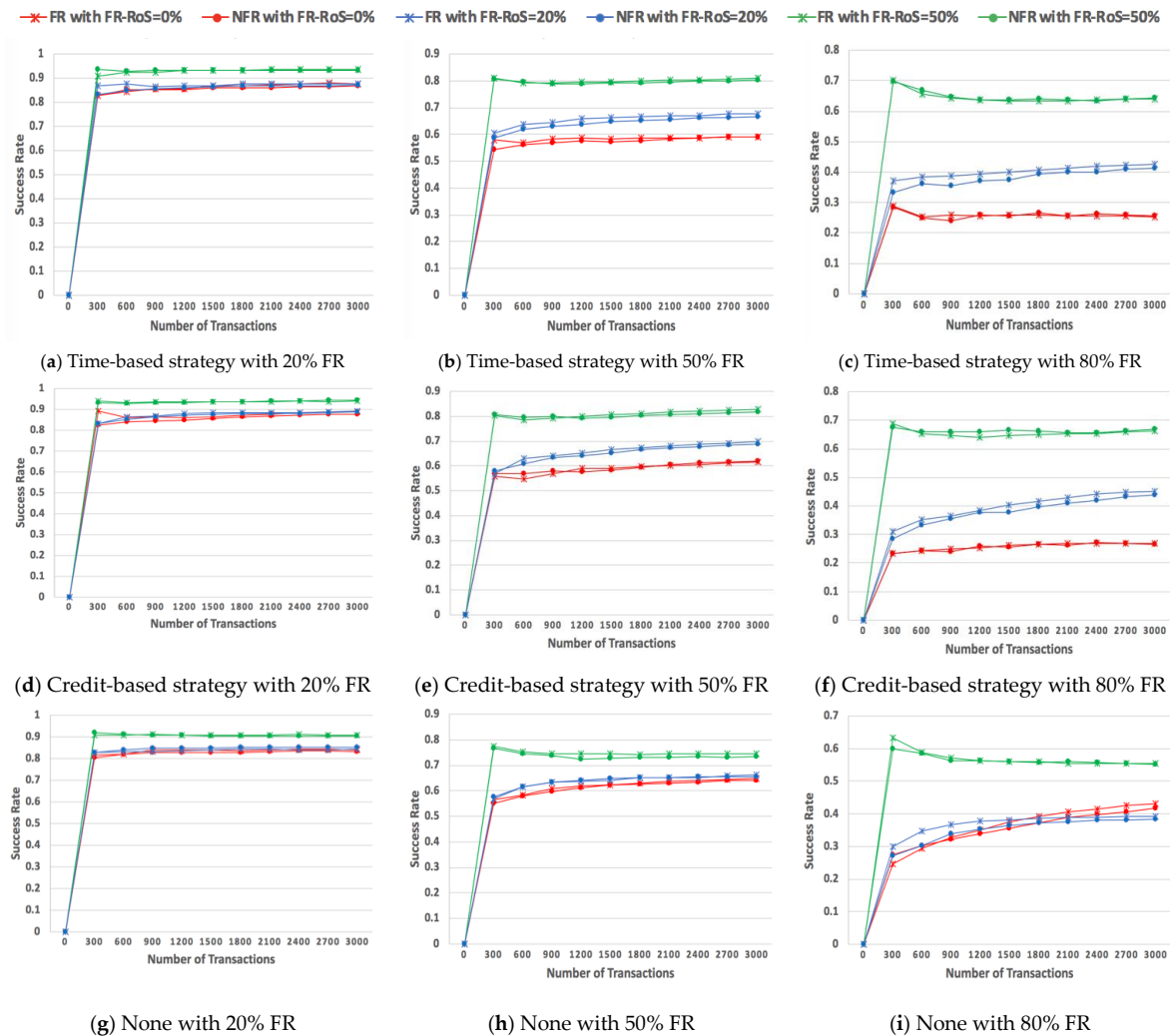


Figure 6. Success rate of each strategy.

Further to the above observation, several remarks can be made from Figure 6 concerning the impact of FR-RoS and percentage of free riders on success rate. We can see that all in scenarios where the FR-RoS is higher, the higher are the success rates for both free and non-free riders. This influence of the FR-RoS on success rate is greater in scenarios with a high percentage of free riders (Figure 6c,f,i), as demonstrated by the larger discrepancy between success rates. Moreover, for a given FR-RoS, it can be seen that the higher percentage of free riders present in the network, the lower the success rate is across strategies. This is because in such situations the small percentage of non-free riders in the system, take on the full workload of sending files requested by other peers, which increases the likelihood of not responding.

5.3. Execution Time

In this section, we measure the simulation running time that the system spends in performing a specific number of transactions using the AFMIA time-based strategy, credit-based strategy, and none algorithm. The results are presented in Figure 7. Each sub-figure shows the execution time required to

perform an increasing amount of transactions with various FR-RoSs. Each sub-figure corresponds to one strategy and a specific percentage of free riders present in the network.

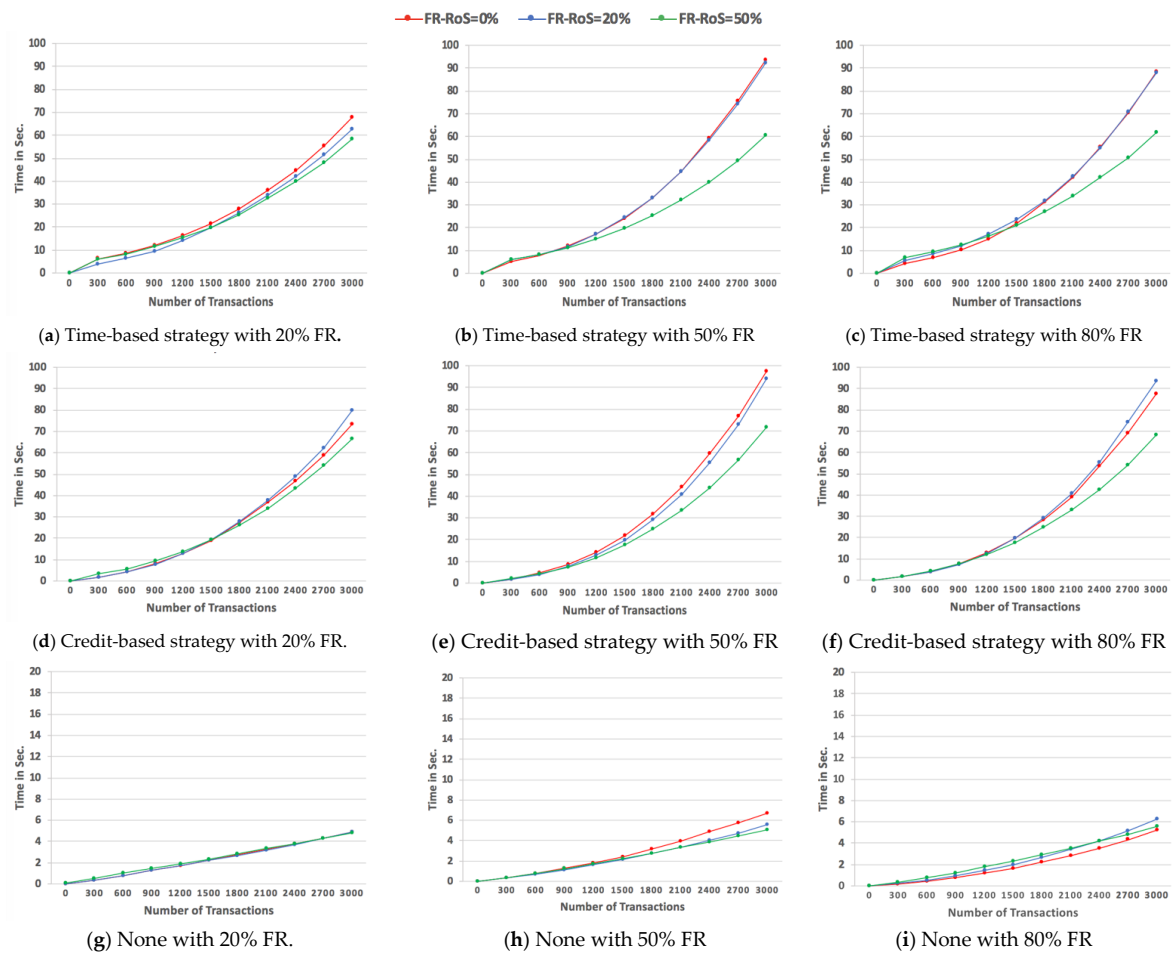


Figure 7. Execution Time of each strategy.

Intuitively, when no strategy to stop free-riders behavior is used, the system’s execution time per n transactions is lower than a system that does have a strategy to remedy the problem. The results demonstrate that adopting the AFMIA algorithm can increase execution time for the same number of transactions compared to none algorithm. This increase is justified by the de-incentivization of free riding, and the decentralization of the decision making process, with the added benefit of incentivizing the most efficient resource allocation, by setting the highest value to the most demanded, least supplied files.

If a peer does not have enough wealth (i.e., previous contribution to the network), the peer must collect more credit first before requesting a file. To do so, a peer first decides whether to upload a local or global file, and then searches the files list sequentially to select a file based on the adopted wealth accumulation strategy (time or credit). The peer continues to upload files until enough credit is earned, and then it can request the desired file. However, it can be seen that, for a given FR-RoS and percentage of free riders, the credit-based strategy (Figure 7d–f) has a higher execution time than the time-based strategy (Figure 7a–c). As explained earlier, when the credit-based strategy is followed, peers aim to maximize their wealth regardless of how many files they need to upload.

Therefore, peers select files with maximum value, and continue to upload files until they earn enough credit. While with the time-based strategy, the goal of peers is not only maximize wealth but also minimize the number of files that need to be uploaded. Therefore, peers select files with the maximum growth rate since their values are likely to continue growing. In this case, peers upload

fewer files and receive the desired files faster. The figures also suggest that FR-RoS has no effect on the execution time in the case of the none algorithm (Figure 7g–i). However, this observation does not extend to the AFMIA time-based (Figure 7a–c) and credit-based (Figure 7d–f) strategies. We can see from these figures that when free riders have low FR-RoS (FR-RoS = 0% and FR-RoS = 20%), execution time is higher compared to scenarios with high FR-RoS (FR-RoS = 50%). The reason for this is that free riders with a low rate of sharing share nothing or few files (depending on FR-RoS), hence they have no or low wealth. Therefore, the presence of such free riders in the system causes an increase in execution time since more decisions have to be taken by the algorithm. For the same reason, a higher percentage of free riders with a low rate of sharing in the system (50% and 80%) increases execution time in both time-based (Figure 7b,c) and credit-based (Figure 7e,f) strategies compared to corresponding scenarios with a low percentage of free riders. The figures also indicate that increasing the percentage of free riders while maintaining a high FR-RoS has a marginal effect on the execution time for both strategies.

6. Conclusions

P2P architecture is a promising paradigm for fog computing as it help overcoming scalability and unavailability in IoT system. This paper presents the AFMIA algorithm, an approach to mitigate free riding behavior in P2P system based on the free market economy. In the proposed algorithm, resources have values that, much like the free market, are set freely by the amount of supply and demand. In addition, each peer has wealth measured by the amount of credit that the peer owns and can be used to pay for requested resources. We state that a peer can afford a resource if the peer's wealth is greater than or equal to the resource's value. We proposed two wealth accumulation strategies that peers can follow when collecting credits to increase wealth; namely the credit-based strategy and the time-based strategy. The former strategy assumes that the peer's goal is to increase wealth regardless of the number of resources that need to be shared. The time-based strategy assumes that the peer's primary objective is to minimize the number of resources that need to be shared in order to receive the desired resource in the shortest possible time.

The experimental results suggest that the proposed algorithm can successfully improve fairness for both free and non-free riders over time, while not compromising on success rate by maintaining similar performance with the baseline case, i.e., the none algorithm. When the credit-based wealth accumulation strategy is adopted, the system reaches fairness faster than the time-based strategy because free riders tend to offer more resources while collecting credits. However, this may slow down the system, and hence the credit-based strategy is outperformed by the time-based strategy with respect to execution time. In addition, it has been observed that the greater the presence of free-riding behavior in the system, represented by the low free-riders rate of sharing and/or the high percentage of free riders, the lower the fairness and success rate values are, and the longer the execution time is in both the time-based and credit-based strategies.

There are several lines of research arising from this work which can be pursued. First, considering network features that are specifically related to the fog context, such as node power and availability, is helpful. It is our intention to investigate this thoroughly in a future work. In addition, due to the computational resource limitations, we have only considered a medium size system of constant number of services and transactions. Studying the system behavior under a different scale will allow us to gain more insight into the algorithm behavior.

Author Contributions: Conceptualization, M.A.; Formal analysis, A.A.; Funding acquisition, H.K.; Methodology, H.K.; Software, S.A.; Supervision, H.K.; Validation, S.A.; Visualization, S.A.; Writing—original draft, H.K.; Writing—review & editing, A.A. and M.A. All authors have read and agree to the published version of the manuscript.

Funding: This research was supported by a grant from Researchers Supporting Unit, Project number (RSP-2020/204), King Saud University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Tracey, D.; Sreenan, C. How to see through the fog? Using Peer to Peer (P2P) for the Internet of Things. In Proceedings of the 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019; pp. 47–52.
2. Aslani, R.; Hakami, V.; Dehghan, M. A token-based incentive mechanism for video streaming applications in peer-to-peer networks. *Multimed. Tools Appl.* **2017**, *77*, 14625–14653. [[CrossRef](#)]
3. Mahini, H.; Dehghan, M.; Navidi, H.; Rahmani, A.M. GaMe-PLive: A new game theoretic mechanism for P2P live video streaming. *Int. J. Commun. Syst.* **2015**, *29*, 1187–1203. [[CrossRef](#)]
4. Gupta, R.; Singh, Y.N. Reputation Aggregation in Peer-to-Peer Network Using Differential Gossip Algorithm. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 2812–2823. [[CrossRef](#)]
5. Abbes, H.; Louati, T. PGTrust: A decentralized free-riding prevention model for DG systems. *Clust. Comput.* **2016**, *19*, 585–599. [[CrossRef](#)]
6. Gupta, R.; Singha, N.; Singh, Y.N. Reputation based probabilistic resource allocation for avoiding free riding and formation of common interest groups in unstructured P2P networks. *Peer-to-Peer Netw. Appl.* **2016**, *9*, 1101–1113. [[CrossRef](#)]
7. Kang, X.; Wu, Y. Incentive Mechanism Design for Heterogeneous Peer-to-Peer Networks: A Stackelberg Game Approach. *IEEE Trans. Mob. Comput.* **2014**, *14*, 1018–1030. [[CrossRef](#)]
8. Cham, H. Study of Intelligence Peer Selection in Enhancing P2P Using Artificial Neural Network. Ph.D. Thesis, Universiti Tunku Abdul Rahman, Selangor, Malaysia, 2014.
9. Ghaderzadeh, A.; Kargahi, M.; Reshadi, M. InFreD: Intelligent Free Rider Detection in collaborative distributed systems. *J. Netw. Comput. Appl.* **2017**, *78*, 134–145. [[CrossRef](#)]
10. Biao, B.O.S.; Oluwatope, A.O.; Odukoya, H.O.; Babalola, A.; Ojo, O.E.; Sossou, E.H. Ayo game approach to mitigate free riding in peer-to-peer networks. *J. King Saud Univ. Comput. Inf. Sci.* **2020**. [[CrossRef](#)]
11. Li, S.; Sun, W.; Liu, J. A mechanism of bandwidth allocation for peer-to-peer file-sharing networks via particle swarm optimization. *J. Intell. Fuzzy Syst.* **2018**, *35*, 2269–2280. [[CrossRef](#)]
12. Mankiw, N.G. *Principles of Microeconomics*; Cengage Learning: Boston, MA, USA, 2020.
13. Whelan, J.; Msefer, J. *Economic Supply & Demand*; Sloan School of Management, MIT: Cambridge, MA, USA, 1996.
14. West, A.G.; Aviv, A.J.; Chang, J.; Prabhu, V.S.; Blaze, M.; Kan-nan, S.; Lee, I.; Smith, J.M.; Sokolsky, O. QuanTM: A quantitative trust management system. In Proceedings of the Second European Workshop on System Security, Nuremberg, Germany, 31 March 2009; pp. 28–35.
15. Meng, X.; Jin, J. A free rider aware topological construction strategy for search in unstructured P2P networks. *Peer-to-Peer Netw. Appl.* **2014**, *9*, 127–141. [[CrossRef](#)]
16. Momeny, S.; Dehghan, M. A new incentive scheduling mechanism for live video streaming in mesh-based P2P networks. In Proceedings of the 2016 8th International Symposium on Telecommunications (IST), Tehran, Iran, 27–28 September 2016; pp. 795–800. [[CrossRef](#)]
17. Kurdi, H.; Alsalamah, S.; Alatawi, A.; Alfaraj, S.; Altoaimy, L.; Ahmed, S.H. HealthyBroker: A Trustworthy Blockchain-Based Multi-Cloud Broker for Patient-Centered eHealth Services. *Electronics* **2019**, *8*, 602. [[CrossRef](#)]
18. Kurdi, H.; Alfaries, A.; Al-Anazi, A.; Alkharji, S.; Addegaither, M.; Altoaimy, L.; Ahmed, S.H. A lightweight trust management algorithm based on subjective logic for interconnected cloud computing environments. *J. Supercomput.* **2018**, *75*, 3534–3554. [[CrossRef](#)]
19. Al-Otaiby, N.; Kurdi, H.; Al-Megren, S. A Hierarchical Trust Model for Peer-to-Peer Networks. *Comput. Mater. Contin.* **2019**, *59*, 397–404. [[CrossRef](#)]
20. Kurdi, H.; Alnasser, S.; Alhelal, M. AuthenticPeer: A Reputation Management System for Peer-to-Peer Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 637831. [[CrossRef](#)]
21. Kurdi, H.A. HonestPeer: An enhanced EigenTrust algorithm for reputation management in P2P systems. *J. King Saud Univ. Comput. Inf. Sci.* **2015**, *27*, 315–322. [[CrossRef](#)]
22. Alhussain, A.; Kurdi, H. EERP: An enhanced EigenTrust algorithm for reputation management in peer-to-peer networks. *Procedia Comput. Sci.* **2018**, *141*, 490–495. [[CrossRef](#)]

23. Alkharji, S.; Kurdi, H.; Altamimi, R.; Aloboud, E. AuthenticPeer++: A trust management system for P2P networks. In Proceedings of the 2017 European Modelling Symposium (EMS), Manchester, UK, 20–21 November 2017; pp. 191–196. [\[CrossRef\]](#)
24. Alhussain, A.; Kurdi, H.; Altoaimy, L. A Neural Network-Based Trust Management System for Edge Devices in Peer-to-Peer Networks. *Comput. Mater. Contin.* **2019**, *59*, 805–816. [\[CrossRef\]](#)
25. Zhang, Q.; Yu, T.; Irwin, K. A classification scheme for trust functions in reputation-based trust management. In Proceedings of the International Semantic Web Conference (ISWC) Workshop on Trust, Security, and Reputation on the Semantic Web, Hiroshima, Japan, 7–11 November 2004.
26. Feldman, M.; Chuang, J. Overcoming free-riding behavior in peer-to-peer systems. *ACM SIGecom Exch.* **2005**, *5*, 41–50. [\[CrossRef\]](#)
27. Joung, Y.-J.; Chiu, T.H.-Y.; Chen, S.-M. Cooperating with free riders in unstructured P2P networks. *Comput. Netw.* **2012**, *56*, 198–212. [\[CrossRef\]](#)
28. Tang, Y.; Wang, H.; Dou, W. Trust based incentive in P2P network. In Proceedings of the IEEE International Conference on E-Commerce Technology for Dynamic E-Business, Beijing, China, 13–15 September 2004; pp. 302–305.
29. Manoharan, S.; Ge, T. A demerit point strategy to reduce free-riding in BitTorrent. *Comput. Commun.* **2013**, *36*, 875–880. [\[CrossRef\]](#)
30. Golle, P.; Leyton-Brown, K.; Mironov, I.; Lillibridge, M. Incentives for sharing in peer-to-peer networks. In *International Workshop on Electronic Commerce*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 75–87.
31. Zhang, Q.; Wang, S.; Liao, D. A Game Theoretic Analysis of Public Goods Allocation in P2P Networks. *KSII Trans. Internet Inf. Syst.* **2015**, *9*, 2854–2874. [\[CrossRef\]](#)
32. Zhang, Y.; Van Der Schaar, M. Peer-to-Peer multimedia sharing based on social norms. *Signal Process. Image Commun.* **2012**, *27*, 383–400. [\[CrossRef\]](#)
33. Sasabe, M. Analysis of optimal piece flow in tit-for-tat-based P2P streaming. *Comput. Netw.* **2018**, *139*, 60–69. [\[CrossRef\]](#)
34. Belmonte, M.-V.; Díaz, M.; Pérez-De-La-Cruz, J.-L.; Reyna, A. COINS: COalitions and INcentiveS for effective Peer-to-Peer downloads. *J. Netw. Comput. Appl.* **2013**, *36*, 484–497. [\[CrossRef\]](#)
35. Nishida, H.; Nguyen, T. A Global Contribution Approach to Maintain Fairness in P2P Networks. *IEEE Trans. Parallel Distrib. Syst.* **2009**, *21*, 812–826. [\[CrossRef\]](#)
36. Awasthi, S.K.; Singh, Y.N. Simplified Biased Contribution Index (SBCI): A mechanism to make P2P network fair and efficient for resource sharing. *J. Parallel Distrib. Comput.* **2019**, *124*, 106–118. [\[CrossRef\]](#)
37. Alotaibi, B.; Alarifi, N.; Abdulghani, M.; Altoaimy, L. Overcoming Free-Riding Behavior in Peer-to-Peer Networks Using Points System Approach. *Procedia Comput. Sci.* **2019**, *151*, 1060–1065. [\[CrossRef\]](#)
38. Hazazi, M.; Almousa, A.; Kurdi, H.; Al-Megren, S.; Alsalamah, S. A credit-based approach for overcoming free-riding behavior in peer-to-peer networks. *J. Comput. Mater. Cont.* **2019**, *59*, 15–29.
39. Srivastava, S.; Gupta, V.; Yadav, R.; Kant, K. Controlling free riding using extended point based incentive mechanism in peer- to-peer networks. In Proceedings of the 2012 Third International Conference on Computer and Communication Technology, Allahabad, India, 23–25 November 2012; pp. 200–205.

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).