Santos, A., Fernández Anta, A., Cuesta, J.A., López Fernández, L. (2016). Fair linking mechanisms for resource allocation with correlated player types. *Computing*, 98(8), pp.: 777–801.

DOI: https://doi.org/10.1007/s00607-015-0461-x

Noname manuscript No.
(will be inserted by the editor)

Fair Linking Mechanisms for Resource Allocation with Correlated Player Types

Agustín Santos · Antonio Fernández Anta · José A. Cuesta · Luis López Fernández

Received: date / Accepted: date

Abstract Resource allocation is one of the most relevant problems in the area of Mechanism Design for computing systems. Devising algorithms capable of providing efficient and fair allocation is the objective of many previous research efforts. Usually, the mechanisms they propose deal with selfishness by introducing utility transfers or payments. Since using payments is undesirable in some contexts, a family of mechanisms without payments is proposed in this paper. These mechanisms extend the Linking Mechanism of Jackson and Sonnenschein introducing a generic concept of *fairness* with *correlated preferences.* We prove that these mechanisms have good incentive, fairness, and efficiency properties. To conclude, we provide an algorithm, based on the mechanisms, that could be used in practical computing environments.

Keywords Resource Allocation, Task Allocation, Linking Mechanism, Fairness, Player Correlation

Agustín Santos
(Corresponding author) Institute IMDEA Networks, Madrid, Spain. E-mail: agustin.santos@imdea.org, Ph: +34 91 481 6979, Fax: Fax: +34 91 481 6965

Antonio Fernández Anta
Institute IMDEA Networks, Madrid, Spain. E-mail: antonio.fernandez@imdea.org

José A. Cuesta
GISC, Mathematics Dept., Universidad Carlos III, Madrid, Spain. E-mail: cuesta@math.uc3m.es

Luis López Fernández
FUN-LAB, Universidad Rey Juan Carlos, Madrid, Spain. E-mail: llopez@gsyc.es

1 Introduction

1.1 Motivation

The success of the Internet has made the problem of resource allocation to emerge in many versions like, for example, deciding which peer must receive bandwidth or disk in a file sharing P2P system [19], determining what access point shall offer its capabilities to a given user in a shared coverage area [9, 37], or deciding to which computational task some CPU is assigned in a collaborative distributed environment [1]. The problem may also appear with a negative formulation (i.e., instead of deciding who shall receive a resource, the problem is deciding who shall not receive it). For example, a paradigmatic example of this negative resource allocation that has been widely analyzed in distributed computing is the problem of computational *task allocation* [13,34, 6]. In this problem, a system composed of heterogeneous nodes[1] is assumed. Those nodes cooperate pursuing some common goal, which requires the execution of a collection of computational tasks. Whenever a new task appears, it is assigned for execution to the most suitable node. Given the popularity and simplicity of this problem, we will use task allocation as a reference.

In all these scenarios, it is very important to devise mechanisms that achieve efficient and fair resource allocation even when players present selfish or non-rational behavior. With that purpose, a number of interesting protocols and mechanisms based on Game Theory concepts [3,37] have been proposed. In such works, it is often assumed that players can transfer their utilities (i.e., use payments). However, there are many systems in which this assumption is not realistic. In particular, in many distributed Internet systems (e.g., all kind of free services), payments are not used or make little sense. For instance, there are no payments in most peer to peer systems, like BitTorrent (and this leads to free riding [19]).

Recently, some mechanisms without payments have been proposed, like those of Procaccia and Tennenholtz [28], or the seminal work of Jackson and Sonnenschein [17,18] in which a new type of mechanism (called *linking mechanism*) is proposed. A linking mechanism, instead of offering incentives or payments to players, limits the spectrum of players' responses to a probability distribution known by the game designer. One of the objectives of this paper is to explore and extend linking mechanisms, introducing a wide spectrum of fairness concepts, while preserving all the original properties. For this, we assume that the game is based on a sequence of single-unit (resource) allocations and that each player has a specific preference for being assigned the resource at a given single-unit allocation. We accept that preferences vary over time and with the type of resource. For example, when dealing with task allocations, at a given time, a node may have free bandwidth but full utilization of its CPU, while its situation could be the opposite at another time. Hence, at a particular instant, a node may have greater ability to perform tasks involving

[1] We will use the terms node, user, and player interchangeably.

communication, while at a later time its state may lead it to prefer tasks more intensive in CPU computation.

In other words, we define the preference as some kind of metric measuring the opportunity of receiving a particular resource at a given time. In the case of task allocations, the preference can be seen as some type of execution cost. In Game Theory, closely related to costs and preferences, there is the notion of utility. Hence, the utility may be defined as the cost savings associated with a work not done. In other works, a player gets more utility whenever it avoids running tasks (by having other nodes running them).

When trying to formalize a model based on these principles, a number of problems arise. First, a player's preferences are only known by the player herself. To illustrate this with our previous example, note that in a distributed computational system it would be difficult to audit or check if a given particular node has more or less CPU capacity. This concept is usually known as *private information.* To obtain the private information of a player, the basic mechanism is to directly ask for it and expect the player to declare the value honestly. Following this, we assume that each player can alter her behavior for her own benefit. She may try to maximize her own payoff without concern for others; in other words, she could be *selfish.* One way to maximize her benefit is declaring false information trying to avoid undesirable results (i.e., she can be *untruthful*). This behavior is one of the factors that may distort the internal workings of the game. (The loss of performance produced by selfish players is called the *price of anarchy* [23,31].) In this work, we assume that the users are selfish and potentially untruthful.

In the same way, providing a notion of fairness also presents challenges. Fairness is, in general, an elusive concept that can be seen from many different perspectives. In our case, fairness refers to some kind of compensations to players in exchange for sacrificing their utility. In game theory, this type of compensation usually takes the form of payments. However, in this work, we do not allow payments. Hence, the implementation of our notion of fairness needs to be based on other assumptions, and in particular, we base it here in *repeated games*[2]. In a repeated game, fairness can be provided by compensating current sacrifices in future iterations. Following this, and coming back to the task allocation example, we could implement fairness by guaranteeing that all players execute, on average, the same number of tasks; or that they invest, on average, the same effort (i.e., cost) on their executions. In this work we assume infinitely repeated games, which by definition have no known end.

Previous work in the area of mechanism design for distributed systems [33] has already faced these problems dealing with tasks allocations in the presence of selfish or irrational independent player. However, when players are dependent (i.e., the distribution of their declared preferences are correlated) those solutions cannot guarantee the desired properties in terms of fairness and

[2] By repeated games we refer to a scenario in which players interact by playing a similar (stage) game several times. Unlike a game played once, a repeated game allows for new strategies to be contingent on past moves, thus allowing for reputation and retribution effects.

efficiency. Therefore, this paper develops a mechanism that achieves fairness and efficiency in the presence of selfish players with correlation.

1.2 State of the Art

Mechanism Design has been gaining increasing popularity in distributed computing during the last few years (see, e.g., [27,16,15]). Even though the mechanisms proposed in these works are interesting, they are usually based on payment systems. Deploying such payment systems in practice is often difficult as noted by Schummer and Vohra [26, Chapter 10][3]. When it applies to computational settings, the payments are simply too difficult to enforce, mainly due to security problems or banking issues. For this reason, mechanisms without payments have also been proposed. Related literature could be found in economics on cooperation [21,7] or similar problems in P2P systems such as reputation [20] and artificial currencies [12]. The work closest to our own, and in which we have based our proposal, is the *linking mechanism* proposed by Jackson and Sonnenschein [17,18]. Related to this work, Engelmann and Grimm [8] presents experimental research on linking mechanisms. An algorithm called QPQ (Quid Pro Quo) [33] has been proposed as an application of this kind of mechanisms to distribute task executions fairly among independent players.

QPQ reflects the main idea behind the concept of linking mechanism: when a game consists of multiple instances of the same basic decision problem (e.g., saying yes or no, choosing among a number of discrete options), it is possible to define selfishness-resistant algorithms by restricting the players' responses to a given distribution. Hence, in that case, the frequency with which a player declares a particular decision is established beforehand. Based on this, QPQ presents quite relevant features as the fact of not requiring payments, the flexibility on the definitions of the utility functions of the players, its applicability in repeated games, the lack of central control authority, etc. While QPQ presents some very interesting properties, it only guarantees fairness and efficiency when users behave independently on each other. Nevertheless, this does not need to be the case in real environments, where users may have correlated preferences. The problem of fairness among players has been widely analyzed in the game theoretical literature and a wide range of fairness concept has been proposed, but, as far as we know, there is no fair linking mechanisms when players have correlated preferences. This motivates the research proposed in this paper.

[3] Schummer and Vohra note that " there are many important environments where money cannot be used as a medium of compensation. This constraint can arise from ethical and/or institutional considerations: many political decisions must be made without monetary transfers; organ donations can be arranged by 'trade' involving multiple needy patients and their relatives, yet monetary compensation is illegal."

1.3 Contributions

Our contributions are twofold. On the one hand, we have extended the idea of linking mechanism introducing fairness and correlation between players, while preserving desirable properties, like efficiency, truthful reporting, incentive compatibility, etc. On the other hand, we propose an algorithm based on these mechanisms that we expect to be used in practical scenarios.

As mentioned, in [33] we have proposed a mechanism that we called *Quid Pro Quo (QPQ) Mechanism* . QPQ presents several interesting properties in relation to the efficiency in presence of selfish or rationally-limited players. However, those properties only hold when players are independent. In this paper, we extend the linking mechanism framework (and therefore QPQ) by introducing a novel scheme providing fair and optimal resource allocation even when players' preferences are correlated.

In our model, fairness is a key element introduced to compensate current sacrifices in future iterations. Due to the large number of notions of fairness that could be defined, it is difficult to find a general model that encompasses any approach. In this work we have proposed a generic fairness definitions, which we hope will serve as a reference to wider models. Hence, our contribution is clear: to the best of our knowledge, no other previous research work has offered a linking mechanism providing fair and efficient decision.

In addition, from a theoretical perspective, we contribute to the progress of the state-of-the-art by proposing a mathematical framework suitable for proving all claimed algorithmic properties. This framework is inspired on previous work on theoretical economics but, as far as we know, it has never been adapted to the specific peculiarities of distributed computing (at least not to solve the resource allocation problem). This technique has proven to be extremely powerful for our specific problem, but it can be re-used in other scenarios with similar assumptions.

Based on the theoretical results, we propose a realization of the mechanism suitable for being implemented as a repeated game in real distributed environments. Unlike in the original linking mechanism, this algorithm does not need to know the probability distribution of the players' responses. We show that this realization does not require central entities and that its computational cost is affordable for current state of the art networks and devices. In addition, through simulations, we confirm the stability of the algorithm demonstrating that few iterations on a repeated game are enough for making the mechanism to converge to a fair resource allocation even when the players' distributions are strongly correlated.

1.4 Structure

The rest of the paper is structured as follows. First, in Section 2 we define the basic mathematical concepts derived from mechanism design that are required for the understanding of our work. Then, in Section 3 we present two especial cases of the problem at hand, to illustrate it and provide intuition on how it is solved. After that, in Section 4 we present the generic QPQ mechanism. In this section we use calculus of variations to prove that a this generic QPQ mechanism can be used to performing efficient task allocation even when players have correlated preferences and behave in selfish or non-rational ways. Based on the generic QPQ mechanism, we propose in Section 5 a practical implementable algorithm suitable to be used in real-live scenarios with tractable computational and communication complexity. In Section 6 we show through simulations that this practical algorithm converges in few iterations and maintains the analytical properties of the generic QPQ mechanism. The conclusions of the paper are presented in Section 7.

2 Model and Definitions

We start by presenting the usual mathematical framework for mechanism design and then we formally define the specific problem we face in this paper.

2.1 Mechanism Design Concepts

The following provides the usual theoretic framework that will be later applied to our problem. We assume that there are n players. The set of players is $N = \{1, 2, \ldots, n\}$. Players are risk-neutral. The alternative or outcome set of the game played is D. In a general setting, D could be defined over $\Delta(N)$ [4], but in this paper we define $D = N$ so that the outcome $d \in D$ is the player to whom the resource will be allocated.

Prior to making the collective choice in the game, each player privately observes her preferences over the alternatives in D. This is modeled by assuming that player i privately observes a parameter or signal θ_i that determines her preferences. (For instance, in resource allocation, θ_i could represent the value player i assigns to the resource.) For a given player i, we say that θ_i is the player type. The set of possible types of player i is Θ_i. We denote by $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ the vector of player types. The set of all possible vectors is $\Theta = \Theta_1 \times \Theta_2 \times \ldots \Theta_n$. We denote by θ_{-i} the vector obtained by removing θ_i from θ.

We denote by $\Pi = \Delta(\Theta)$ the set of all probability distributions over Θ. It is assumed that there is a common prior distribution $\pi \in \Pi$ that is shared by all the players. We denote by $\pi_i \in \Delta(\Theta_i)$ the marginal probability of θ_i. We define $\beta_i(\theta_{-i}|\theta_i)$ as the conditional probability distribution of θ_{-i} given θ_i. That is, for any possible type $\theta_i \in \Theta_i$, $\beta_i(\cdot|\theta_i)$ specifies a probability distribution over the set Θ_{-i} representing what player i would believe about the types of the other players if her own type were θ_i. Beliefs $(\beta_i)_{i \in N}$ are *consistent*, since individual belief functions β_i can all be derived from the common prior π. This implies that $\pi(\theta_{-i}|\theta_i) = \beta_i(\theta_{-i}|\theta_i)$.

[4] We denote by $\Delta(S)$ the set of all probability distribution over some set S.

Individual players have preferences over outcomes, which are represented by a utility function $u_i(d, \theta_i) \in \mathbb{R}$ defined over all $d \in D$ and $\theta_i \in \Theta_i$.

The set of outcomes D, the set of players N, the type sets in Θ, the common prior distribution $\pi \in \Pi$, and the payoff functions $u_i, i \in N$ are assumed to be *common knowledge* among all the players. The game rules defined by a specific mechanism are also common knowledge. However, the specific value θ_i observed by player i is *private information* of player i.

A strategy for the player i is any map $\sigma_i : \Theta_i \to \Delta(\Theta_i)$, where $\sigma_i(\hat{\theta}_i|\theta_i)$ is the conditional probability that the player reports $\hat{\theta}_i$ when her true type is θ_i. A reporting strategy σ_i is *truthful* if for every pair $(\hat{\theta}_i, \theta_i)$, the probability is concentrated at $\hat{\theta}_i = \theta_i$. As usual, we will use $\hat{\theta}_i$ to denote the reported type and θ_i the actual type.

Given that the prior distribution π is known, player i can not change it. Hence, we say that a player i has a *limited strategy space*, since her strategy can not change the beliefs of other players. Intuitively, player i has a limited strategy space if beliefs over reports are the same as actual beliefs.

As stated before, we consider games with private information; those games are also called *Bayesian* [14]. In such games, players have limited information about certain aspects of the game (e.g. payoff functions). However, each player has a subjective probability distribution over this information. *Bayesian mechanism* design governs the design and analysis of these game [16]. Bayesian Equilibrium is the fundamental solution concept for Bayesian games [25]. Informally, a *Bayesian equilibrium* is a set of strategies, and beliefs, so that no player has a profitable deviation. We say that a mechanism is *Bayesian incentive compatible* when, truthfully revealing private information is a Bayesian equilibrium.

In this work we design Bayesian mechanisms. For a given Bayesian mechanism $\langle \Theta, g \rangle$ we shall write $q_i(\cdot|\theta_i)$ for player i's interim probability density function on D conditional on player i's type being θ_i.

In this paper, we are looking for a mechanism $\langle \Theta, g \rangle$, where $g(\cdot)$ is the decision function, without utility transfers (payments) and that implements some social choice function f under equilibrium when the induced game is Bayesian. In addition, we introduce fairness as a key tool to compensate or reward players. We call this kind of mechanisms *Quid Pro Quo (QPQ) Mechanisms.*

2.2 Fairness

In our model, we use fairness as a very abstract concept. For us, fairness is the property of balancing in expectation some game parameters (modelled with a real function) among all players. Our model was originally built with two examples in mind: *fairness in utility* ("players have same expected utility") and *fairness in assignment* ("same expected number of assignments"). But these two examples are just special cases of our model. Additionally, we have contemplated the possibility that some scenarios require allocations other than equiproportional; or that the game must be constrained to several fairness concepts at the same time. All this is modelled by introducing a set of functions $\eta_{i,l} : \theta \rightarrow \mathbb{R}$ and ratios $\delta_{i,l}$, all defined for each player $i \in N$ and for each fairness concept $l = 1, \ldots, m$ (m is the number of fairness concepts). The function $\eta_{i,l}$ represents a fairness concept. For instance, for *fairness in assignment* this function could be defined as $\eta_{i,l}(\theta) = 1$. Similarly, *fairness in utility* is applied when $\eta_{i,l}(\theta) = \theta_i$. On the other hand, $\delta_{i,l}$ is the ratio for player i when fairness l is applied. Typically, this ratio is $\delta_{i,l} = \frac{1}{n}$. Then, formally, our concept of fairness is defined as follows.

Definition 1 (Fairness) *Given functions $\eta_{i,l} : \Theta \rightarrow \mathbb{R}$, and values $\delta_{i,l}$, we say that a mechanism $\langle \Theta, g \rangle$ is fair (or η-fair) when, for all $i \in N$ and $l = 1, \ldots, m$,*

$$\int_{\Theta} \eta_{i,l}(\theta) q_i(\theta) \; d\pi(\theta) = \delta_{i,l} \sum_{j \in N} \int_{\Theta} \eta_{j,l}(\theta) q_j(\theta) \; d\pi(\theta) \qquad (1)$$

In this paper, we deal mathematically with this general concept of fairness, but for the algorithm and simulations we used a particular concept of fairness, where players will have *equal number* of allocated resources (in expectation).

2.3 Resource Allocation Problem

We now formally define the problem we study in this work. Intuitively, the problem is like a repeated game where the stage game is in fact a single-unit auction [24], and the mechanism that decides how to allocate the resource in each auction is a QPQ mechanism. Hence there are no payments and the allocation must satisfy a notion of fairness.

The problem of resource allocation is a tuple $\langle R, N, \Theta \rangle$ where, N and Θ are as defined above, and $R = \{r_1, r_2, \ldots\}$ is the ordered set of resources that have to be allocated by the system over time. Resources are received by the system in their order in R, they are independent among them, and the system must allocate resource r_k to a single player before receiving resource r_{k+1}. R is assumed to be infinite.

As mentioned previously, in this problem the outcome set is $D = N$, where an outcome of $d \in D$ for resource r_k means that r_k is allocated to player d. In [33], we have proposed a QPQ algorithm that implements this function when the type of players follow mutually independent distributions. As in that work, we assume here that the type of each player is normalized using a *Probability Integral Transform* (PIT, see Section 5), so that it takes real values in the interval $[0, 1]$ and follows a uniform distribution within that support. Hence, we assume that $\Theta_i = [0, 1]$. Finally, as mentioned, we assume that players have a limited space strategy (i.e., π is known a priori and cannot be changed by the players).

The social choice function (scf) $g(\cdot)$ we are looking for is one that optimizes the social utility restricted by fairness conditions. The social choice function (scf) is represented by the assigment $q_i(\theta)$ that maximizes the functional,

$$F[q] \equiv \sum_{i \in N} \int_{\Theta} u_i(d(\theta), \theta_i)\ q_i(\theta)\ d\pi(\theta) \tag{2}$$

s.t.

$$\int_{\Theta} \eta_{i,l}(\theta) q_i(\theta)\ d\pi(\theta) = \delta_{i,l} \sum_{j \in N} \int_{\Theta} \eta_{j,l}(\theta) q_j(\theta)\ d\pi(\theta), l = 1, \ldots, m, \forall i \in N. \tag{3}$$

In this functional, (2) is the expected social utility, which is simply the sum of the expected utilities of the players. On its hand, (3) are the fairness restrictions to be satisfied. In this formulation there are m such restrictions. Each restriction l and player i has its own fairness functions $\eta_{i,l}(\cdot)$, and a parameter $\delta_{i,l}$ that defines the proportion of the total fairness that must be "assigned" to a player i. In order to find the assignment function $q_i(\theta)$ that maximizes this functional, calculus of variations can be used.

As an example, we study the fairness concept where each player i will receive a *proportional number of resources* δ_i. Hence, we obtain that the scf is the assigment of $q_i(\theta)$ that maximizes the functional

$$F[q] \equiv \sum_{i \in N} \int_{\Theta} u_i(d, \theta_i)\ q_i(\theta)\ d\pi(\theta), \tag{4}$$

s.t.

$$\int_{\Theta} q_i(\theta)\ d\pi(\theta) = \delta_i, \forall i \in N. \tag{5}$$

Another fairness concept that we study as an instance of this framework is *players with proportional utility*. Under this fairness concept every player will obtain a proportional expected utility. The equations are similar in this case.

$$F[q] \equiv \sum_{i \in N} \int_{\Theta} u_i(d, \theta_i)\ q_i(\theta)\ d\pi(\theta), \tag{6}$$

s.t.

$$\int_{\Theta} u_i(d, \theta_i)\ q_i(\theta)\ d\pi(\theta) = \delta_i \sum_{j \in N} \int_{\Theta}\ u_j(d, \theta_j)\ q_j(\theta)\ d\pi(\theta), \forall i \in N. \tag{7}$$

Without loss of generality, we can define the utility of a player i as follows,

$$u_i(d, \theta_i) = \begin{cases} \theta_i & \text{if } d = i, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

In this paper, we are interested in dynamic mechanisms where truth-telling is a Bayesian equilibrium of the static QPQ mechanism. In that case we call the QPQ mechanism Bayesian incentive compatible. That means that a player obtains a higher utility when reporting truthfully.

3 Simple Resource Allocation Examples

In order to illustrate the problem we face in this paper, and the techniques used to solve it, we believe that it is useful to provide simpler versions of the problem first. These examples of the resource allocation problem are simpler than the general case in two aspects: they consider only two players and they consider the specific versions of fairness mentioned above. Although simpler, the resulting problems have similar structures as the general one, and the lessons that are extracted from them suggest the general treatment.

3.1 Same Number of Resources

We consider first the problem of distributing a large (infinite) number of resources between two players so that both players get the same proportion of resources, and the aggregate utility is maximized. The players have utilities for each possible decision and these valuations may be correlated. Suppose that we choose as decision function for every resource the assignment that maximizes the sum of the utilities. The difficulty is that this social choice function is not fair. In Figure 1 we show four examples of possible joint distribution of the two players (shown as density and contour plots). If we assign resources using as decision function $\arg\max_{i \in N}(\theta)$, the result may be clearly unfair.

So the idea is to find a new decision function that takes into account the joint distribution π and assigns resources optimally and fairly. For the moment, we will assume that this decision function is deterministic and it could be represented by a line. This line divides the square into two areas: the area where resources are assigned to player 1 and the area for player 2. At this moment, the shape of this line is unknown, but suppose that this decision function looks somewhat like the line presented in Figure 2 (left). Note that the decision function can be expressed as $\arg\max_{1,2}(\psi(\theta_1), \theta_2)$, where ψ is a transformation function obtained from the line in Figure 2 and the joint distribution π. With all this in mind, the problem we aim to solve is to find the function ψ that maximizes:

$$\int_0^1 \theta_1 \int_0^{\psi(\theta_1)} \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 + \int_0^1 \theta_2 \int_0^{\psi^{-1}(\theta_2)} \pi(\theta_1, \theta_2) d\theta_1 d\theta_2, \tag{9}$$

s.t.

$$\int_0^1 \int_0^{\psi(\theta_1)} \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 = \tfrac{1}{2}. \tag{10}$$

Note that the left term of (9) is the utility of Player 1, which corresponds to the blue areas in the sample assignments presented in Figure 2. Similarly, the right term of (9) is the utility of Player 2, and corresponds to the white

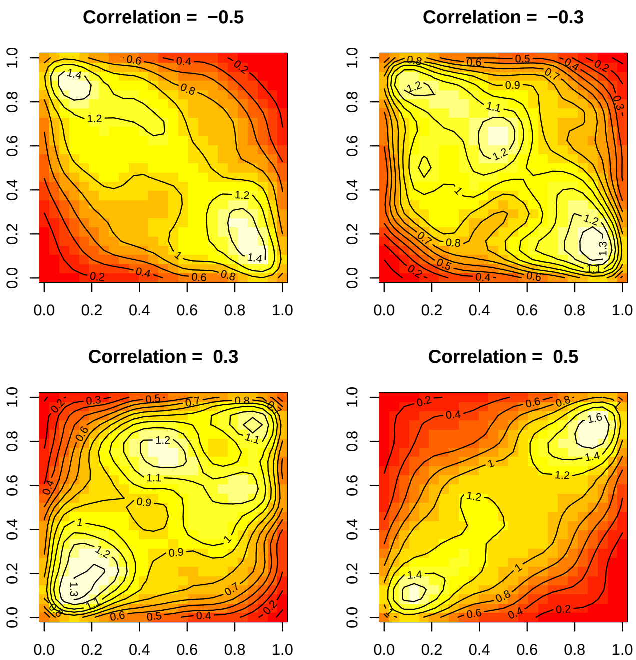

Fig. 1 Example of join distributions π for two correlated players. The preferences are normalized in $[0, 1]$. Colors represent density.

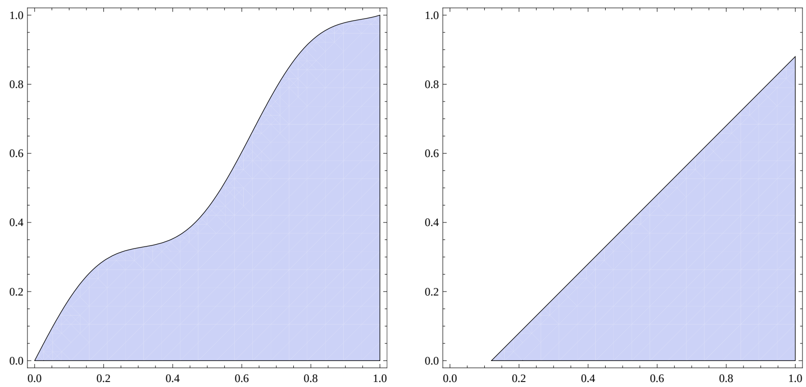

Fig. 2 (Left) Posible decision function for two players. (Right) Example of a decision function ψ for two players and same number of resources fairness. Colors represent different areas assigned to each player.

areas. In addition, if the objective is to minimize utility (for instance, costs) the problem has a similar formulation:

$$\int_0^1 \theta_1 \int_{\psi(\theta_1)}^1 \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 + \int_0^1 \theta_2 \int_{\psi^{-1}(\theta_2)}^1 \pi(\theta_1, \theta_2) d\theta_1 d\theta_2, \tag{11}$$

s.t.

$$\int_0^1 \int_{\psi(\theta_1)}^1 \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 = \tfrac{1}{2}. \tag{12}$$

In this case the left term of (11) corresponds to the white areas in Figure 2 and the right term to the blue areas. Observe that both formulations can be solved using calculus of variations. Moreover, they both reach the same solution $\psi(x)$ (see Appendix A), only the decision function is different (max versus min). Solving the system with respect to $\psi(x)$ (see Appendix A), we obtain the solution of this system as

$$\psi(\theta_1) = \theta_1 + \lambda, \tag{13}$$

for some suitable constant λ. This confirms that the decision function is based on a straight line. The right plot in Figure 2 represents this idea.

3.2 Proportional Utility

In the second special case we present, the fairness criterion to be satisfied is assigning resources with "proportional expected utility" for each player. In particular, we want both players to get the same expected utility. In that case, the functional to optimize (for the minimization case) becomes

$$\underset{\psi}{min} \left(\int_0^1 \theta_1 \int_{\psi(\theta_1)}^1 \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 + \int_0^1 \theta_2 \int_{\psi^{-1}(\theta_2)}^1 \pi(\theta_1, \theta_2) d\theta_1 d\theta_2\right) \tag{14}$$

s.t.

$$\int_0^1 \theta_1 \int_{\psi(\theta_1)}^1 \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 = \int_0^1 \theta_2 \int_{\psi^{-1}(\theta_2)}^1 \pi(\theta_1, \theta_2) d\theta_1 d\theta_2 \tag{15}$$

Finding the solution involves using similar mathematical techniques as in the previous case. The function $\psi(x)$ obtained is,

$$\psi(\theta_1) = \lambda\theta_1, \tag{16}$$

for a suitable constant λ.

Note that in the two examples we have obtained a separatrix with the form of a straight line. In fact, this will hold also for the general case presented in the next section.

4 The Fair Quid Pro Quo Mechanism

We now derive the general QPQ Mechanisms that implement the social choice functions given by Eqs. 4 and 6 under equilibrium, as special cases of the solution to Eq. 2.

Theorem 1 *The QPQ Mechanism that implements the social function of Eq. 2 with η-fairness is a set of functions $\psi = (\psi_1, \cdots \psi_n)$ that defines a line $y = \psi_i(x)$ for each player i with deterministic assignment $d = g_\psi(\theta) = \arg\max_{i \in N}(\psi_i(\theta))$ (except at some points where the decision is indifferent).*

Proof The problem we aim to solve is to find the decision function g that maximizes

$$\int_\Theta \sum_{i \in N} \theta_i \; q_i(\theta) \; d\pi(\theta), \tag{17}$$

under the constraints given in eqn. 2. Using Lagrange multipliers, this is tantamount to maximizing the functional

$$\mathcal{F}[q] \equiv \int_\Theta \sum_{i \in N} \theta_i \; q_i(\theta) \; d\pi(\theta) + \\ \sum_{k \in N} \sum_{l=1}^{m} \lambda_{k,l} \int_\Theta \left\{ \eta_{k,l}(\theta) q_k(\theta) - \delta_{k,l} \sum_{j \in N} \eta_{j,l}(\theta) \; q_j(\theta) \right\} d\pi(\theta), \tag{18}$$

which can be rewritten as

$$\mathcal{F}[q] = \int_\Theta \sum_{i \in N} \psi_i(\theta) q_i(\theta) \; d\pi(\theta), \tag{19}$$

where

$$\psi_i(\theta) \equiv \theta_i + \sum_{l=1}^{m} \lambda_{i,l} \; \eta_{i,l}(\theta) - \sum_{k \in N} \sum_{l=1}^{m} \lambda_{k,l} \; \delta_{k,l} \; \eta_{i,l}(\theta). \tag{20}$$

Let $d = \arg\max_{i \in N}(\psi_i(\theta))$. Since $0 \leq q_i(\theta) \leq 1$ and $\sum_{i \in N} q_i(\theta) = 1$ for all $\theta \in \Theta$, then for each $\theta \in \Theta$,

$$\sum_{i \in N} \psi_i(\theta) q_i(\theta) \leq \psi_d(\theta) \tag{21}$$

The upper bound is reached if, and only if, for that value of θ we have $q_d(\theta) = 1$ and $q_k(\theta) = 0$ for all $k \neq d$. This proves the theorem for the case when the largest value among the $\psi_i(\theta)$ is unique.

Let us consider, on the other hand, that there are several such largest values, i.e., $d_1, \ldots, d_r$ are such that $\psi_{d_1}(\theta) = \cdots = \psi_{d_r}(\theta) > \psi_k(\theta)$ for all $k \neq d_1, \ldots, d_r$. Then the upper bound is $\psi_{d_1}(\theta)$, but this time is reached for any choice of the functions $q_i(\theta)$ such that $q_{d_1}(\theta) + \cdots + q_{d_r}(\theta) = 1$ and $q_k(\theta) = 0$ for all $k \neq d_1, \ldots, d_r$. Hence, any decision among $d_1, \ldots, d_r$ is indifferent. □

For convenience, we build the decision function of our mechanism introducing a *transformation function* $\psi : \Theta \to \mathbb{R}^n$ that returns a vector of n real values. The decision function is then obtained as $d = g(\theta) = g_\psi = \arg\max_{i \in N}(\psi_i(\theta))$. We say that ψ determines the "decision rule" or "decision function". Our main theorem gives us insight into what we can expect about the set of functions ψ.

Given our definition of $\psi_i(\theta)$ we can derive some intuition about the decision function. The theorem tells us that we can restrict our attention to deterministic solutions except when $\psi_i(\theta) = \psi_j(\theta), i, j \in N$. At these points, the decision is indifferent. The above theorem also gives us an optimality result.

Corollary 2 *Assume that all players follow a truthful reporting strategy, mechanism M defined using the decision function $d = \arg\max_{i\in N}(\psi_i(\theta))$ maximizes the utility of the system subject to fairness constraints.*

Finally, when fairness is symmetric in the sense that each player has the same fairness function, then each ψ_i depends only on the player's profile θ_i and therefore $\psi_i(\theta_i, \theta_{-i})$ could be reduced to $\psi_i(\theta_i)$. This last aspect allows us to state the following corollary.

Corollary 3 *When fairness is symmetric in the sense of $\eta_i(\theta) = \eta(\theta_i)$ $\forall i \in N$, and players have limited space strategy, then the probability q_i depends only on the player's value, that is $q_i(\theta) = q_i(\theta_i)$.*

Proof The proof follows from the definition of $\psi_i(\theta)$ and therefore the decision function could be reduced to $d = \arg\max_{i\in N}(\psi_i(\theta_i))$. As beliefs can not be changed by the strategy of others players, the probability $q_i(\theta)$ is only defined as a function of θ_i. □

Revisiting our particular cases of fairness defined as equal-number of resources (Eq. 4) and equal utility (Eq. 6) we can check that the solutions for ψ are in both cases straight lines. When fairness is defined as equal-number of resources (Eq. 4), $\psi_i(\theta)$ becomes (see Eq. 20)

$$\psi_i(\theta) \equiv \theta_i + \lambda_i - \textstyle\sum_{k=1}^{n} \lambda_k \delta_k, \tag{22}$$

and therefore $\psi(\theta_i) = \theta_i + \lambda_i - \sum_{k=1}^{n} \lambda_k \delta_k$.

This solution has a very nice property that was already observed in our original work (QPQ with independent players [33]). The mechanism designer could aggregate players when studying a single player. The mechanism designer can see the game as player i against the system formed by all other players ($j \in N, j \neq i$). In this case, player i has to compute just two values for λ, her own value λ_i and the aggregate value $\lambda_j = \sum_{k=1}^{n} \lambda_k \delta_k$. That is, $\psi(\theta_i) = \theta_i + \lambda_i - \lambda_j$, or even simpler, $\psi(\theta_i) = \theta_i + \lambda$. if we redefine λ as a new single real parameter that represents $\lambda_i - \sum_{k=1}^{n} \lambda_k \delta_k$. This confirms that the decision function is a straight line where the parameter λ determines the point at which the line crosses the y-axis. And this is true for all players. (See Figure 2 in Section 3.)

On the other hand, when fairness is defined as a function of utility (Eq. 6), our ψ function is given by using

$$\psi_i(\theta) \equiv \theta_i \ (1 + \lambda_i - \textstyle\sum_{k=1}^{n} \lambda_k \delta_k), \tag{23}$$

and therefore $\psi(\theta_i) = \theta_i \ (1 + \lambda_i - \sum_{k=1}^{n} \lambda_k \delta_k)$. Again, the decision function is a straight line where λ determines the slope. Aggregating players, the above solution could be reduced to $\psi(\theta_i) = \theta_i \ (1 + \lambda_i - \frac{1}{n}\lambda_j)$, or $\psi(\theta_i) = \lambda \ \theta_i$.

Properties. The Fair QPQ Mechanism with Correlated players (M_{fair}) has the following properties:

1. M_{fair} is (ex-ante) **individual-rational**. This means that the expected utility of a player is at least its expected outside utility.
2. M_{fair} is not **allocative-efficient**, but assigns resources efficiently subject to some fairness conditions. This property is a clear conclusion from Corollary 2.
3. There is no incentive for any of the players to lie about or hide their private information from the other players. Players will report truthfully in a Bayesian equilibrium. We said that M_{fair} is Bayesian incentive compatible.

The two first properties are quite evident. The last property follows from the next theorem.

Theorem 4 *When players have limited space strategy, and fairness is symmetric in the sense that $\eta_i(\theta) = \eta(\theta_i)$ $\forall i \in N$, then M_{fair} is Bayesian incentive compatible.*

Proof For the sake of contradiction, let us suppose this proposition is false. Hence, there is some set of assignments for which, if i does not follow a truthful reporting strategy, she will obtain more utility in expectation.

From Corollary 3, this holds for any strategy of the aggregate player j, and in particular when all her players follow a truthful reporting strategy. Hence, we can consider in the rest of the proof that the rest of $n-1$ players follow a truthful reporting strategy.

Additionally, using the same corollary, we know that every player, $j \neq i \in N$, will obtain the same expected utility (regardeless of whether i lies or not),

$$\int_{\Theta} u_j(d, \theta_j)\ q_j(\theta)\ d\pi(\theta) = \int_{\Theta} u_j(d, \theta_j)\ \hat{q_j}(\theta)\ d\hat{\pi}(\theta)$$

Now we can define a new mechanism M that assigns a task to player i (when i declares θ_i) with the same probability as the original QPQ assigns the task to the player i when she declares a false value $\hat{\theta}_i$. Then, $q_i(\theta_i) = \hat{q}_i(\hat{\theta})$. Note this new mechanism conserves the same fairness constraints as the original one. However, if the above were true, QPQ would not be optimal, since a mechanism that reproduces the same decisions under i lying (in presence of players that follow a truthful reporting strategy) would have different (lower) utility. Clearly, this is in contradiction of optimality of QPQ. Therefore, the best strategy for a player (the one optimizing her normalized utility) is to follow a truthful reporting strategy. □

5 Practical QPQ Algorithm

After describing the different ingredients of our solution, we are able to propose an application of our mechanism. To demonstrate the usability of our mechanism, we discuss an algorithm for a specific case. We propose an algorithm where the resource allocation achieves fairness in the number of resources allocated to each player. This algorithm could be easily extended to other fairness concepts. The details can be observed in Algorithm 1.

Algorithm 1 QPQ Correlated mechanism (code for node i)

Estimate the preference θ_i
Publish the normalized value $\bar{\theta}_i = PIT(\theta_i)$
Wait to receive the normalized values $\bar{\theta}_j$ from the other players
for all $j \in N$ **do**
    if not $GoF\_Test(\bar{\theta}_j, Historic)$ **then**
        $\bar{\theta}_j \leftarrow Random(\bar{\theta}_{-j}, Historic)$
    end if
end for
$Historic \leftarrow Historic \cup \{\bar{\theta}\}$
Let $d = \arg\max_{j \in N}\{\psi_j(\bar{\theta}_j)\}$
if $d = i$ **then**
    Resource is assigned to node i
end if
Update $\lambda_j, \forall j \in N$: $\lambda_{k+1,j} = \lambda_{k,j} + \epsilon_k(T_{k,j} - 1/n)$.

In the algorithm, $T_{k,j}$ denotes the fraction of decisions assigned to player j, computed at round k. As it can be observed, for each round, each player estimates her own value and publishes it. Publication means broadcasting a message with the value to all players (although any other means of distribution, like shared memory, can be used). By assumption, a player sends its value before it receives any of the others (concurrency, which implies that they do not depend from each other), and all the values are correctly received at each player (reliability). Then, the algorithm assigns the resource to the player that publishes the highest value modified by a particular ψ_k.

5.1 Preference Normalization

To normalize the players' preferences we use a transformation called *Probability Integral Transformation* (PIT). Our idea is to use the known fact that any cumulative probability distribution function has in itself a uniform distribution [2]. More formally, the PIT is defined as

Definition 2 (Probability Integral Transformation) *Let X be a continuous random variable with a Cumulative Distribution Function (CDF) F; that is $X \sim F$. Then, the probability integral transformation defines a new random variable Y as: $Y = F(X)$.*

As mentioned above, our interest in the PIT is due to the following lemma.

Lemma 5 (PIT follows a uniform distribution) *Let X be a continuous random variable with CDF F, then F follows a uniform distribution on interval $[0, 1]$. That is, the random variable Y defined by the probability integral transformation $Y = F(X)$ is a normalized uniform distribution.*

Note that X does not need to be a continuous random variable. In the case that the player's costs follow a discrete distribution, it is still possible to perform a similar transformation called *Generalized Distributional Transform* [10], whose properties are equivalent to those of the PIT.

Definition 3 (Generalized Distributional Transform) *Let X be a random variable (not necessarily continuous) with a cumulative distribution probability F and let $V \sim U(0,1)$ be a random variable with uniform distribution in $[0,1]$ independent of X. The* modified distribution function $F(x, \lambda)$ *is defined as $F(x, \lambda) = (1-\lambda)Pr(X < x) + \lambda Pr(X \leq x)$. From this, we can define the general distributional transform of X as $Y = F(X, V)$, which can be proved to be a uniform distribution on the unit interval.*

Proofs of these properties can be found in [10]. Many studies in economics use this definition and its properties, such [5] or [32]. In our case, to simplify the notation, we just call PIT to both transformations independently on whether the base distribution is continuous or discrete.

5.2 Acceptance Test

We are assuming that players are reporting values using a uniform $[0, 1]$ distribution. If their original distribution is not the uniform, we apply here the same normalization transformation proposed in [33] based on the *Probability Integral Transform* (PIT). Given the properties of the PIT, the idea is that any player applying correctly the PIT on her real type distribution, must generate a uniform distribution on the unit interval on her published normalized values. Hence, from the point of view of the mechanism designer, the problem amounts to determining whether these published values follow or not that uniform distribution. There are a wide range of tests that allow checking that. These tests are called Goodness-of-Fit or GoF tests.

Continuing with this argument, we propose to implement the acceptance test of our algorithm by using some GoF test on the declared transformed sequence of values published by the player. Whenever a player declares the values by applying the PIT transformation on her own distribution, these values will be uniformly distributed in the unit interval. In that case (with high probability) the GoF tests will accept the samples. More importantly, this process has an error which tends to zero when the number of samples (rounds) increases for any reasonable value of the threshold. For the study of our analytical results, we assume that GoF tests are perfect and this error is zero.

A tremendous amount of GoF tests have been proposed in the scientific literature. Some of them may be applied over discrete distributions and others over continuous ones. The Kolmogorov-Smirnov (KS) test [22,36] is probably the best-known test when dealing with continuous distributions, basically due to its simplicity. Hence, we propose to use this test as the GoF test of QPQ. However, in contrast to our previous work with independent players, in this case it is necessary to add a second test. The goal for this new test is to check if a player is trying to modify the joint distribution. In this aspect, Copulae has proved to be a very useful tool in the analysis of dependency structures. The concept of copulae was introduced by Sklar [35] and several copula GOF approaches have been proposed in literature [29, 4, 11]. For our work, we have used some GoF tests implemented as R-Cran packages. We note that no approach is always the best.

5.3 Punishment

In the case that a dishonest player tried to lie, one possible strategy is to generate increasing (or decreasing) $\hat{\theta}$ values, so that the PIT transformed values are close to one. However, this type of behavior is quickly detected by the test. In that case, the question is how to establish a punishment. Inspired on previous works on linking mechanisms, the proposal is to reject the value declared by the player and generate a new random value according to the joint prior distribution. At a first sight, this strategy may seem a very poor punishment, given that there is always a chance that the player emerges victorious in her lie. However, this is not only enough to discourage dishonest players, but also a crucial ingredient to guarantee that our mechanism has the right properties, as formally shown in [18, 33] for similar scenarios, and in the simulations below (see Section 6).

5.4 Practical Computation of λ

The above solution reduces the problem to finding the value of λ that adjusts the tasks performed by players. In principle, we can ask the players to declare the joint distribution and calculate that parameter accordingly. But in general, we should not expect to find an analytic equation. That is, it is possible that π does not have an analytical expression, or even if it exists, players must estimate it empirically. There are multiple methods for π estimation, both parametric and nonparametric. The major difficulty with these systems is the convergence speed, making it necessary a large number of samples. There is a relationship between the dimension of the feature and the number of samples needed. In our case, the dimension would be given by the number of players. Fortunately, each player can compute the QPQ mechanism using just only two dimensions (itself and the aggregate system).

However, players do not need to know the joint density function π, they only need to know the function $T(\cdot)$ that indicates the (expected or empirical) number of resources given a parameter λ. We denote by $T(\lambda)$ the number of resources that the player obtains when the decision value ψ is determined by the parameter λ. Again, we can not expect an analytic form for T, but under the right assumptions, we can approximate λ using stochastic approximation methods. Due to the characteristics of the transformation function and noting

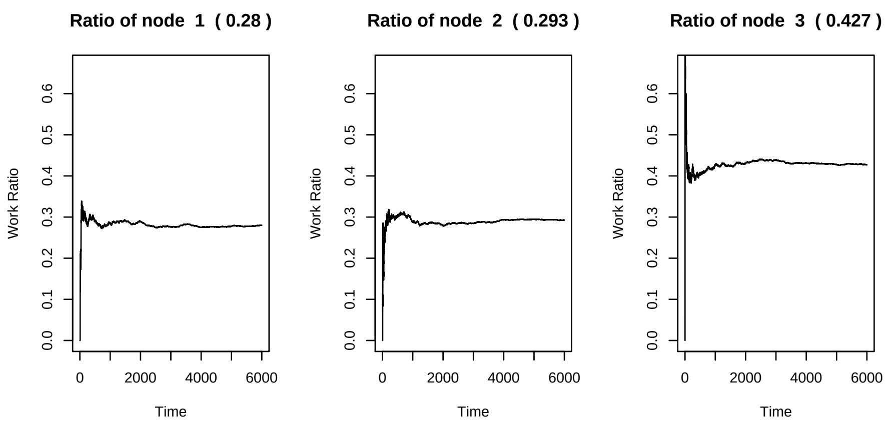

Fig. 3 Evolution of the allocation ratio (ratio of resources allocated to a player) of the different players without QPQ. The simulation has 2 correlated players and 1 independent player. In this example, resources have a negative utility (task to be executed) but similar conclusions apply to resources with positive utility (goods).

how it influences the number of tasks, we can expect the function $T(\lambda)$ to be continuous and decreasing (or increasing in the direction of λ). That is, there is always a value of λ for each percentage of desired tasks. Our proposal is to approximate λ by a sequence $\lambda_0, \lambda_1, \lambda_2, \lambda_3, \dots \to \lambda$ constructed using a stochastic approximation method. The best known method, although not the only one, is perhaps the Robbins-Monro method [30]. Then, our algorithm must compute, for each iteration k,

$$\lambda_{k+1} = \lambda_k + \epsilon_k (T_k - 1/n). \tag{24}$$

Where T_k is an estimation of the fraction number of tasks performed by the player and where ϵ_k is a sequence of values that satisfies $\epsilon_k > 0$, $\epsilon_k \to 0, \sum_k \epsilon_k = \infty$. Note that, in order to estimate T_k we don't need to store previous samples and memory consumption is low.

6 Simulations

By performing simulations, we have checked various aspects of our practical QPQ algorithm. First, we wanted to show that in fact a simple allocation based on the preferences announced and using $\arg\max_{i \in N}(\theta_i)$ to allocate the resource would not be fair if the players were correlated. Figure 3 shows the observed behavior of a simulation applying this technique in a scenario where two players are correlated and the third is independent. As can be seen, the correlated players get less resources. Something worth noticing in this figure is that the distribution of resources gets very stable in only a few rounds.

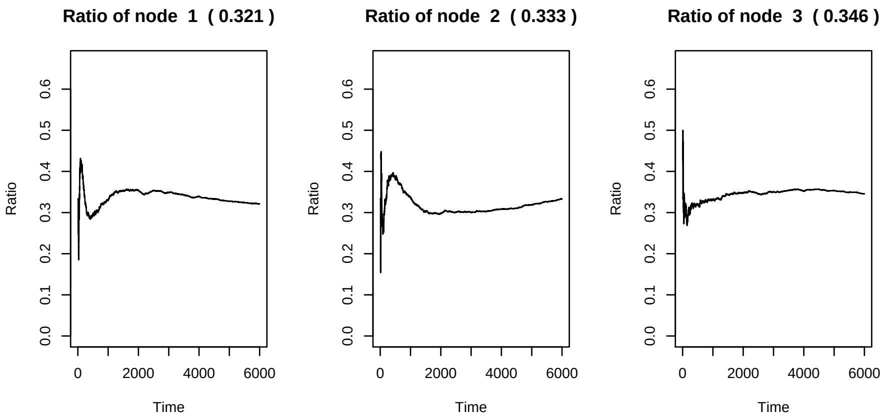

Fig. 4 Evolution of the allocation ratio of the different players with QPQ using $\epsilon_k = 1/k$. The simulation has 2 correlated players and 1 independent player.

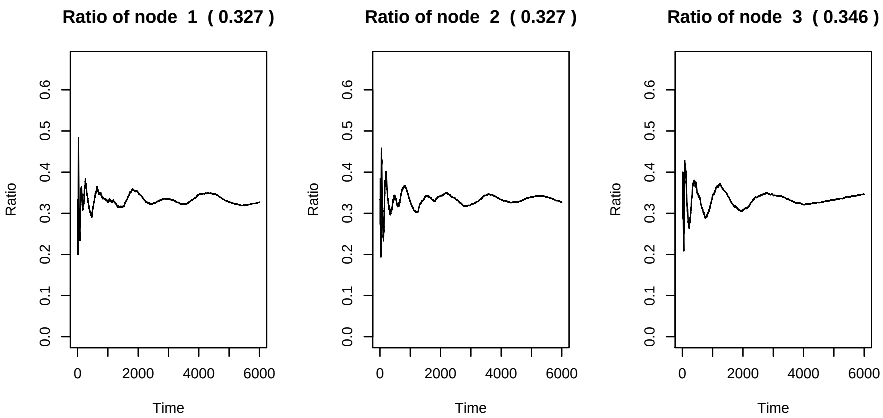

Fig. 5 Evolution of the allocation ratio of the different players with QPQ using $\epsilon_k = 1/\sqrt{k}$. The simulation has 2 correlated players and 1 independent player.

Hence, we have also evaluated the performance of the practical QPQ algorithm presented. For instance, we wondered how Robbins-Monro algorithm performs in time. We have simulated several alternatives for the generation of the sequence of values ϵ_k, starting with the original proposal of Robbins-Monro's work, which was to use $\epsilon_k = 1/k$. The good news is that the simulations show that the allocation of resources is now fair in the scenario with two correlated players described above, as can be seen in Figure 4. As alternative, we have simulated the same scenario using $\epsilon_k = 1/\sqrt{k}$ but the speed of convergence is far from ideal and, as can be seen in Figures 6 and 7, there are significant oscillations in the values of λ.

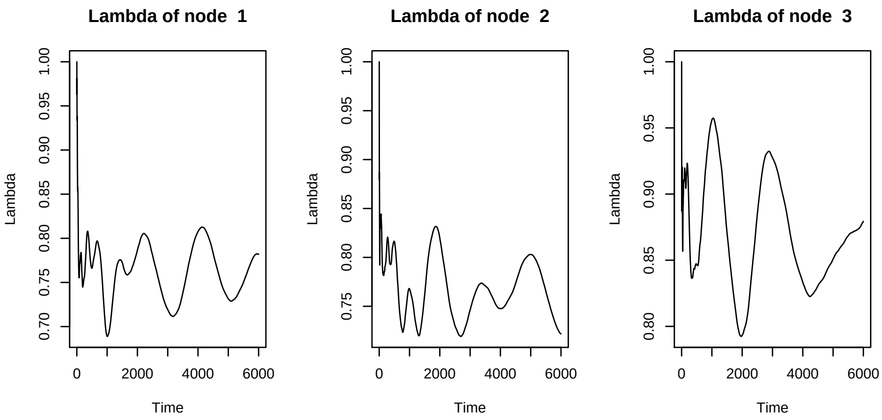

Fig. 6 Evolution of the values of λ with QPQ using $\epsilon_k = 1/\sqrt{k}$. The simulation has 2 correlated players and 1 independent player.

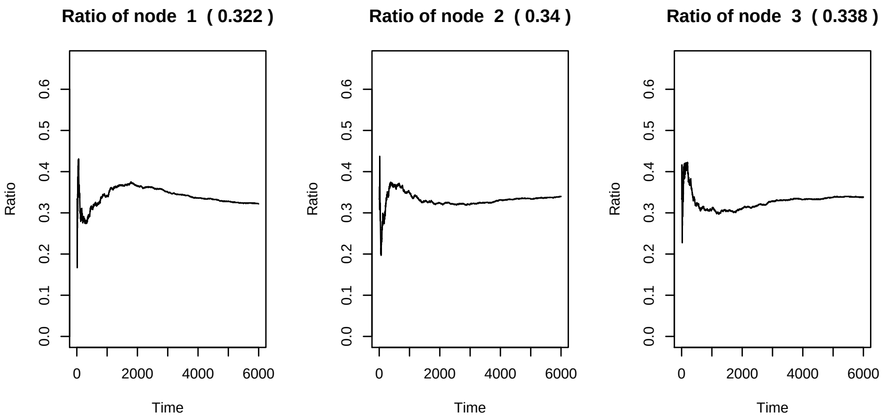

Fig. 7 Evolution of the allocation ratio of the different players with QPQ using $\epsilon_k = \frac{1}{k+\log k}$. The simulation has 2 correlated players and 1 independent player.

Then, we have tried a third alternative, namely $\epsilon_k = \frac{1}{k+\log k}$. This approach gave much better results in the stability of the λ-values, the distribution of resources, and the convergence time. This can be observed in Figures 7 and 8.

7 Conclusions and Future Work

In this paper we have created a novel scheme capable of providing efficient resource allocation in distributed systems even in the presence of selfish correlated players. We have shown that, for a general notion of fairness, the mechanism can be proved to perform efficiently and to maintain the incentive

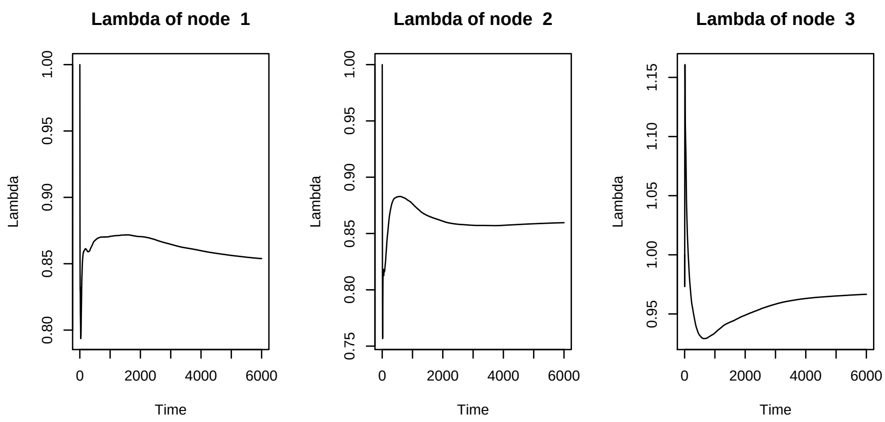

Fig. 8 Evolution of the values of λ with QPQ using $\epsilon_k = \frac{1}{k+\log k}$. The simulation has 2 correlated players and 1 independent player.

of players to participate. In addition, we have proposed a specific realization of the mechanism as an algorithm implementable in real distributed environments with affordable computational and communication costs. This algorithm is susceptible of being used in repeated task allocations given that our simulations demonstrate its rapid convergence, which open new horizons for systems based on open systems for distributed collaborative tasks execution.

Despite this, the authors consider necessary to extend the current research is several directions. First, The algorithm proposed leads to an allocation of equal number of resources to each player. It is left as future work to devise algorithms that have other fairness objectives. Second, the model requires knowledge on the number of players that participate. We may find scenarios where this is not reasonable, e.g., scenarios in which several players "hide" and play the game with a single identity, which may result in the mechanism not achieving fairness. Third, it would be important to analyze the problem when more flexible space strategies are possible. One of our main assumptions has been to consider that correlations are fixed and that players are not able to alter them through their strategies. This assumption is reasonable when information is private and the mechanism is designed in such a way that players cannot make their declared (true or false) values on an iteration dependent on the values of others at the same iteration. However, there are many real-life scenarios where players may be able to share their values making more complex interdependent strategies possible. This would break the properties of our proposed algorithm.

Acknowledgements

We would like to thank the anonymous referees for their useful comments.

References

1. Ahmad, I., Ranka, S., Khan, S.U.: Using game theory for scheduling tasks on multi-core processors for simultaneous optimization of performance and energy. In: Proceedings of the IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2008, pp. 1–6. IEEE (2008)
2. Angus, J.E.: The probability integral transform and related results. SIAM Review **36**(4), 652–654 (1994). DOI 10.1137/1036146
3. Bell, M.G.: A game theory approach to measuring the performance reliability of transport networks. Transportation Research Part B: Methodological **34**(6), 533–545 (2000)
4. Berg, D.: Copula goodness-of-fit testing: an overview and power comparison. The European Journal of Finance **15**(7-8), 675–701 (2009)
5. Burgert, C., Rüschendorf, L.: On the optimal risk allocation problem. Statistics & Decisions **24**(1/2006), 153–171 (2006)
6. Camerer, C.: Behavioral game theory: Experiments in strategic interaction. Princeton University Press (2011)
7. Ellison, G.: Cooperation in the prisoner's dilemma with anonymous random matching. The Review of Economic Studies **61**(3), 567–588 (1994)
8. Engelmann, D., Grimm, V.: Mechanisms for Efficient Voting with Private Information about Preferences. The Economic Journal **122**(563), 1010–1041 (2012)
9. Fang, Z., Bensaou, B.: Fair bandwidth sharing algorithms based on game theory frameworks for wireless ad-hoc networks. In: Proceedings of the Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004, vol. 2, pp. 1284–1295. IEEE (2004)
10. Ferguson, T.S.: Mathematical statistics: A decision theoretic approach. Academic press (2014)
11. Fermanian, J.D.: An overview of the goodness-of-fit test problem for copulas. In: Copulae in Mathematical and Quantitative Finance, pp. 61–89. Springer (2013)
12. Friedman, E.J., Halpern, J.Y., Kash, I.: Efficiency and nash equilibria in a scrip system for p2p networks. In: Proceedings of the 7th ACM conference on Electronic commerce, pp. 140–149. ACM (2006)
13. Gerkey, B.P., Matarić, M.J.: A formal analysis and taxonomy of task allocation in multi-robot systems. The International Journal of Robotics Research **23**(9), 939–954 (2004)
14. Harsanyi, J.C.: Games with incomplete information played by ?bayesian? players, i–iii: Part i. the basic model&. Management science **50**(12_supplement), 1804–1817 (2004)
15. Jackson, M.O.: A crash course in implementation theory. Social choice and welfare **18**(4), 655–708 (2001)
16. Jackson, M.O.: Mechanism theory. Available at SSRN 2542983 (2014)
17. Jackson, M.O., Sonnenschein, H.F.: The Linking of Collective Decisions and Efficiency. Microeconomics 0303007, EconWPA (2003). URL `http://ideas.repec.org/p/wpa/wuwpmi/0303007.html`
18. Jackson, M.O., Sonnenschein, H.F.: Overcoming Incentive Constraints by Linking Decisions. Econometrica **75**(1), 241–257 (2007). DOI 10.1111/j.1468-0262.2007.00737.x
19. Jun, S., Ahamad, M.: Incentives in BitTorrent induce free riding. In: Proceedings of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems, P2PECON '05, pp. 116–121. ACM, New York, NY, USA (2005). DOI 10.1145/1080192.1080199. URL `http://doi.acm.org/10.1145/1080192.1080199`
20. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The eigentrust algorithm for reputation management in p2p networks. In: Proceedings of the 12th international conference on World Wide Web, WWW 2003, pp. 640–651. ACM (2003)
21. Kandori, M.: Social norms and community enforcement. The Review of Economic Studies **59**(1), 63–80 (1992)
22. Kolmogorov, A.N.: Sulla Determinazione Empirica di una Legge di Distribuzione. Giornale dell'Istituto Italiano degli Attuari **4**, 83–91 (1933)
23. Koutsoupias, E., Papadimitriou, C.H.: Worst-case equilibria. Computer Science Review **3**(2), 65–69 (2009)
24. Krishna, V.: Auction theory. Academic press (2009)

25. Myerson, R.B.: Bayesian equilibrium and incentive-compatibility: An introduction. Social goals and social organization: Essays in memory of Elisha Pazner pp. 229–260 (1985)
26. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic game theory, vol. 1. Cambridge University Press Cambridge (2007)
27. Papadimitriou, C.H.: Games, algorithms, and the Internet. In: S. Srinivasan, K. Ramamritham, A. Kumar, M.P. Ravindra, E. Bertino, R. Kumar (eds.) Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011, pp. 5–6. ACM (2011)
28. Procaccia, A.D., Tennenholtz, M.: Approximate mechanism design without money. ACM Transactions on Economics and Computation **1**(4), 18 (2013)
29. Rakonczai, P., Zempléni, A.: Copulas and goodness of fit tests. Recent Advances in Stochastic Modeling and Data Analysis pp. 198–206 (2007)
30. Robbins, H., Monro, S.: A Stochastic Approximation Method. The Annals of Mathematical Statistics **22**(3), 400–407 (1951)
31. Roughgarden, T.: Selfish Routing and the Price of Anarchy, vol. 174. The MIT Press, Cambridge (2005)
32. Rüschendorf, L.: On the distributional transform, Sklar's theorem, and the empirical copula process. Journal of Statistical Planning and Inference **139**(11), 3921–3927 (2009)
33. Santos, A., Fernández Anta, A., López Fernández, L.: Quid Pro Quo: A Mechanism for Fair Collaboration in Networked Systems. PloS one **8**(9), e66,575 (2013)
34. Shehory, O., Kraus, S.: Methods for Task Allocation via Agent Coalition Formation. Artificial Intelligence **101**(1), 165–200 (1998)
35. Sklar, A.: Fonctions de répartition à n dimensions et leurs marges. Publications de l'Institut de Statistique de l'Universit de Paris **8**, 229–231 (1959)
36. Smirnov, N.V.: On the estimation of the discrepancy between empirical curves of distribution for two independent samples. Bull. Math. Univ. Moscou **2**, 3–16 (1939)
37. Srivastava, V., Neel, J.O., MacKenzie, A.B., Menon, R., DaSilva, L.A., Hicks, J.E., Reed, J.H., Gilles, R.P.: Using game theory to analyze wireless ad hoc networks. IEEE Communications Surveys and Tutorials **7**(1-4), 46–56 (2005)

A Solution of the System with Same Number of Resources

We prove here the following theorem.

Theorem 6 *The function ψ that optimizes the assignment with equal expected number of resources for two players defines a straight line $\psi(\theta_1) = \theta_1 + \lambda$. The decision function is an assignment $d = g_\psi(\theta) = \arg\max_{1,2}(\psi_1(\theta_1), \theta_2) = \arg\max_{1,2}(\theta_1 + \lambda, \theta_2)$ when the objective is to maximize the utility, and $d = g_\psi(\theta) = \arg\min_{1,2}(\psi_1(\theta_1), \theta_2) = \arg\min_{1,2}(\theta_1 + \lambda, \theta_2)$ when the objective is to minimize.*

Proof What we want to prove is that the solution to the following system is $\psi_1(x) = x + \lambda$.

$$\begin{aligned} &\underset{\psi}{min}\ (\textstyle\int_0^1 \theta_1 \int_{\psi(\theta_1)}^1 \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 + \int_0^1 \theta_2 \int_{\psi^{-1}(\theta_2)}^1 \pi(\theta_1, \theta_2) d\theta_1 d\theta_2), \\ &\text{s.t.} \\ &\textstyle\int_0^1 \int_{\psi(\theta_1)}^1 \pi(\theta_1, \theta_2) d\theta_2 d\theta_1 = \frac{1}{2}. \end{aligned} \tag{25}$$

In order to calculate the optimal decision function, we define

$$F_1(\theta_1, w) = \int_w^1 \pi(\theta_1, \theta_2) d\theta_2, \tag{26}$$

$$F_2(\theta_2, w) = \int_w^1 \pi(\theta_1, \theta_2) d\theta_1. \tag{27}$$

Inserting these expressions into the integral 25, we obtain

$$\begin{aligned} &\underset{\psi}{min}\ (\textstyle\int_0^1 \theta_1\ F_1(\theta_1, \psi(\theta_1)) d\theta_1 + \int_0^1 \theta_2\ F_2(\theta_2, \psi^{-1}(\theta_2)) d\theta_2), \\ &\text{s.t.} \\ &\textstyle\int_0^1\ F_1(\theta_1, \psi(\theta_1)) d\theta_1 = \frac{1}{2}. \end{aligned} \tag{28}$$

Note that we are considering here the particular case of one independent variable (θ_1), one function $\psi(\theta_1)$, and an integrand that depends at most on the first derivative of the function. Using a Lagrange multiplier $\lambda(\theta_1)$, this expression defines a functional that depends on ψ. The Lagrange multipliers are, in general, functions of the independent variable. However, as it can be easily seen from above equation, when the integrand and the constraint are independent of θ_1 themselves, then each Lagrange multiplier is a constant (denoted by λ).

$$\int_0^1 \theta_1\ F_1(\theta_1, \psi(\theta_1)) d\theta_1 + \int_0^1 \theta_2\ F_2(\theta_2, \psi^{-1}(\theta_2)) d\theta_2 + \lambda \int_0^1\ F_1(\theta_1, \psi(\theta_1)) d\theta_1 \tag{29}$$

Thus, (29) is equivalent to

$$\begin{aligned} &\textstyle\int_0^1 \theta_1\ I(\theta_1, \psi, \psi') d\theta_1, \\ &\text{where} \\ &I(\theta_1, \psi, \psi') = (\theta_1 + \lambda) F_1(\theta_1, \psi) + \psi F_2(\psi, \theta_1) \psi'. \end{aligned} \tag{30}$$

The usual variational procedure with respect to the function $\psi(\theta_1)$ is to use the Euler-Lagrange equation

$$\partial_\psi I(\theta_1, \psi, \psi') - \frac{\mathrm{d}}{\mathrm{d}\theta_1} \partial_{\psi'} I(\theta_1, \psi, \psi') = 0. \tag{31}$$

That leads to the following Euler-Lagrange equation

$$\partial_\psi I(\theta_1, \psi, \psi^{'}) = (\theta_1 + \lambda)\partial_\psi F_1(\theta_1, \psi) + \psi^{'} F_2(\psi, \theta_1) + \psi^{'} \psi \partial_\psi F_2(\psi, \theta_1) \tag{32}$$

$$\partial_{\psi^{'}} I(\theta_1, \psi, \psi^{'}) = \psi F_2(\psi, \theta_1), \tag{33}$$

$$\frac{\mathrm{d}}{\mathrm{d}\theta_1}\partial_{\psi^{'}} I(\theta_1, \psi, \psi^{'}) = \frac{\mathrm{d}}{\mathrm{d}\theta_1}\psi F_2(\psi, \theta_1) = \psi^{'} F_2(\psi, \theta_1) + \psi \partial_\psi F_1(\theta_1, \psi) + \psi^{'} \psi \partial_\psi F_2(\psi, \theta_1), \tag{34}$$

And,

$$(\theta_1 + \lambda)\partial_\psi F_1(\theta_1, \psi) + \psi^{'} F_2(\psi, \theta_1) + \tag{35}$$

$$\psi^{'} \psi \partial_\psi F_2(\psi, \theta_1) - \psi^{'} F_2(\psi, \theta_1) - \psi \partial_{\theta_1} F_2(\psi, \theta_1) - \psi^{'} \psi \partial_\psi F_2(\psi, \theta_1) = . \tag{36}$$

Solving,

$$(\theta_1 + \lambda)\partial_\psi F_1(\theta_1, \psi) = \psi \partial_{\theta_1} F_2(\psi, \theta_1). \tag{37}$$

Our next step will be trying to simplify this expression. Using the Leibniz integral rule we have:

$$\partial_\psi F_1(\theta_1, \psi) = \frac{\partial}{\partial \psi} \int_{\theta_1}^{1} \pi(\theta_1, \theta_2)\, d\theta_1 = \tag{38}$$

$$\int_{\theta_1}^{1} \frac{\partial}{\partial \psi} \pi(\theta_1, \theta_2)\, d\theta_1 + \pi(\theta_1, 1)\frac{\partial}{\partial \psi} 1 - \pi(\theta_1, \psi(\theta_1))\frac{\partial}{\partial \psi}\psi(\theta_1) = -\pi(\theta_1, \psi(\theta_1)), \tag{39}$$

and

$$\partial_{\theta_1} F_2(\psi, \theta_1) = \frac{\partial}{\partial \theta_1} \int_{\theta_1}^{1} f(x, y)\, d\theta_1 = \tag{40}$$

$$\int_{\theta_1}^{1} \frac{\partial}{\partial \theta_1} \pi(\theta_1, \theta_2)\, d\theta_1 + \pi(1, \theta_2)\frac{\partial}{\partial \theta_1} 1 - \pi(\theta_1, \theta_2)\frac{\partial}{\partial \theta_1}\theta_1 = -\pi(\theta_1, \psi(\theta_1)). \tag{41}$$

And then, (37) reduces to:

$$(x + \lambda) \cdot (-\pi(\theta_1, \psi(\theta_1))) = \psi \cdot (-\pi(\theta_1, \psi(\theta_1))). \tag{42}$$

Solving $\psi(x)$, we finally obtain the solution as:

$$\psi(\theta_1) = \theta_1 + \lambda. \tag{43}$$

Finally, note that $\arg\min_{i \in N}(\psi_i(\theta_i)) = \arg\min_{1,2}(\theta_1 + \lambda, \theta_2)$, given that,

$$\arg\min_{1,2}(\psi_1(\theta_1), \psi_2(\theta_2)) = \arg\min_{1,2}(\theta_1 + \lambda_1, \theta_2 + \lambda_2) = \tag{44}$$

$$\arg\min_{1,2}(\theta_1 + \lambda_1 - \lambda_2, \theta_2) = \arg\min_{1,2}(\theta_1 + \lambda, \theta_2). \tag{45}$$

Which completes the proof for the case of minimization. The proof for maximization is essentially identical. Observe that both cases lead to the same value of λ. □