Novel notions of fairness and resource allocation for congested networked systems

Francesca Fossati

THÈSE DE DOCTORAT DE
LA SORBONNE UNIVERSITÉ

Spécialité

Informatique

École doctorale Informatique, Télécommunications et Électronique de Paris

Présentée par

Francesca FOSSATI

Pour obtenir le grade de

DOCTEUR de la SORBONNE UNIVERSITE

Sujet de la thèse :

Novel notions of fairness and resource allocation for congested networked systems

soutenue le 29/11/2019

devant le jury composé de :

M. David COUDERT	Rapporteur	Directeur de recherche - INRIA
M. Joaquin SANCHEZ-SORIANO	Rapporteur	Professeur - UMH
Mme. Nancy PERROT	Examinateur	Chercheur, PhD - Orange Labs
M. Andre-Luc BEYLOT	Examinateur	Professeur - ENSEEIHT
M. Deepankar MEDHI	Examinateur	Professeur - NSF, UMKC
M. Patrice PERNY	Examinateur	Professeur - Sorbonne Université
M. Stefano MORETTI	Co-encadrant	Chargé de recherche - Paris Dauphine
M. Stefano SECCI	Directeur de thèse	Professeur - CNAM

Invités :

M. Stéphane ROVEDAKIS	invité	Maître de conference - CNAM
M. Jeremie LEGUAY	invité	Chercheur, PhD - Huawei

THÈSE DE DOCTORAT DE
LA SORBONNE UNIVERSITÉ

Spécialité

Informatique

École doctorale Informatique, Télécommunications et Électronique de Paris

Présentée par

Francesca FOSSATI

Pour obtenir le grade de

DOCTEUR de la SORBONNE UNIVERSITE

Sujet de la thèse :

Nouvelles notions d'équité et d'allocation des ressources pour les systèmes congestionnés

soutenue le 29/11/2019

devant le jury composé de :

M. David COUDERT	Rapporteur	Directeur de recherche - INRIA
M. Joaquin SANCHEZ-SORIANO	Rapporteur	Professeur - UMH
Mme. Nancy PERROT	Examinateur	Chercheur, PhD - Orange Labs
M. Andre-Luc BEYLOT	Examinateur	Professeur - ENSEEIHT
M. Deepankar MEDHI	Examinateur	Professeur - NSF, UMKC
M. Patrice PERNY	Examinateur	Professeur - Sorbonne Université
M. Stefano MORETTI	Co-encadrant	Chargé de recherche - Paris Dauphine
M. Stefano SECCI	Directeur de thèse	Professeur - CNAM

Invités :

M. Stéphane ROVEDAKIS	invité	Maître de conference - CNAM
M. Jeremie LEGUAY	invité	Chercheur, PhD - Huawei

Acknowledgment

Firstly, I would like to express my sincere gratitude to my advisors Prof. Stefano Secci and Mr. Stefano Moretti, PhD, for the continuous support during the PhD study and the related research, for their patience, motivation, and knowledge.

I would like to thank the experts who were involved in the works done during these years, for their inputs and feedbacks. In particular, I would like to thank: Prof. Deep Medhi, for welcoming me to Kansas City and giving me valuable advice; Prof. Patrice Perny, Prof. Stéphane Rovedakis, Prof. Sahar Hoteit and Prof. Catherine Rosenberg, for the valuable suggestions that have increased the quality of my work.

I must express my very profound gratitude to my parents, to my sister, to Onur and to my relatives (in particular, my uncles and my aunt) for providing me support and continuous encouragement throughout the last three years and through the process of researching and writing this thesis. I would like to dedicate this thesis to my two grandmothers and thank them for everything they taught me and for giving me a great example of life. To all of you "grazie".

I would like to add to my thanks my old friends who, despite the physical distance, are always present with their support and their advice.

And finally, last but by no means least, also I would thank my colleagues; it was great sharing laboratory with all of you during these years.

Thanks for all your encouragement!

Abstract

Fairness is a topic that emerges in many fields and that is linked to resource allocation and fair division problems. In networking and computing the legacy approach to solve these situations is to model them as a single-decision maker problem, using classical resource allocation protocols as the proportional rule or the max-min fair allocation. The evolution of telecommunication network technologies, the advances in computing power and in software design practices allow giving a high degree of freedom and programmability to resource allocation and routing decision-making logics. Furthermore, software-defined radio and virtualized network platforms can be used on top of a shared infrastructure making possible a real-time auditability of the system by its tenants and users. Therefore, novel networking contexts such that tenants can be aware of other users' demands and the available amount of the resource, or they can have a partial information on the system, have to be considered. Moreover, in the decision-making modeling for 5G systems, it is necessary to move from single-resource allocation to multi-resource allocation. In fact, with the introduction of network slicing, we need logically-isolated network partitions that combine network, computation and storage programmable resources. In this thesis we aim to provide a theoretical and formal analysis and redefinition of fairness of resource allocation for congested networked systems, i.e., systems that are in the challenging situation in which resources are limited and not enough to fully satisfy users' demand. We analyze, propose and evaluate numerically centralized, decentralized, single and multi-resource allocation rules.

Résumé

L'équité est un concept qui émerge dans de nombreux domaines et qui est lié à l'allocation des ressources. L'approche classique pour résoudre ces situations utilise les protocoles d'allocation connues comme la règle proportionnelle ou l'allocation max-min fair. L'évolution des technologies des réseaux de télécommunications et les progrès de la puissance de calcul et de la conception des logiciels accroît la liberté et la programmabilité de l'allocation des ressources et des logiques décisionnelles de gestion du trafic. De plus, des plates-formes radio et réseau virtualisées définies par logiciel sont utilisées en complément d'une infrastructure partagée permettant une contrôlabilité en temps réel du système par ses utilisateurs. Par conséquent, des nouveaux contextes qui permettent aux utilisateurs d'être au courant des demandes des autres utilisateurs et de la quantité disponible de ressource, ou qui leur permettent d'avoir une information partielle sur le système, sont à considérer. Parallèlement il est nécessaire de passer d'une allocation mono-ressource à une allocation multi-ressource pour tenir en compte de façon adéquate le modèle complet. En fait, avec l'introduction du concept du network slicing, il faut considérer des partitions du réseau logiquement isolées qui combinent des ressources réseau, de calcul et de stockage programmables.

Dans cette thèse, nous élaborons une analyse théorique et formelle et une redéfinition de l'équité de l'allocation des ressources pour un réseau congestionné, c'est-à-dire quand les ressources sont limitées et insuffisantes pour satisfaire la demande des utilisateurs. Nous analysons, proposons et évaluons des règles d'allocation centralisées, décentralisées, mono-ressource et multi-ressources.

Contents

List of Figures

List of Tables

1. Introduction

1.1 Background and motivations

Fairness is an important and interdisciplinary concept that emerges in many fields and that is strictly linked to resource allocation and fair division problems. There is no consensus about the meaning of the word fairness but we can state it concerns an equal treatment of the individuals and the idea of a just and impartial re-partition of goods. For example, in Oxford English Dictionary the fairness is defined as *"impartial and just treatment or behavior without favoritism or discrimination"* [1] and in Cambridge Dictionary as *"the quality of treating people equally or in a way that is right or reasonable"* [2]. Other definitions of fairness are related to the envy-freeness of the allocations [3] or to the treatment of individuals in a way that is consistent with what they deserve [4].

Fairness has been discussed and studied in many fields, from the more abstract to the more practical and technology-related ones. In philosophy and political science the fairness deals with the ethic concept of sameness (i.e., everyone is equal does not matter his need), deservedness (i.e., what one gets is consistent with what he deserves) and need (i.e., who has more should contribute with a greater percentage to help who has less). In economy fairness is related both to welfare and social politics or to understand the correct sharing of revenues obtained through investments.

In networking and computing fairness issues come up in resource allocation (in some contexts also referred to as resource scheduling, pooling, or sharing), that is a phase, in a network protocol or system management stack, when a group of individual users or clients have to receive a portion of the resource in order to provide a service. The legacy approach to solve these situations is to consider a single-decision maker problem using classical resource allocation rules as the proportional [5] or the max-min fair one [6, 7].

With the evolution of telecommunication network technologies and thanks to the advances in computing power and software design, new paradigms as the software defined radio (SDR) and software defined network (SDN) are emerging [8]. This allows an increasing degree of freedom and programmability to network and system resource

allocation and traffic management decision-making logic. Furthermore, as predicated with 5G, software-defined radio and virtualized network platforms are used on top of a shared infrastructure making possible a real-time auditability of the system [9]. Due to all these reasons novel networking contexts such that tenants can be aware of other users' demands and the available amount of the resource or they can have a partial view on the system have to be considered.

Together with new type of decision-making solution for 5G systems, it is necessary to move from single-resource allocation to multi-resource allocation to adequately take into account the composite system. In fact, if in legacy systems spectrum is allocated independently to the link bandwidth, to the availability of network function and to the processing resource, in 5G, with the introduction of the network slicing concept,there is a need to find ways to built logically-isolated network partitions that combines network, computation and storage programmable resources [10]. Both centralized and decentralized approaches need to be studied. The first one model the case in which only one provider can provide each resource necessary to serve the service, while the second one the case in which the resources are managed by different decision-maker (or platform, orchestrator, controller).

1.2 Contributions

In this thesis we aim to provide a theoretical and formal analysis of fairness and resource allocation in new technology able to capture the enhanced view users can have on the system. In particular we investigate how we should move from legacy single-resource approaches to novel multi-resource approaches in order ensure fairness in 5G environments, where resource sharing among tenants (slices) needs to be made acceptable by users and applications, which therefore need to be better informed about the system status via ad-hoc (northbound) interfaces than in legacy environments.

In the thesis we always refer to the challenging resource allocation problems in which resources are limited and not enough to fully satisfy users' demand. These are situations where there is room to talk about fairness because it is necessary to find allocations able to not strongly advantage or disadvantage a user.

Table 1.1 summarizes the contribution of each chapter. Our main research contributions are presented in Chapter 3, 4, 5, 6 and are articulated as follows.

- In Chapter 3 we argue that, under awareness about the available resource and other users demands, a cooperative setting has to be considered in order to revisit and adapt the concept of fairness. We identify in the individual satisfaction rate the key aspect of the challenge of defining a new notion of fairness in systems with complete information sharing and, consequently, a more appropriate resource allocation algorithm. We generalize the concept of user satisfaction considering the set of admissible solutions for bankruptcy games and we adapt to it the fairness indices. Accordingly, we propose a new allocation rule we call Mood Value: for each user, it equalizes our novel game-theoretic definition of user satisfaction with respect to a distribution of the resource. We test the mood value and a new fairness index through extensive simulations about the cellular frequency scheduling use-case, showing how they better support the fairness analysis. We complete the chapter with further analysis on the behavior of the mood value in the presence of multiple competing providers and with cheating users.

- In Chapter 4 we analyze inaccurate information sharing situations, i.e., such that users can be aware, up to a small error, about the other users' demands and the available global resource. Consequently, given an allocation rule, users can predict an allocation that will not necessarily coincide with the actual one. We provide an estimation of the error for the proportional allocation, the Max-Min Fair allocation and the Mood value, both in case there is an error on the available resource or on the demands vector. Fairness considerations shows the superiority of the Mood value compared to the classical solutions in case of inaccurate information sharing context.
- In Chapter 5 we move from single-resource allocation rules analyzed in Chapter 3 and 4 to multi-resource allocation frameworks. If in legacy networks, resources such as link bandwidth, spectrum, computing capacity are allocated independently of each other, in 5G environments, the concept of network slicing is introduced. This implies that resource allocation problem deals with more than one resource. We address the problem of fairly sharing multiple resources between slices, in the critical situation in which the network does not have enough resources to fully satisfy slice demands. We model the problem as a multi-resource allocation problem, proposing a versatile optimization framework based on the Ordered Weighted Average (OWA) operator, that takes into account different fairness approaches. We show how, adapting the OWA utility function, our framework can generalize classical single-resource allocation methods, existing multi-resource allocation solutions at the state of the art, and implement novel multi-resource allocation solutions. We compare analytically and by extensive simulations the different methods in terms of fairness and system efficiency. We then take into account that a slice needs to fulfill a Service Level Agreement (SLA), that is a contract between the slice provider and the tenants on the quality of service and reliability, expressed for a diverse set of physical resources (spectrum, link capacity, computing power, etc). We provide two scheduling algorithms that take into account SLA requirements in terms of minimum and nominal resource quantity demands. We show that the algorithm that considers the availability rate of the service, in addition to providing the minimum capacity, has better performances in terms of time-fairness. For both scheduling algorithms we consider a user delaying policy able to take into account SLA priority and latency requirements.
- In Chapter 6 we again address the network slicing resource allocation problem. In the previous chapter, a centralized slice orchestration approach has been proposed, where a multi-domain orchestrator (called also network slice provider) allocates the resources, using a multi-resource allocation rule. Nonetheless, while simplifying the algorithmic approach, centralization can come at the expense of scalability and performance and generally each computing resources (CPU, RAM, storage), is managed by a distinct decision-maker, platform, provider, orchestrator or controller. In this chapter, we propose new ways to decentralize the slice resource allocation problem, using cascade or parallel resource allocations. We provide an exhaustive analysis of the advantages and disadvantages of the different approaches together with a numerical analysis in a realistic environment.

	Contributions
Chapter 2	- To provide an overview on fair resource allocation in networking - To formalize the resource allocation problem as a bankruptcy game and provide an overview on game solutions - To provide an overview on fairness measures - To provide an overview on fair multi-resource allocation rules
Chapter 3	- To propose a new measure of users satisfaction in complete information context - To propose a new allocation rule: the mood value - To provide an analysis of the properties of the allocation - To propose a new index of fairness - To test the new allocation and index of fairness in 2 use-cases - To provide an analysis of the allocation in dynamic context with multiple provider
Chapter 4	- To evaluate the allocated error in case of error on the available resource - To provide fairness considerations in case of error on the available resource - To evaluate the allocated error in case of error on the demand vector - To provide fairness considerations in case of error on the demand vector
Chapter 5	- To provide an overview on the network slicing - To provide an overview on the ordered weighted average (OWA) operators and the justification of their use in multi-resource allocation context - To propose a general framework for multi-resource allocation in network slicing - To provide an analysis of the properties of the proposed allocations - To propose an adaptation of the framework to consider Service Level Agreement-driven constraints - To test the proposed framework in realistic scenario and provide an evaluation of the Service Level Agreement-driven algorithms
Chapter 6	- To propose two cascading and two parallel approaches to decentralize the 5G slice orchestration logic for multi-resource allocation - To provide the analysis of the budget delay of each algorithm, including the centralized ones - To provide the analysis of the pros and cons of each algorithm, including the centralized ones - To provide a numerical analysis of the algorithms
Chapter 7	- To provide a conclusion and a summary of the work - To provide open questions and future work directions

Table 1.1: Summary of contributions

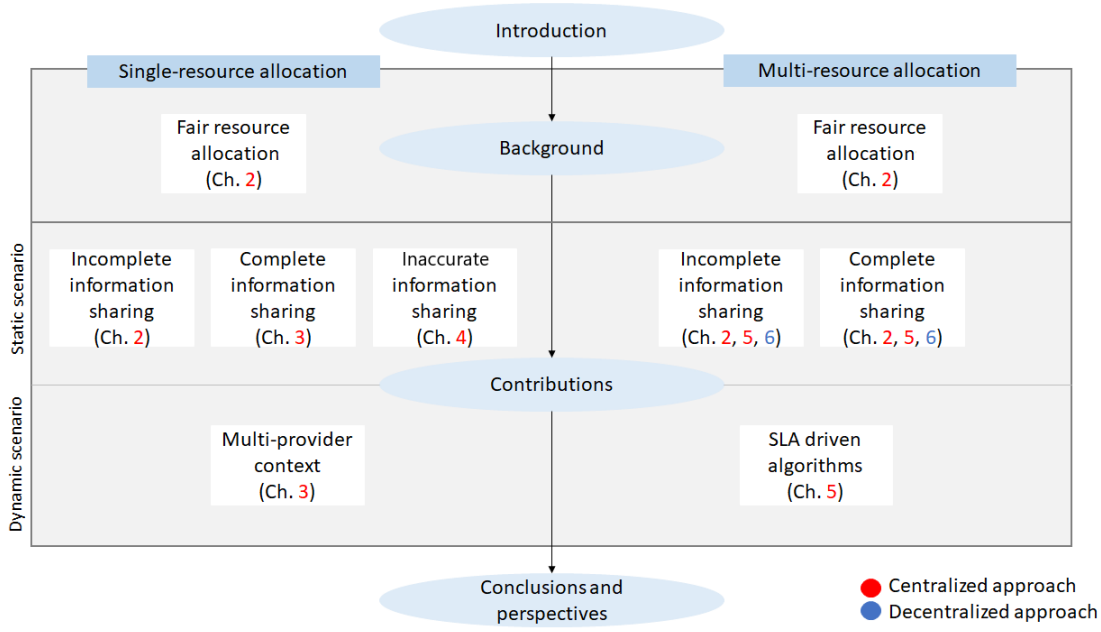


Figure 1.1: Thesis organization

1.3 Thesis organization

Figure 1.1 shows the map of the thesis and the position of the contribution of each chapter with respect to the type of resource (single or multi), the type of scenario (static or dynamic) and the type of approach (centralized or decentralized). In particular, if users demand only one resource we are in the single-resource allocation case, if more than one we are in the multi-resource allocation case; if we consider allocations in a given instant of time we talk of static scenario, while if we consider a window of time in which users submit more than one demand, or, maintaining the same demand, can move from one operator to another, we talk of dynamic scenario, also sometimes referred with the term scheduling; if only one provider manages the required demand/s we use centralized approaches to share resource/es, if the resources are managed by different providers we use decentralized approaches.

1.4 List of publications

Journals

Fossati F., Hoteit S., Moretti S., Secci S. *Fair Resource Allocation in Systems With Complete Information Sharing.* IEEE/ACM Transactions on Networking 26.6, 2801-2814, 2018.

Fossati F., Medhi D., Moretti S., and Secci S. *Error Estimate and Fairness in Resource Allocation with Inaccurate Information Sharing.* IEEE Networking Letters 1.4, 173-177, 2019.

Fossati F., Moretti S., Perny P., Secci S. *Multi-Resource Allocation for Network Slicing.* Submitted.

Conferences

Fossati F., Moretti S., and Secci S. *A mood value for fair resource allocations.* IFIP Networking Conference and Workshops, 2017.

Fossati F., Moretti S., and Secci S. *Multi-Resource Allocation for Network Slicing under Service Level Agreements.* IEEE 10th International conference on the Network of the Future (NoF' 19), 2019.

Fossati F., Moretti S., Rovedakis S. and Secci S. *Decentralization of 5G slice resource allocation.* IEEE/IFIP Network Operations and Management Symposium (NOMS 2020)

2. Background on fair resource allocation

In this chapter we provide an overview on how resources should be allocated in order to satisfy a fairness criterion and how to measure the fairness of an allocation.

Generally in computers networks with the term *resource allocation* we refer to the allocation of different flow in the network. In this work we refer to the sharing of resources in order to provide a service. This implies that each users has a demand with respect to the resource that is tailored to his need. For example we can imagine that a user need a web service that provides computing capacity in the cloud as the one provided by Amazon [11]. Depending on the type of job he has to run, he can ask for different types of service with heterogeneous values of memory, vCPU and so on.

In case of single-resource context a resource allocation problem can be defined as follows.

Definition 2.0.1 — Resource allocation problem. A *resource allocation problem* is characterized by a pair (d,R), in which $d \in \mathbb{R}^n$ is the vector of demands (claims) from n users (claimants) and $R \in \mathbb{R}$ is the resource (estate) that should be shared between them. The set of users is N=$\{1,...,n\}$.

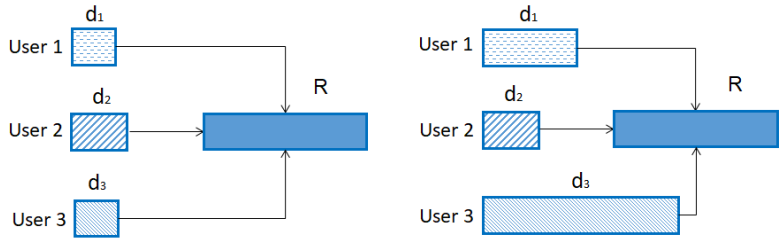
An allocation is a solution of the problem and can be defined as follows.

Definition 2.0.2 — Allocation, Allocation rule. *An allocation* $a \in \mathbb{R}^n$ is a solution vector that satisfies three basic properties:

- *Non-negativity*: each user should receive at least zero.
- *Demands boundedness*: each user cannot receive more than its demand.
- *Efficiency*: the sum of all allocations should be R.

An allocation rule is a function that associates a unique allocation vector a to each (d,R).

As already explained we analyze the challenging problem in which R is not enough to satisfy all the demands, i.e., $\sum_{i=1}^{n} d_i \geq R$ (Fig. 5.15b). In fact in the case in which the resource



(a) $\sum_{i=1}^{n} d_i \leq R$ (b) $\sum_{i=1}^{n} d_i \geq R$

Figure 2.1: Examples of resource allocation problems

is enough to cover the users demands (Fig. 5.15a) the allocation for each user coincides with the demands itself.

Classically resource allocation problems are formulated as convex optimization problem aimed to maximize the aggregate utility [5]. Called $\mathscr{U}_i(a_i)$ the utility function of user $i \in N$ with an allocation equal to a_i, the objective is to maximize $\sum_{i=1}^{n} \mathscr{U}_i(a_i)$ under the capacity constraints. So the problem can be formulated as follows:

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^{n} \mathscr{U}_i(a_i) \\ \text{subject to} \quad & \sum_{i=1}^{n} a_i \leq R \\ & 0 \leq a_i \leq d_i, \forall i \in N \end{aligned} \tag{2.1}$$

The utility function needs to capture the individual's evaluation of the worth of the good that the user requests. It is assumed to be a smooth (i.e, with derivatives of all orders everywhere in its domain) concave function. The concavity is used referring to the economical "law of diminishing returns". It affirms that if one factor of production increases, while the others remain constant, the marginal benefit declines [12]. An example of utility function is depicted in Figure 2.2, where we can see that increasing of the same amount the allocation of an user i ($\Delta a_i = a_i^2 - a_i^1 = a_i^4 - a_i^3$) the marginal contribution of the utility diminishing increasing the value of the allocation (i.e.,$\mathscr{U}_i(a_i^2) - \mathscr{U}_i(a_i^1) >$

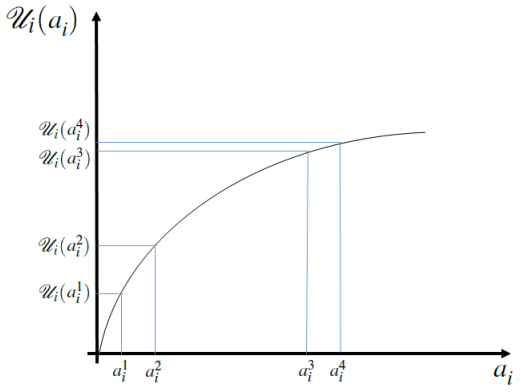


Figure 2.2: Example of utility function

$\mathscr{U}_i(a_i^4) - \mathscr{U}_i(a_i^3)$).

From the optimization theory we know that the maximization of a concave function over a convex set is a convex optimization problem and it produces an unique solution [13]. Being the set produced by the constraints in (2.1) convex, it follows that our problem has a unique solution.

In the following section we show the most famous allocation rules used in networking and computing resource allocation problem and the fairness criterion behind them.

2.1 Fair single resource allocation in networking and computing

The most well-known allocation rules are the proportional and the Max-Min Fair (MMF) rule. In general, we can consider a class of utility function called α-fair utility function that for specific values of the parameter α provides the two mentioned allocation rules.

2.1.1 The proportional and weighted-proportional allocation rule

The proportional allocation rules is obtained when when we consider the problem (2.1) and the evaluation of the worth of the allocation is expressed by the logarithmic function, i.e. $\mathscr{U}_i(a_i) = \log a_i$. Assigning a weight to the users functions (i.e., $\mathscr{U}_i(a_i) = w_i \log a_i$) we obtain the so called weighted proportional allocation. So we can consider the proportional allocation as a specific case of weighted proportional allocation when, for each user, the weight is equal to 1[1].

Being the logarithmic function a concave function as we already explain the solution of the problem is unique (2.1) and it can be found using the Karush–Kuhn–Tucker (KKT) conditions that are the generalization of the method of Lagrange multipliers which allows inequality constraints [14, 15].

The fairness properties that characterize the proportional allocation is stated in the following proposition [5, 16].

Proposition 2.1.1 Let a^p be the allocation vector obtained with the logarithmic utility function, then for any other allocation vector a it holds:

$$\sum_{i=1}^{n} \frac{a_i - a_i^p}{a_i^p} \leq 0. \tag{2.2}$$

Equation (2.2) shows that the sum of the proportional variation in each users' rate is non-positive. This means that if the allocation of an user A is increased it exists at least another user B whose allocation decreases and the loss of B in proportion is larger than the gain of A. For this reason the allocation is called proportional fair.

Similarly when we consider general weights w_i to the users utility functions we obtain an allocation such that:

$$\sum_{i=1}^{n} w_i \frac{a_i - a_i^{wp}}{a_i^{wp}} \leq 0. \tag{2.3}$$

where a^{wp} is the allocation vector solution and a is any other allocation vector.

As already explained generally the resource allocation refers to problems in which users has no demands and the constraints of the problem are given from the link capacity,

[1] Pay attention to the fact that the proportional allocation described here is not the allocation that assigns the same portion of demand to each user.

instead in our case users has a demands linked to the service they need. When we choose the weights w_i in the utility function equal to the demand d_i for each user i it holds the following theorem.

Theorem 2.1.2 The allocation that is solution of the optimization problem (2.1) with utility functions of type $\mathscr{U}_i(a_i) = d_i \log a_i$ is such that each users receives the same portion of the demand d_i.

Proof. We need to prove that the solution is of type $a_i = \frac{R}{\sum_{j=1}^{n} d_j} d_i$. The lagrangian of the problem is:

$$L(a,\mu,\lambda) = \sum_{i=1}^{n} d_i' \log a_i - \mu^T (D - Aa) - \lambda (R' - \sum_{i=1}^{n} a_i)$$

where the vector μ and λ are the lagrangian multipliers, D is the vector of the demands and A is the identity matrix of dimension n. The optimal point coincides with the stationary point of the Lagrangian function; so setting $\frac{\partial L}{\partial a_i} = 0$ we get $a_i = \frac{d_i}{\mu_i + \lambda}$. Using the KKT conditions we obtain the optimal solution when we choose $\mu^T = 0$ and $\lambda \neq 0$. In fact in this case we have $\sum_{i=1}^{n} \frac{d_i}{\lambda} = \frac{1}{\lambda} \sum_{i=1}^{n} d_i = R$. It follows that $\lambda = \frac{1}{R} \sum_{i=1}^{n} d_i$ is greater or equal to 1 and $a_i = \frac{d_i}{\lambda}$ is less or equal to d_i, that is an admissible solution.

■

The theorem explain another fairness concept characterizing the weighted proportional allocation with weights equal to the demand: each user has the same satisfaction of the other users because the same percentage of the demands is allocated to everyone.

We conclude the section with an example of proportional and weighted proportional solution.

■ **Example 2.1** Let (d,R) be the situation of Fig. 5.15b with d=$(3,2,13)$ and R=10. Table 2.1 shows the proportional allocation a^p and the weighted proportional allocation a^{wp} when the weights coincides with the users demands.

User demands	a^{wp}	a^p
3	1.67	3
2	1.11	2
13	7.22	5

Table 2.1: Allocation rules comparison - Proportional vs weighted proportional rule

We can notice that, when weights coincides with demands, users receive the same percentage of resource:

$$\frac{a_1}{d_1} = \frac{1.67}{3} \simeq 0.556, \quad \frac{a_2}{d_2} = \frac{1.11}{2} \simeq 0.556, \quad \frac{a_3}{d_3} = \frac{7.22}{13} \simeq 0.556$$

and this percentage is equal to $\frac{R}{d_1+d_2+d_3} = \frac{10}{18} \simeq 0.556$.

We can also check the conditions (2.2), (2.3) characterizing the two allocations when we consider another generic allocation $a = (2,2,6)$. An example is given in the Table 2.1 where we can see that the sum of the proportional variation of all user is negative.

■

User demands	a^p	a	$\frac{a_i - a_i^p}{a_i^p}$
3	3	2	-1/6
2	2	2	0
13	5	6	1/5
$\sum_{i=1}^{n} \frac{a_i - a_i^p}{a_i^p}$			-2/5

User demands	a^p	a	$w_i \frac{a_i - a_i^p}{a_i^p}$
3	1.67	2	0.59
2	1.11	2	1.6
13	7.22	6	-2.2
$w_i \sum_{i=1}^{n} \frac{a_i - a_i^p}{a_i^p}$			-0.01

Table 2.2: Check of conditions (2.2), (2.3)

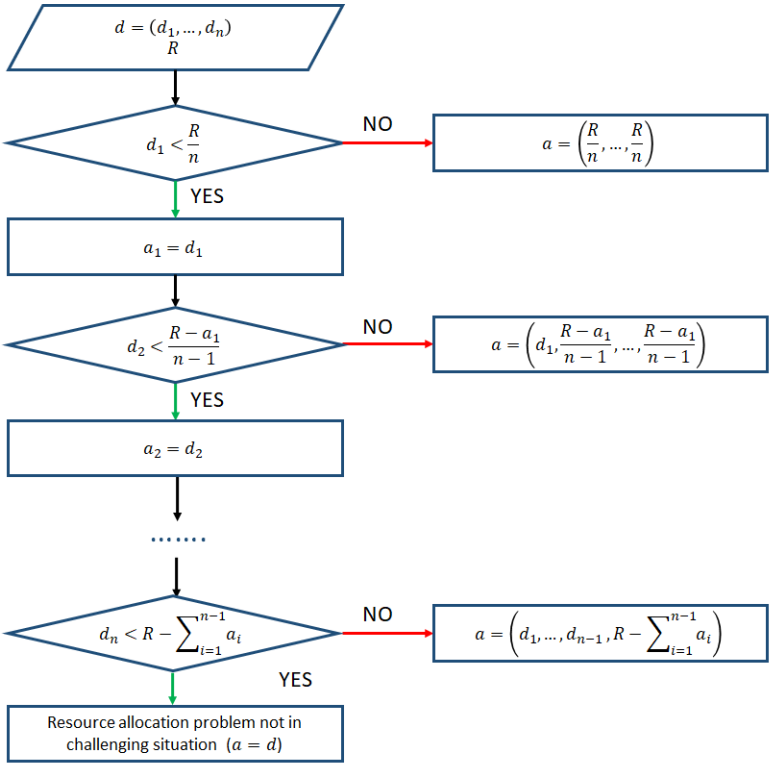


Figure 2.3: MMF allocation given the ordered demand vector d and the available resource R

2.1.2 The Max-Min Fair (MMF) allocation rule

Another well-known rule is the MMF allocation rule. It is based on the egalitarian notion of fairness, described in political philosophy by Rawls [17]. The egalitarian theory of justice he develops refers to social justice in which persons collaborate and are in harmony with the institutions that assign rights and benefits. The MMF allocation protects weaker users and can be calculated as follows: if we order the claimants according to their increasing demand, i.e., $d_1 \leq d_2 \leq \cdots \leq d_n$, then MMF allocation for user i is given by:

$$a_i^{MMF} = \min\left(d_i, \frac{R - \sum_{j=1}^{i-1} a_j^{MMF}}{n - i + 1}\right). \tag{2.4}$$

Figure 2.3 describes how we can calculate the MMF allocation through the formula (2.4).

The MMF allocation is characterized by the following property [6, 16].

Proposition 2.1.3 The MMF allocation a^{MMF} is such that for any other allocation a satisfying the capacity constraints the following is true: if $a_s^{MMF} < a_s$ for some $s \in N$ then there

exists at least an user $l \in N$ such that $a_l^{MMF} \leq a_s^{MMF}$ and $a_l < a_l^{MMF}$.

The following example shows how we can calculate recursively the MMF allocation using the formula and the check of the property described in proposition when we consider another generic allocation.

■ **Example 2.2** Let (d,R) the same resource allocation problem of Example 2.1. In order to calculate the MMF allocation we firstly order the users demands $d = (2,3,13)$. Following the flowchart in Figure 2.3 we have:

- $d_1 < \frac{10}{3} \rightarrow a_1^{MMF} = 2$
- $d_2 < \frac{8}{2} \rightarrow a_2^{MMF} = 3$
- $d_3 \geq 5 \rightarrow a_3^{MMF} = 5$

The allocation MMF is $a^{MMF} = (2,3,5)$. If we choose another allocation vector as $a = (2,2,6)$ we can notice that the property of proposition 2.1.3 holds because if we increase an allocation of a user (in our case the third one) we are obliged to decrease the allocation of another user (in our case the second one) that has a MMF allocation smaller ($a_2^{MMF} < a_3^{MMF}$). ■

We can notice that in example 2.2 the value of the MMF allocation coincides with the proportional allocation. This is always true in our resource allocation problem due to the following proposition.

Proposition 2.1.4 In the case of a single resource the MMF allocation and the proportional allocation coincide.

This proposition can be stated considering the following one about flow allocations on a network: in the case of a single bottleneck link the MMF allocation and the proportional allocation coincide.

2.1.3 The α-fairness allocation rule

The α-fairness allocation rule is a family of allocation where the utility function captures different fairness criteria [16, 18].

Definition 2.1.1 — α-fair utility. The α-fair utility function is:

$$\mathscr{U}_i(a_i) = w_i \frac{a_i^{1-\alpha}}{1-\alpha} \tag{2.5}$$

with $\alpha > 0$ and $\alpha \neq 1$.

Different values of α and of w_i yield different allocations included the well-known ones. Table 2.3 summarize the most famous cases [16].

Rule	Value of α	Value of w_i
Proportional	$\alpha \rightarrow 1$	1
Weighted proportional	$\alpha \rightarrow 1$	any value
MMF	$\alpha \rightarrow \infty$	1

Table 2.3: Correspondence between classical and α-fair allocations

2.2 Game theoretic rules

Given a resource allocation problem, different division rules can be proposed and they can be related to solution concepts of the theory of cooperative games [19]. If the allocation rule proposed in the previous paragraph refers to a networking context, here the division rules are studied in mathematical social science and refer to the so called "claim problem" where it is necessary to find well-behaved rules to associate to each claimant a part of the available resource. [19] provides a survey on the resource allocation problems in case of scarcity of resource, presenting the most famous division rules and their equivalence in terms of bankruptcy game and bargaining game solutions. We here provide the description of the division rules (section 2.2.1), a background on the most famous solutions for bankruptcy games and the bargaining game (sections 2.2.2, 2.2.3) and we conclude the section relating the division rules with the game solutions (section 2.2.4).

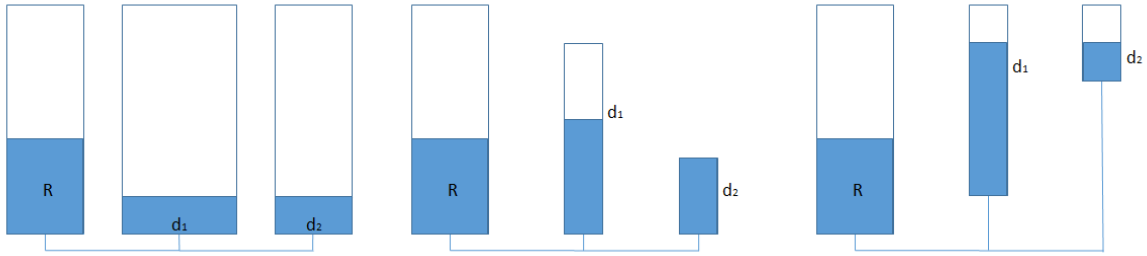
2.2.1 Division rules

The rules mostly commonly used in the claim problem are [19]:

- the *proportional (P) rule* defined as $a^P = \lambda d$ where λ is chosen so that $\sum_{i=1}^{n} \lambda d_i = R$. It is the rule that makes award proportional to users demands[2].
- the *adjusted proportional (AP) rule* that allocates the minimal right to each user and then the remainder is divided proportionally to the revised claims.
- the *constrained equal award (CEA) rule* defined as $a_i^{CEA} = \min\{\lambda, d_i\}$ where λ is chosen so that $\sum_{i=1}^{n} \min\{\lambda, d_i\} = R$.
- the *constrained equal losses (CEL) rule* defined as $a_i^{CEL} = \max\{d_i - \lambda, 0\}$ where λ is chosen so that $\sum_{i=1}^{n} \max\{d_i - \lambda, 0\} = R$.
- the *Talmud (T) rule* defined as the CEA rule, if R is not enough to satisfy the half-sum of the claims. Otherwise, each agent receives the half of his claim and the CEL rule is applied to distribute the remaining resource.
- the *random arrival (RA) rule* defined imagining claimants arriving one at time to get compensated. They are fully honored until there is room. Depending on the claimants arrival order, the allocation is given by the arithmetical average over all orders of arrival.

[20] introduces a fascinating new concept to represent the division rules using hydraulic rationing. Authors proposes a physical device wherein vessels correspond to claims and water corresponds to the available resource. Figure 2.4 shows how we can interpret the proportional, CEA and CEL rule. The proportional rule is the most intuitive one, so, if we imagine the resource R as a liquid in a tank, we have that each player is represented by a container whose section is equal to the demand d_i. All the containers have the inferior basis at the same level. For the CEA each claimant can be represented by a container with unitary section but having height equal to the demand d_i. As for the proportional rule, they have the inferior basis at the same level. For the CEL each claimant can be represented by a container with unitary section but having height equal to the demand d_i but in this case the superior basis is at the same level. Clearly the CEA allocation coincides with the MMF

[2] It is possible to make confusion between the notation used in section 2.1.1 and here; the proportional allocation here defined corresponds to the weighted proportional allocation with weights equal to the demands of section 2.1.1. From now when we talk about proportional rule we refer to the one described here.



(a) Interpretation of the proportional rule (b) Interpretation of the CEA rule (c) Interpretation of the CEL rule

Figure 2.4: Interpretation of the division rules using communicating vessels

allocation.

Each of these division rules is characterized by desirable properties [19]. Naturally, the three required properties for being an allocation (Non-negativity, Demand boundedness and Efficiency) hold for each division rules. We list here some extra properties likable for the allocation rules and in Table 2.4 we show which of them are satisfied by the division rules described above. Many other properties are described in [19] and the ones here presented are not enough to provide a characterization of the division rules. We refer to the division rules with $f(d,R)$.

- *Equal treatment of equals*
  The property states that agents with the same demand should be treated identically. Formally: $\forall i, j \in N$ such that $d_i = d_j$ it holds $f_i(d,R) = f_j(d,R)$.
- *Scale invariance*
  The property states that a rule should be invariant with respect to changements in scale and consequently it should not depend on the unit of measure. Formally: Let $\alpha > 0$, then $f(\alpha d, \alpha R) = \alpha f(d,R)$.
- *Composition*
  The property states that the division problem can be solved in two steps, splitting the estate in two part. Formally: Let $R_1 > 0$ and $R_2 > 0$ such that $R_1 + R_2 = R$, then $f(d,R) = f(d,R_1) + f(d - f(d,R_1),R_2)$.
- *Resource monotonicity*
  The property states that, if the available resource in one allocation problem is bigger than in another one, users should receive at least the same amount of resource they receive with the second one. Formally: Let $R_1 > R_2 > 0$, then $f_i(d,R_1) \geq f_i(d,R_2)$, $\forall i \in N$.
- *Consistency*
  The property is related to the stability, because it prevents subgroups of agents to renegotiate once the allocation is provided. Formally: $\forall S \subset N$ and $\forall i \in S$, it holds $f_i(d,R,N) = f_i(d_S, \sum_{i \in S} f_i(d,R,N), S)$.
- *Independence of claim truncation*
  The property states that if users demand an amount of resource superior to the available one then their claim has to be truncated. Formally: Let $d_i^T = \min\{d_i, R\}$ $\forall i \in N$, then $f_i(d,R) = f_i(d^T,R)$.

Property	*P*	*AP*	*CEA*	*CEL*	*RA*	*T*
Equal treatment of equals	✓	✓	✓	✓	✓	✓
Scale invariance	✓	✓	✓	✓	✓	✓
Composition	✓	-	✓	✓	-	-
Resource monotonicity	✓	✓	✓	✓	✓	✓
Consistency	✓	-	✓	✓	-	✓[a]
Independence of claim truncation	-	-	✓	-	✓	✓

Table 2.4: Summary of the division rules properties.

[a]It satisfy bilateral consistency that is a weaker property obtained by considering only subgroups of two remaining agents

2.2.2 The bankruptcy game

The analysed resource allocation problem, in which the resource can not cover the users demands, is known in game theory as bankruptcy game [19]. As the name suggest this game models the firm bankruptcy and the different solutions of the game represent how it is possible to divide among creditor the liquidation value of the firm. Different works in networking context models the resource allocation problem as a game [21–23].

The bankruptcy game is a coalitional (or cooperative) Transferable Utility (TU) game [24]. A cooperative game is defined as follow.

Definition 2.2.1 — Cooperative game. A cooperative game is a pair (N,v) where N=$\{1,\dots,n\}$ denotes the set of *players* and $v : 2^N \to \mathbb{R}$ is the *characteristic function* with $v(\emptyset)$=0 by convention.

In bankruptcy games [24, 25] the value of each coalition S of players is given by [3]:

$$v(S) = \max\{R - \sum_{i\in N\setminus S} d_i, 0\} \tag{2.6}$$

where $R \geq 0$ represents the estate to be divided and $d \in \mathbb{R}_+^N$ is a vector of claims satisfying the condition $\sum_{i\in N} d_i > R$ [26]. The value of each coalition can be interpret as the minimum payoff a coalition can get. In fact if the complementary coalition is fully satisfied, getting $\sum_{i\in N} d_i$, and there is still resource available ($R - \sum_{i\in N} d_i > 0$), the value of the coalition is exactly the available left resource, otherwise it is zero.

■ **Example 2.3** We consider the resource allocation problem of the Example 2.1 where d=$(3,2,13)$ and R=10. The value of each coalition S is in Table 2.5.

S	$\emptyset$	{1}	{2}	{3}	{1,2}	{1,3}	{2,3}	{1,2,3}
$v(S)$	0	0	0	5	0	8	7	10

Table 2.5: Example of bankruptcy game

■

The bankruptcy game is *superadditive*[4], that is:

$$v(S\cup T) \geq v(S) + v(T), \quad \forall S,T \subseteq N | S\cap T = \emptyset$$

[3]It exits an alternative definition of the bankruptcy game called optimistic one where $v(S) = \min\{\sum_{i\in S} d_i, R\}$.

[4]Note that the definition of superadditivity holds for general cooperative games.

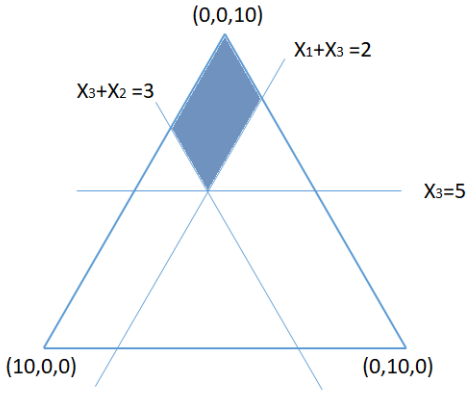


Figure 2.5: Core of the game in Table 2.5.

and *supermodular*[5] (or, equivalently, *convex*) [26], that is:

$$v(S\cup T)+v(S\cap T)\geq v(S)+v(T)\ \forall S,T\subseteq N$$

The superadditivity states that being together is better than being separated because the sum of the utility of two disjoint subset of users S and T is less or equal to the utility of the coalition formed by the sum of the two subset of user $S\cup T$.

Solutions of a cooperative game proposed in literature are several and they can be (i) a set of possible solution as the core or (ii) a one-point solution as the Shapley value, the nucleolus or the τ-value.

The classical set-value solution for a TU-game is the *core* $C(v)$, which is defined as the set of *allocation vectors* $a\in\mathbb{R}^N$ for which no coalition has an incentive to leave the grand coalition N and it is defined as follow.

Definition 2.2.2 — Core of a game. Let (N,v) be a cooperative game then the *core* of the game is [24]:

$$C(v)=\{a\in\mathbb{R}^N:\sum_{i\in N}a_i=v(N),\sum_{i\in S}a_i\geq v(S)\quad\forall S\subset N\}.$$

The core of a game can be empty but it holds that if the game is convex it has a non-empty core [27], thus the bankruptcy game has non-empty core.

■ **Example 2.4** Let us consider the game in Table 2.5. The core of the game is given by the set of solutions belonging to the blue region in Figure 2.5. ■

A *one-point solution* (or simply a *solution*) for a class $\mathscr{C}^N$ of TU games with N as set of players is a function $\psi:\mathscr{C}^N\to\mathbb{R}^N$ that assigns a payoff vector $\psi(v)\in\mathbb{R}^N$ to every TU game in the class.

A well-known solution for TU-games is the *Shapley value* [28] $\phi(v)$ of a game (N,v), defined as the weighted mean of the players' marginal contributions over all possible coalitions and computed as follows.

Definition 2.2.3 — Shapley value. Let (N,v) be a cooperative game then the *Shapley*

[5] Note that the definition of supermodularity holds for general cooperative games.

value is given by:

$$\phi_i(v) = \sum_{S \subseteq N : i \in S} w_i(S)(v(S) - v(S \setminus \{i\})),$$

with $w_i(S) = \frac{(s-1)!(n-s)!}{n!}$ where s denotes the cardinality of $S \subseteq N$.

It is alternatively defined with an axiomatic characterization: it is the only solution satisfying the four following properties.

- Efficiency: $\sum_{i=1}^{n} \phi_i(v) = v(N)$.
- Symmetry: for every A not containing i and j if $v(A \cup i) = v(A \cup j)$ then $\phi_i(v) = \phi_j(v)$.
- Null player: if $v(A) = v(A \cup i)$ for each coalition A not containing i then $\phi_i(v) = 0$.
- Additivity: for every v, w characteristic functions of two games then $\phi(v+w) = \phi(v) + \phi(w)$.

It exists alternative axiomatic characterization of the Shapley (e.g., [29, 30]).

Another well studied solution for TU-games is the *nucleolus*, based on the idea of minimizing the maximum discontent [31]. It is defined as follows:

Definition 2.2.4 — Nucleolus. Given a TU-game (N, v) and an allocation $a \in \mathbb{R}^N$, let $e(S,a) = v(S) - \sum_{i \in S} a_i$ be the *excess* of coalition S over the allocation a, and let $\leq_L$ be the *lexicographic* order on $\mathbb{R}$. Given an imputation a, $\theta(a)$ is the vector that arranges in decreasing order the excess of the 2^n-1 non-empty coalitions over the imputation a[a]. The *nucleolus* $\nu(v)$ is defined as the imputation a such that $\theta(a) \leq_L \theta(y)$ for all y imputations of the game v.

[a] An imputation is a payoff vector such that $\sum_{i \in N} a_i = v(N)$ and $a_i \geq v(\{i\})$ for each $i \in N$.

The *pre-nucleolus* is defined analogously to the nucleolus but over the set of the pre-imputation, i.e., a payoff vector such that $\sum_{i \in N} a_i = v(N)$.

As compromise between the utopia and the disagreement points, a third important solution for quasi-balanced games is the τ-*value* [32]. It is defined as follows.

Definition 2.2.5 — τ-value. Let $v : 2^N \to \mathbb{R}$ be a cooperative game then the τ-*value* is given by:

$$\tau(v) = \alpha m(v) + (1 - \alpha) M(v) \tag{2.7}$$

where $\alpha \in [0,1]$ is uniquely determined so that the solution is efficient ($\sum_{i=1}^{n} a_i = R$), $M(v)$ is the utopia payoff, and $m(v)$ is the minimum right payoff. The utopia payoff is the marginal contribution of player i to the grand coalition N that utopistically could be assigned to i. The minimum right payoff is $\max_{S: i \in S} R(S,i)$, where $R(S,i)$ is the remainder (the amount which remain for player i when coalition S forms and all the other player in S obtain their utopia payoff).

■ **Example 2.5** Let us consider the resource allocation problem of the Example 2.3 where d=$(3,2,13)$ and R=10 and the value of the game in Table 2.5.

Table 2.6 shows how we can calculate the Shapley value for the first user. Similarly we can calculate the value for the other users. The resulting solution is $(1.5, 1, 7.5)$.

The nucleolus is generally difficult to calculate but in case of bankruptcy games it can be easily calculate using the method described in [33]. In this case it coincides with the Shapley value.

To calculate the τ-value we need to calculate the minimum right payoff and the utopia

payoff. Only user 3 has a minimal right different from zero so we obtain:
user 1: $\alpha \cdot 0 + (1-\alpha) \cdot 3$,
user 2: $\alpha \cdot 0 + (1-\alpha) \cdot 2$,
user 3: $\alpha \cdot 5 + (1-\alpha) \cdot 10$.
Using the efficiency we obtain $\alpha = 0.5$ and $a = (1.5, 1, 7.5)$. ∎

2.2.3 The bargaining game

The bargaining model aims to find the "right" way to distribute an amount of good between users. Mathematically, a bargaining game is a pair (C,d) where C is a bounded, closed and convex set and it coincides with the *feasible set*, i.e., the set of utility vectors attainable by the users and n is a point of the set called *nadir* or *disagreement point* and it coincides with the users utility when there is no possibility to reach an agreement between them [34].

A solution of bargaining problem is a function that associates to each game a unique point in the feasible set. The most known solution are the Nash solution [34] and the Kalai-Smorodinsky solution [35].

The *Nash solution* is obtained maximizing the product of the users utility gains from n. The obtained solution is the only one satisfying four property: (i) *invariance with respect to admissible transformation of utility function* stating that changing the origin and the measure units on the axes the solution changes accordingly, (ii) *symmetry* stating that the solution does not distinguish two equal players and their ability to negotiate is the same (iii) *independence from irrelevant alternatives* stating that if adding alternatives to the set C does not bring the solution outside C, then the solution remains the same of when we do not include the alternatives and (iv) *efficiency* that in bankruptcy situation is $\sum_{i=1}^{n} a_i(v) = R$.

The *Kalai-Smorodinsky solution* is obtained taking the segment joining the disagreement and the utopia points and by considering the unique point of the segment lying on the boundary of C. The utopia payoff U is the maximum a player can get in the bargaining process. This solution satisfies property (i), (ii), (iv) and the *monotonicity* property stating that if enlarging the set of choice for a player i the others does not change the maximal available utility, then player i should not get less than before.

■ **Example 2.6** Let us consider a resource allocation problem (d,R) with 2 players so that we can plot in a bi-dimensional space the set C and the solutions. The demand vector is $d = (4,11)$ and the resource is $R = 10$. Figure 2.6 shows the Nash (N) solution and the Kalai-Smorodinsky (KS) solution.

∎

S	$S \setminus \{i\}$	$w_i(S)$	$v(S) - v(S \setminus \{i\})$	$w_i(S) \cdot (v(S) - v(S \setminus \{i\}))$
{1}	∅	1/3	0	0
{1,2}	{2}	1/6	0	0
{1,3}	{3}	1/6	3	0.5
{1,2,3}	{2,3}	1/3	3	1
			ϕ_1	1.5

Table 2.6: Shapley value for user $i = 1$

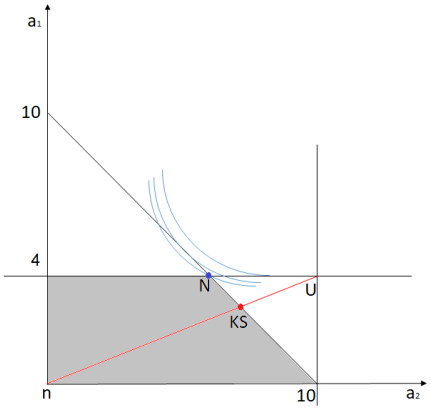


Figure 2.6: Example of Nash and Kalai-Smorodinsky solutions

2.2.4 Relating division rules and game solutions

We can state the following theorems relating the division rules described in section 2.2.1 and the game solutions described in sections 2.2.2 and 2.2.3.

Theorem 2.2.1 The following correspondences between division rules and bankruptcy game solutions hold:

- the random arrival and the Shapley value [25];
- the Talmud rule and the prenucleolus [26];
- the adjusted proportional rule and the τ-value [36].

Theorem 2.2.2 The following correspondences between division rules and bargaining solutions hold:

- the CEA and the Nash bargaining solution [37];
- the proportional rule and the weighted Nash solution with weights equal to the claims [37];
- the adjusted proportional rule and the Kalai-Smorodinsky solution [37].

2.3 Fairness measures for single-resource allocation problems

The most famous index of fairness related to single-resource allocation problems is the Jain's index defined as follows [38].

Definition 2.3.1 — Jain's fairness index. Given an allocation problem (d,R) and an allocation a, the *Jain's fairness index* is:

$$J = \left[\sum_{i=1}^{n}\left(\frac{a_i}{d_i}\right)\right]^2 \Big/ \left[n\sum_{i=1}^{n}\left(\frac{a_i}{d_i}\right)^2\right]$$

The Jain's index is bounded between $\frac{1}{n}$ and 1 [38]. The maximum fairness is measured when all the users obtain the same fraction of demand and the minimum fairness is measured when it exists only one user that receives all the resource. The Jain's index has the following good properties:

- *Population size independence*: applicable to any user set, finite or infinite.
- *Scale and metric independence*: not affected by the scale.

Index name	$f(a)$	β	q_i	$\frac{1}{\lambda}$	r	$F_{\beta,\lambda}(a)$	Value range
n*Jain [38]	$f_\beta(a)$	-1	NA	0	NA	$\left[\sum_{i=1}^n (a_i)\right]^2 / \left[\sum_{i=1}^n (a_i)^2\right]$	$[0,n]$
Max-Ratio [39]	$f_\beta(a)$	$\beta \to \infty$	NA	0	NA	$-max_i\left\{\frac{\sum_{i=1}^n a_i}{a_i}\right\}$	$(-\infty,0]$
Min-Ratio [39]	$f_\beta(a)$	$\beta \to -\infty$	NA	0	NA	$min_i\left\{\frac{\sum_{i=1}^n a_i}{a_i}\right\}$	$[0,+\infty)$
Proportional [5]	$f_\beta(a)$	$\beta \to 1$	NA	0	NA	$\sum_{i=1}^n log(a_i)$	$(0,+\infty)$
α-fair [18] ($\beta=\alpha$)	$f_\beta(a,q)$	$\beta \in (0,1)$ $\beta \in (1,\infty)$	1	$\frac{1-\beta}{\beta}$	$1-\frac{1}{\beta}$	$sign(1\text{-}\beta)\left[\sum_{i=1}^n (a_i)^{1-\beta}\right]^{\frac{1}{\beta}}$	$[0,+\infty)$
Atkinson-1 [40]	$f_\beta(a,q)$	$1-\varepsilon, \varepsilon \in [0,1]$	$\frac{1}{n}$	0	-1	$-\frac{\left[\frac{1}{n}\sum_{i=1}^n (a_i)^{1-\varepsilon}\right]^{\frac{1}{1-\varepsilon}}}{\left[\sum_{i=1}^n (a_i)/n\right]}$	$[0,1]$

Table 2.7: Common fairness indices and their parameters using (2.8)-(2.10) - NA = Not Available.

- *Boundedness*: can be expressed as a percentage.
- *Continuity*: able to capture any change in the allocation.

It is worth mentioning that this index as the ones presented hereafter are used in the context of resource allocation frameworks where the satisfaction rate of the users is not boolean (either satisfied or unsatisfied) and there are no strict Service Level Agreements to be fully satisfied.

In general it is possible to consider the following family of fairness measures proposed in [39]. In the work it is shown that it exists an unique family of fairness measures given by:

$$F_{\beta,\lambda}(a) = f(a)\left(\sum_i a_i\right)^{\frac{1}{\lambda}} \tag{2.8}$$

where a is the allocation, $\frac{1}{\lambda}$ and β are parameters belonging to $\mathbb{R}$ and $f(a)$ is a symmetric fairness measure as $f_\beta(a)$ or an asymmetric one as $f_\beta(a,q)$:

$$f_\beta(a) = sign(1-\beta)\left[\sum_{i=1}^n \left(\frac{a_i}{\sum_j a_j}\right)^{1-\beta}\right]^{\frac{1}{\beta}} \tag{2.9}$$

$$f_\beta(a,q) = sign(-r(1+r\beta))\left[\sum_{i=1}^n q_i\left(\frac{a_i}{\sum_j a_j}\right)^{-r\beta}\right]^{\frac{1}{\beta}} \tag{2.10}$$

where q_i is user i specific weight and $r \in \mathbb{R}$ is a constant.

This family of measures unifies different fairness indices belonging to different fields as networking, economy and political philosophy. The most common fairness indices are described with the parameters of (2.8)-(2.10) in Table 2.7. We find in the table the Jain index and the objective function of the proportional allocation and of the α-fair allocation, that can be used as measure of fairness. We also find two measures called Max-Ratio and Min-Ratio, the first one measuring the fairness, in case of congestion, as the ratio of the available resource over the minimum allocation and the second one measuring the fairness, in case of congestion, as the ratio of the available resource over the higher allocation. The last measure on the table is the Atkinson fairness measure that is a measure of income inequality. Differently from the Jain index to higher values of this index corresponds lower fairness.

2.4 Fair multi-resource allocations

In the literature, the first work adopting a multi-resource allocation approach for multi-resource environments, going beyond single-resource abstraction, concerns cloud optimization in which a central scheduler has to decide the number of simultaneous tasks of multiple types to run, while ensuring fairness [7, 41, 42]. Conceptually, these models can also be applied when instead of the number of tasks to run, we have a portion of the demand that has to be satisfied for each user, i.e., in the case it exists at least one resource that is not enough to satisfy the demand.

We can model a multi-resource allocation problem as follows.

Definition 2.4.1 — Multi-resource allocation problem. Let $N = \{1,...,n\}$ be the set of tenants and let $M = \{1,...,m\}$ be the set of available resources. A *multi-resource allocation problem* can be modeled as a pair (R,D) where $R = (r_1,...,r_m)$ is a vector of positive numbers, r_j representing the amount of each available resource j in M, and $D = \begin{bmatrix} d_{11} & ... & d_{1m} \\ ... & ... & ... \\ d_{n1} & ... & d_{nm} \end{bmatrix}$ is the demand matrix with $d_{ij} \in D$ equal to the quantity of resource j demanded by tenant i in N.

The allocation, solution of the allocation problem is defined as follows.

Definition 2.4.2 — Allocation matrix. Let $x = (x_1,...,x_n)$, with $0 \leq x_i \leq 1 \ \forall i \in N$, be the vector of the percentage of resources allocated to each tenant. The allocation matrix A corresponding to x is given by $\begin{bmatrix} a_{11} & ... & a_{1m} \\ ... & ... & ... \\ a_{n1} & ... & a_{nm} \end{bmatrix} = \begin{bmatrix} d_{11} \cdot x_1 & ... & d_{1m} \cdot x_1 \\ ... & ... & ... \\ d_{n1} \cdot x_n & ... & d_{nm} \cdot x_n \end{bmatrix}$.

The allocation has to belong to the admissible region $\mathscr{F}$ s.t. $\sum_{i \in N} a_{ij} \leq r_j, \forall j \in M$. We can notice that, modeling the problem in this way, the linear resource dependency is always respected[6] .

We describe in the following some of the most well-known allocation rules of the state of the art that are largely adopted in both economical and networking literature.

Dominant Resource Fairness (DRF) rule

The DRF rule is proposed in [41] as a generalization of the MMF rule. It considers, for each user, the dominant share (i.e., for a user, the maximum among all its resource shares) and the dominant resource (i.e., the resource corresponding to the dominant share), and it equalizes user's dominant shares. The allocation produced by the DRF policy is the solution of the following problem[7]:

$$\begin{aligned} &\text{maximize} && x \\ &\text{subject to} && ds_i x_i = ds_j x_j, \ \ \forall i,j \in N \\ & && x \in \mathscr{F} \end{aligned} \tag{2.11}$$

where $ds_i = \max_j \{\frac{d_{ij}}{r_j}\}$ is the dominant share of user i.

The DRF satisfies interesting fairness properties as:

[6] We will better stress this concept in Chapter 5.

[7] To maximize a vector means to maximize each component of the vector. Due to the constraints on the available resources and the ones equalizing the resource allocated for the the dominant resource, the problem can be reduced to the maximization of one component of the vector. The maximization of the others then follows.

- *Pareto efficiency*: it should not possible to increase an allocation of a user without decreasing the one of at least another user;
- *strategy proofness*: a user should not improve its allocation lying about its resources demand;
- *envy-freeness*: a user should not prefer the allocation of another user

and in [41] it is demonstrated that it is the only allocation rule satisfying a set of desirable properties. For this reason it is presented as the fairest one.

■ **Example 2.7** Let us consider a multi-resource allocation problem with $D = \begin{bmatrix} 8 & 1 \\ 20 & 1 \end{bmatrix}$ and $R = [16, 1]$. The first resource is Gbps and the second the number of CPU. The dominant shares are: $ds_1 = \max\{\frac{8}{16}, \frac{1}{1}\} = 1$ and $ds_2 = \max\{\frac{20}{16}, \frac{1}{1}\} = \frac{5}{4}$. The DRF allocation is the solution of:

$$\begin{aligned} &\text{maximize} && x \\ &\text{subject to} && x_1 = \frac{5}{4}x_2 \\ &&& 8x_1 + 20x_2 \leq 16 \\ &&& x_1 + x_2 \leq 1 \end{aligned} \tag{2.12}$$

and the solution is $x = (0.56, 0.44)$ and $a = \begin{bmatrix} 4.48 & 0.56 \\ 8.8 & 0.44 \end{bmatrix}$. ■

Asset Fairness (AF) rule

The fairness idea behind this allocation is that equal shares of different resources are worth the same and its aim is to equalize the aggregated resource value allocated to each user [41]. It is obtained solving:

$$\begin{aligned} &\text{maximize} && x \\ &\text{subject to} && \sum_{j=1}^{m} (s_j d_{ij}) x_i = \sum_{j=1}^{m} (s_j d_{kj}) x_k, \forall i, k \in N \\ &&& x \in \mathscr{F} \end{aligned} \tag{2.13}$$

where s_j is the worth of the resource j given by $s_j = \frac{r_{max}}{r_j}$, $\forall j \in N$, with r_{max} equal to the value of the greater resource in absolute value.

■ **Example 2.8** Let us consider one more time the multi-resource allocation problem with $D = \begin{bmatrix} 8 & 1 \\ 20 & 1 \end{bmatrix}$ and $R = [16, 1]$. The value of the worth of the resources are: $s_1 = \frac{16}{16} = 1$ and $s_1 = \frac{16}{1} = 16$. The Asset fairness allocation is the solution of:

$$\begin{aligned} &\text{maximize} && x \\ &\text{subject to} && (8+16)x_1 = (20+16)x_2 \\ &&& 8x_1 + 20x_2 \leq 16 \\ &&& x_1 + x_2 \leq 1 \end{aligned} \tag{2.14}$$

and the solution is $x = (0.6, 0.4)$ and $a = \begin{bmatrix} 4.88 & 0.6 \\ 8 & 0.4 \end{bmatrix}$.

■

Nash product rule

This allocation is well known in microeconomic theory also with the name of Competitive Equilibrium from Equal Income (CEEI) [43, 44]. It coincides with the Nash bargaining solutions already described in Section 2.2.3. It is obtained solving:

$$\begin{aligned} &\text{maximize} && \prod_{i \in N} x_i \\ &\text{subject to} && x \in \mathscr{F} \end{aligned} \tag{2.15}$$

The Nash product rule does not satisfy the strategy-proof property while it satisfy the Pareto efficiency and the envy-freeness [41].

■ **Example 2.9** Let us consider one more time the multi-resource allocation problem with $D = \begin{bmatrix} 8 & 1 \\ 20 & 1 \end{bmatrix}$ and $R = [16, 1]$. The Nash product rule allocation is the solution of:

$$\begin{aligned} &\text{maximize} && x_1 x_2 \\ &\text{subject to} && 8x_1 + 20x_2 \leq 16 \\ & && x_1 + x_2 \leq 1 \end{aligned} \tag{2.16}$$

and the solution is $x = (0.5, 0.5)$ and $a = \begin{bmatrix} 4 & 0.5 \\ 10 & 0.5 \end{bmatrix}$. ■

Other allocation rules known in literature are the Bottleneck Max Fairness, able to provide a more favorable efficiency-fairness tradeoff [42] compared to the MMF and the one proposed in [45], using the "no justified complaints" as objective. For this second one each user is entitled to a fixed percentage of the resource and the proposed allocation is considered fair because every user receives his entitlement on at least one bottleneck resource. An exhaustive survey on multi-resource allocations is [46].

2.5 Congestion level and ratio of available resource

We conclude the chapter providing a formal definition of the congestion level and of the ratio of the available resources, that are two measures we use in the following to estimate the degree of congestion inside the network. Dealing with problem in which the available resource is not enough to cover the user demand we can define the *congestion level* (μ) as follows.

Definition 2.5.1 The congestion level (μ) of a resource R is defined as the ratio between the sum of the demands for the resource and the available quantity of resource, i.e., $\mu = \frac{\sum_{i=1}^{n} d_i}{R}$.

Clearly, if $\mu > 1$ the resource is congested. In a dual way we can measure the *ratio of the available resource* (ρ) as the percentage of global demand satisfied by the available resource. The formal definition is the following.

Definition 2.5.2 The ratio of the available resource (ρ) is defined as the ratio between the available quantity of resource and the sum of the demands for the resource , i.e., $\rho = \frac{R}{\sum_{i=1}^{n} d_i}$.

It is clear that the relationship between the two resources is $\mu = \frac{1}{\rho}$ and that high values of μ correspond to high congested systems with low level of ρ, i.e., where only a small percentage of demand can be satisfied by the resource.

3. Fair resource allocation in complete information context

In the networking literature, the resource allocation problem is, as shown in Chapter 2, historically solved as a single-decision maker problem in which users are possibly not aware of the other users' demands and of the total amount of available resource. In this chapter we are particularly interested in novel networking contexts such that users can be aware of other users' demands and the available amount, as depicted in Fig. 3.1. In legacy resource allocation models, users' interaction with the system only implies issuing a resource request and receiving a resource allocation, therefore with an assessment of user's satisfaction only based on this information; in systems with demand and available amount awareness, users are made more conscious about the system setting with a signaling channel from the system to the users providing information about resource availability and other users' demands. As such, rational users shall compute their satisfaction also based on the presence of other users and the system resource availability.

In fact, such networking contexts with demand and resource availability awareness are making surface in wired and wireless network environments with an increasing level of programmability, i.e., using software-defined radio and virtualized network platforms on top of a shared infrastructure, as predicated with 5G. Sharing an infrastructure logically implies regular and possibly real-time auditability of the system, to ensure that various tenants esteem that they are fairly treated by the infrastructure provider [49]. In fact, users in such scenarios can be prone to change providers if their satisfaction can improve with another provider. In existing SND/NFV systems, using north-bound Application Programming Interfaces (API) tenant applications and policy manager applications can already gather resource information and share data stores with each-other. Besides forthcoming 5G systems, methods allowing raising end-user awareness exist in current systems such as those supporting spectrum sharing; for such systems, a large number of auctions mechanisms are proposed in the literature [49–51], assuming either a signaling channel or a sensing solution allowing demand (bid) and available resource awareness.

The work presented in this chapter is partially presented in [47] and [48]

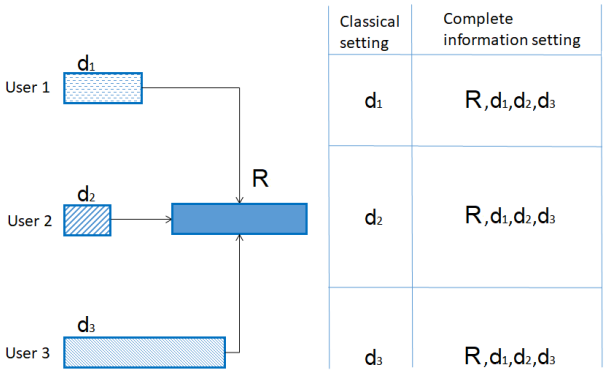


Figure 3.1: A representation of strategic network setting without and with complete information sharing. The amount d_i is the resource demand of user $i \in \{1,2,3\}$; R is the amount of shared resource available.

The main motivation is reasoning toward a new notion of user satisfaction for such resource allocation situations with demand and resource awareness. Let us briefly clarify our motivation with a basic allocation example. A user i asks a quantity of resource that is bigger than the resource itself (as user 3 in Fig. 3.1). Classical fairness indices [18], [38], [40] tend to qualify the user satisfaction as maximum when i obtains exactly what it asks. In the case where i asks more than the available amount, it cannot reach the maximum satisfaction due to the fact that its demand exceeds the available resource. Instead, under complete information sharing, it would be more reasonable that its satisfaction is maximum when it obtains all the available resource. Furthermore, if all the other users together ask a quantity of good inferior to the resource, a minimum portion of it, equal to the difference between the resource and the sum of the demands of all the others, is guaranteed to i. Under a dual reasoning, it also appears more acceptable that the minimum satisfaction of a user is reached when it receives the minimum portion of the available resource, instead of when it receives zero. If users are in complete information context the classical approach can lead to unreasonable outcomes.

In the following, modeling the resource allocation problem as a bankruptcy game, we define a new measure of users satisfaction together with a new resource allocation and a new measure of fairness.

3.1 Measurement of the users satisfaction

A natural way to quantify the satisfaction of a user, as proposed by Jain, is through the proportion of the demand that is satisfied by an allocations [38].

Definition 3.1.1 — Demand Fraction Satisfaction rate. Given the user i with demand d_i and an allocation a_i, the *Demand Fraction Satisfaction (DFS) rate* of i is:

$$DFS_i = \frac{a_i}{d_i}.$$

This rate takes a value between 0 and 1 since it represents the percentage of the demand that is satisfied.

Unavoidably, this way to quantify the user satisfaction makes the weighted proportional allocation the fairest one since it allocates proportionally to the demand. There are, however, situations in which the common sense does not suggest to allocate in a proportional way;

e.g., if there is a big gap between the demands, in order to protect the 'weaker' users and guarantee them a minimum portion of the estate. For such cases, the MMF allocation can be preferable. Furthermore, as mentioned in the introduction, the presence of other users, aware of other users' demand and of the available resource, should rationally be considered not to distort the evaluation of each user satisfaction.

For these reasons, we aim at defining an alternative satisfaction rate that satisfies the following two properties we name demand relativeness and relative null satisfaction:

- *Demand relativeness*: a user is fully satisfied when it receives its maximal right, based on the available resource;
- *Relative null satisfaction*: a user has null satisfaction when it receives exactly its minimal right, based on other users' demands and the available resource.

The minimal right for a player is the difference between the available amount and the sum of the demands of the other users (i.e., taking a worst-case assumption that the others get the totality of their demand), and the maximal right is equal to the maximum available resource, i.e., d_i if $d_i < R$, or it is equal to R otherwise. Remembering the definition of the characteristic function of a bankruptcy game ($v(S) = \max\{R - \sum_{i \in N \setminus S} d_i, 0\}$) we have that:

- the *minimal right* for player i is $v(i) = \max\{R - \sum_{j \in N \setminus i} d_j, 0\}$
- the *maximal right* for player i is $v(N) - v(N \setminus i) = \min\{R, d_i\}$

Thus we introduce the *Player Satisfaction (PS) rate*, which satisfies the above two properties by considering the value of the bankruptcy game associated to the allocation problem[1].

Definition 3.1.2 — Player Satisfaction Rate. Given a bankruptcy game such that $\sum_{i=1}^{n} d_i > E$ and an allocation a_i, the *Player Satisfaction (PS) rate* for i is:

$$PS_i = \frac{a_i - min_i}{max_i - min_i},$$

where: $min_i = v(i)$, $max_i = v(N) - v(N \setminus i)$. If $\sum_{i=1}^{n} d_i = R$ the player satisfaction rate is $PS_i = 1$, $\forall i \in N$.

The introduced satisfaction rate 'corrects' the *DFS* one since it replaces the interval of possible values $[0, d_i]$ for a_i with the interval $[min_i, max_i]$. Consequently, if for the *DFS* rate the maximum satisfaction for i is measured when it gets d_i and the minimum when it gets 0, with *PS*, i is measured to be totally satisfied when it gets max_i (i.e., d_i if available, otherwise R), and totally unsatisfied when it gets min_i (i.e., $\max\{R - \sum_{j \in N \setminus \{i\}} d_j, 0\}$).

■ **Example 3.1** Let us consider the resource allocation problem (d, R) of Fig. 5.15b with d=$(3, 2, 13)$ and E=10.

Allocation rule	user demand	user allocation	*DFS*	*PS*
Proportional allocation	13	7.22	0.555	0.444
MMF allocation	13	5	0.3846	0

Table 3.1: *DFS* and *PS* for user 3

Table 3.1 shows that in both cases player 2 is less satisfied than what expected with the *DFS* rate, when we measure its satisfaction through the PS rate. This is due to the fact that the game guarantees player 2 to get at least 5.

[1] it is possible to generalize the PS measure of fairness for all the quasi-balanced game (i.e. if $m(v) \leq M(v)$ and $\sum_{i=1}^{n} m_i(v) \leq v(N) \leq \sum_{i=1}^{n} M_i(v)$), considering as minimum the minimum right payoff $m_i(v)$ and as maximum the utopia payoff $M_i(v)$.

■

Let us show some interesting properties of the PS rate.

Theorem 3.1.1 If the allocation a belongs to the core of the bankruptcy game, $PS_i \in [0,1] \ \forall i \in N$.

Proof. If a solution a belongs to a core it holds:
$a_i \geq v(i)$ and $a_i \leq v(N) - v(N \setminus i)$.
Thus $v(i)$ and $v(N) - v(N \setminus i)$ are the minimum and the maximum value that an allocation in the core can take. If $a_i = v(i) = min_i$ then $PS_i = 0$, if $a_i = v(N) - v(N \setminus i) = max_i$ then $PS_i = 1$. ■

Proposition 3.1.2 It is possible to summarize the bankruptcy regimes of the *PS* rate in four possible cases as in Table 3.2.

	$d_i < R$		$d_i \geq R$	
	PS	*case*	*PS*	*case*
$v(i) = 0$	$\frac{a_i}{d_i}$	GM	$\frac{a_i}{R}$	GG
$v(i) \neq 0$	$\frac{a_i - v(i)}{d_i - v(i)}$	MM	$\frac{a_i - v(i)}{R - v(i)}$	MG

Table 3.2: Value of PS in the four possible cases.

Proof. Let us treat each possible cases of Table 3.2:

- *Case* GM*:* $v(i) = 0$, $d_i < R$.
  Using the definition of bankruptcy game, it holds:
  $v(N) - v(N \setminus i) = R - max\{0, R - d_i\} = R - R + d_i$.
  It follows $PS_i = a_i / d_i$.
- *Case* GG*:* $v(i) = 0$, $d_i \geq R$.
  Using the definition of bankruptcy game, it holds:
  $v(N) - v(N \setminus i) = R - max\{0, R - d_i\} = R$.
  It follows $PS_i = a_i / R$.
- *Case* MM*:* $v(i) \neq 0$, $d_i < R$.
  As in case MG, $v(N) - v(N \setminus i) = R - max\{0, R - d_i\} = d_i$.
  It follows $PS_i = (a_i - v(i)) / (d_i - v(i))$.
- *Case* MG*:* $v(i) \neq 0$, $d_i \geq R$.
  As in case GG , $v(N) - v(N \setminus i) = R - max\{0, R - d_i\} = R$.
  It follows $PS_i = (a_i - v(i)) / (R - v(i))$.

■

Case terminology

The PS rate differentiates four possible cases we name GM, GG, MM, MG. If a player asks less than R we call it *moderate player (*M*)* while if it asks more than R it is a *greedy player (*G*)*. In similar way, if the sum of the demand of a group of $n-1$ players exceeds R, that means $v(i) = 0$, the group is a *group of greedy players (G)* otherwise if $v(i) \neq 0$ we have a *group of moderate players (M)*. In the terminology we have used, the first character refers to the group of players while the second refers to the player itself.

Proposition 3.1.2 highlights that not only there is a relation between the DFS rate and the PS rate, but that the satisfaction of a user should be modified when it is considered as a

player inside a cooperative game. In particular, we can notice that:

- for case GM the PS rate coincides with the DFS one, i.e., $PS_i = DFS_i$;
- for case GG, the user satisfaction measured with the PS rate is higher than when it is measured with the DFS rate, i.e., $PS_i \geq DFS_i$;
- in the MM case, we have instead that $DFS_i \geq PS_i$.

We can also notice that the denominator of the PS rate is always different than zero. In cases GM and GG this is obviously true, in case MM the denominator is zero when $\sum_{i=1}^{n} d_i = R$ but in this case we set $PS_i = 1$ and in case MG the denominator is zero when $\sum_{j \in N, j \neq i} d_j = 0$ that is impossible.

Furthermore, from Proposition 3.1.2 it follows that if an allocation, i.e., a solution of an allocation problem that satisfies efficiency, non-negativity and demand boundedness, is an imputation, then $PS_i \in [0,1]$ for all the users. This holds due to the fact that for an allocation, in each of the four cases presented above, it is always verified that $v(N) - v(N \setminus i)$ is an upper bound for a_i.

Looking at the possible combinations of scenarios it is possible to characterize the players of an allocation problem, and hence how they measure their satisfaction, as follows.

Proposition 3.1.3 Given an allocation problem with $n = 2$ users, the following combinations are possible:

- GG: All the players are in scenario GG.
- MM: All the players are in scenario MM.
- GM-MG: One player is in scenario MG and the others are in scenario GM.

If $n \geq 3$, three combinations are added to the previous ones:

- GM: All the players are in scenario GM.
- GM-GG: Two groups of players: some players are in scenario GM and the others in scenario GG.
- GM-MM: Two groups of players: some players are in scenario GM and the others in scenario MM.

Proof. In case of three users, Table 3.3 validates the existence of the six scenarios listed above. We prove that all the other combinations of scenarios, i.e. MG, GG-MM, GG-MG, MM-MG, are impossible.

- MG: all the user has a demand $d_i \geq R$. This implies that for all user i it holds $\sum_{j \neq i} d_j > R$, but this is in contradiction with the fact that $v(i) \neq 0$.
- GG-MM: for each user i of type MM it holds $\sum_{j \neq i} d_j < R$ but it exists at least one user of type GG such that $d_i \geq R$. This implies that $\sum_{j \neq i} d_j > R$ that is in contradiction with the fact that $v(i) \neq 0$.
- GG-MG: all the users has a demand bigger or equal to R but it exists at least one user i in configuration MG such that $\sum_{j \neq i} d_j < R$. This is impossible due to the fact that each demand exceeds R.
- MM-MG: for each user i it holds $\sum_{j \neq i} d_j < R$ but it exists at least one user such that $d_i \geq R$. This produces a contradiction.

In case GM-MG, if there exists two users i, j of type MG, it holds that $d_i > R$ and $d_j > R$ and $\sum_{k \neq i} d_k < R$ and $\sum_{k \neq j} d_k < R$. This produces a contradiction because $d_i > R$ implies $\sum_{k \neq j} d_k > R$ and $d_i > R$ implies $\sum_{k \neq j} d_k > R$.

In case of two users, also the following scenarios are impossible:

- GM: both the users have a demand inferior to R ($d_1 < R$, $d_2 < R$). It follows $v(1) = R - d_2 > R$ and $v(2) = R - d_1 > R$ that contradicts $v(1) = v(2) = 0$.

- GM-GG the user 1 of type GM has $d_1 < R$ so $v(2) = R - d_1 > 0$. This implies that 2 can not be of type GG.
- GM-MM: as in case GM both users have a demand inferior to R. It follows $v(1) = R - d_2 > R$ and $v(2) = R - d_1 > R$, so none of the two user can not be of type GM.

■

Problem	Example
GM	$c = (5,5,5), R = 10$
GG	$c = (12,12,12), R = 10$
MM	$c = (4,4,4), R = 10$
GM-GG	$c = (3,8,12), R = 10$
GM-MM	$c = (2,6,6), R = 10$
GM-MG	$c = (2,3,12), R = 10$

Table 3.3: Allocation problems with three players.

3.1.1 Game-theoretical interpretation

To support and justify the use of the new satisfaction rate, we show an interesting game-theoretic interpretation.

Gately [52] introduced the concept of propensity to disrupt in order to remove the less fair imputations from the core. The idea was to investigate the gain of the player from the cooperation or, instead, its propensity to leave the cooperation, and to eliminate the imputation for which the propensity to leave the coalition for some players is excessively high. The formal definition of the propensity to disrupt is given in [53].

Definition 3.1.3 — Propensity to disrupt. For any allocation vector a, the *propensity to disrupt* $ptd(a,S)$ of a coalition $S \subset N$ $(S \neq \emptyset, N)$ is the ratio of the loss incurred by the complementary coalition $N \setminus S$ to the loss incurred by the coalition S itself if the payoff vector is abandoned. That is:

$$ptd(a,S) = \frac{a(N \setminus S) - v(N \setminus S)}{a(S) - v(S)}.$$

An equivalent definition of $ptd(a,S)$, when $\widetilde{a}(S) = v(N) - v(N \setminus S)$ is [52]:

$$ptd(a,S) = \frac{\widetilde{a}(S) - v(S)}{a(S) - v(S)} - 1.$$

The propensity to disrupt of a coalition S quantifies its desire to leave the coalition:

- when $a(S) = v(S)$ the propensity to disrupt of S is infinite and the desire of S to leave the coalition is maximum;
- when $a(S) > v(S)$ but $a(S) - v(S)$ is small, the value of $d(a,S)$ is very high and again S does not like the agreement;
- when $a(S) = v(N) - v(N \setminus S)$ the propensity to disrupt is zero and S has the propensity not to destroy the coalition;
- when $a(S) > v(N) - v(N \setminus S)$ the index is negative and there is an hyper-enthusiasm for such an agreement.

It holds the following interesting relationship between the propensity to disrupt and the player satisfaction rate.

Theorem 3.1.4 The relationship between the player satisfaction rate and the propensity to disrupt is: $PS_i = (ptd(a,i)+1)^{-1}$.

Proof. Using the alternative definition of $ptd(a,i)$ we have $ptd(a,i) = \frac{v(N)-v(N\setminus i)-v(i)}{a_i-v(i)} - 1$ but $\frac{v(N)-v(N\setminus i)-v(i)}{a_i-v(i)}$ is equal to $\frac{1}{PS_i}$ so $ptd(a,i) = \frac{1}{PS_i} - 1$. ■

It is worth noting that if $ptd(a,i)$ goes to infinity, then PS_i goes to 0 and if $ptd(a,i) = 0$ then $PS_i = 1$. This gives another interpretation of the PS rate. The higher the satisfaction is, the bigger the enthusiasm of i, for being in the coalition, is. On the contrary, the closer to zero the user satisfaction is, the higher the propensity of user i to leave the coalition is.

3.2 The mood value

In this section, we define a new resource allocation rule that we call the *mood value*. The fairness idea behind this rule is the same of the one behind the Jain's index. A repartition of a resource is fair when all the users have the same satisfaction. Furthermore, in the next section, we propose novel fairness indices as a modification of the classical fairness ones.

Using the proposed PS rate, we can define the mood value as follows.

Definition 3.2.1 — Mood value. Given an allocation problem characterized by (c,R), the allocation a such that $PS_i = PS_j \ \forall i,j \in N$ is called *mood value*.

Due to the relation between the propensity to disrupt and the player satisfaction, the fairest solution corresponds to the one in which every player has the same propensity to leave the coalition. Equalizing the propensity to disrupt of the users, this allocation equalizes the mood of each player. In particular, given a game, it exists a unique mood such that the satisfaction of each user is the same. The closer to zero the mood is, the more unsatisfied user i is; the closer to one the mood is, the more enthusiast the user i is.

Theorem 3.2.1 Let (d,R) characterize an allocation problem. There exists a unique mood m such that $PS_i = m \ \forall i \in N$:

$$m = \frac{R - min}{max - min} \tag{3.1}$$

where: $min = \sum_{i=1}^{n} v(i) = \sum_{i=1}^{n} min_i$, $max = \sum_{i=1}^{n} [R - v(N \setminus i)] = \sum_{i=1}^{n} max_i$.

The mood value is:

$$a_i^m = min_i + m(max_i - min_i). \tag{3.2}$$

Proof. Let $PS_i = m \ \forall i \in N$. It follows:

$$a_i = m(R - v(N \setminus i)) + (1-m)v(i). \tag{3.3}$$

Due to the efficiency property it holds: $\sum_{i=1}^{n} [m(R - v(N \setminus i)) + (1-m)v(i)] = R$. Thus equation (3.1) holds. Since a_i is the mood value iff $PS_i = m$, $\forall i \in N$ it holds $\frac{a_i - v(i)}{E - v(N\setminus i) - v(i)} = m$ and (3.2) remains proved.

■

From (3.1) we can notice that the mood depends only on the game setting, thus, given a bankruptcy game, we can know a priori the value of the mood that produces a fair allocation. Knowing m, on can easily calculate the mood value a_i^m.

The formula (3.2) shows that each user receives the minimum possible allocation $v(i)$ plus a portion m of the quantity $max_i - min_i$. The nearer to 1 the mood m is, the greater the happiness of each user is, and the closer to the maximum the allocation is. In fact, when m is equal to 1, the player receives exactly $R - v(N \setminus i)$, that is the maximum portion of resource that it can get, being inside a bankruptcy game. Depending only on the value of the minimum and the maximum payoff, the mood value coincides with the τ-value solution for bankruptcy games, also called *adjusted proportional rules (AP-Rule)* [36]. Before detailing this relationship, let us mention that in the bankruptcy games the core is $C(v) = \{a \in \mathbb{R}^N : \sum_{i \in N} a_i = v(N), v(i) \leq a_i \leq v(N) - v(N \setminus i), \forall i \subset N\}$ [36]. Moreover, the core cover $CC(v)$ is defined as the set of $a \in \mathbb{R}^N$ such that $\sum_{i \in N} a_i = v(N)$ and $m(v) \leq a \leq M(v)$.

Theorem 3.2.2 The mood value coincides with the τ-value solution for bankruptcy games, where the α value of the τ-value coincides with $1 - m$.

Proof. The τ-value is the linear combination of the minimal and the utopia payoff (2.7) and, given the alternative definition of the mood value (3.3), we have to simply prove that the utopia payoff for each player is given by $R - v(N \setminus i)$ and the minimal one by $v(i)$. α multiplies the minimal payoff in (2.7) while m the utopia one in (3.3), so trivially $\alpha = 1 - m$. As already argued in [36], the core $C(v)$ coincides with the core cover $CC(v)$. It follows that $m_i(v) = v(i)$ and $M_i(v) = v(N) - v(N \setminus i)$. ■

3.2.1 Properties

The mood value owns some interesting properties:

1. it is an allocation thus it satisfies non-negativity, demand boundedness and efficiency property;
2. it is stable, that means it belongs to the core of the game (Theorem 3.2.3);
3. it guarantees more than minimal right to each player ($a_i^m > v(i)$);
4. if $v(i) = v(j)$ and $v(N \setminus i) = v(N \setminus j)$ then $a_i^m = a_j^m$;
5. it is a strategy-proof allocation because a user has no advantages in splitting his demand.

Property 4 implies the equal treatment of equals ($d_i = d_j \Rightarrow a_i^m = a_j^m$) and equal treatment of greedy claimants (given a bankruptcy game, let G be the set of greedy players, i.e. such that $c_i > R$: if $|G| \geq 2$ then $a_i^m = a_j^m \; \forall i, j \in G$). This last property guarantees that even if a user has a cheating behavior, its demand is bounded by the available amount of resource R (see section 3.2.2 for more detail).

Curiel et al. in [36] prove that the τ-value solution for bankruptcy games can be characterized by (i) minimal right property, (ii) equal treatments of equals and (iii) strategy proofness property.

Theorem 3.2.3 The mood value belongs to the core of (N, v).

Proof. We should prove that $a_S^m \geq v(S), \forall S \subseteq N$.
If $v(S) = 0$ the condition holds due to the fact that $a_i^m < 0, \forall i \in N$. Now consider the case $v(S) > 0$. Suppose that $a_S^m < v(S) = R - \sum_{i \in N \setminus S} d_i$. For the efficiency property it holds

$E = a^m_S + a^m_{N\setminus S}$, implying $a^m_{N\setminus S} > \sum_{i\in N\setminus S} d_i$, which yields a contradiction with the fact that, according to the mood value solution, each user receives at most its demand. ■

In case of two players, it holds the following proposition.

Proposition 3.2.4 In a game with two players, the mood value coincides with the Shapley value and the mood is equal to 0.5.

Proof. Using (3.1) and (3.2) we have $m = 0.5$ and
$a^m_i = \frac{1}{2}v(i) + \frac{1}{2}(R - v(N\setminus i))$ for $i = \{1,2\}$.
The Shapley solution for a game with two players is:
$\phi(1) = \frac{1}{2}v(1) + \frac{1}{2}(R - v(2))$, $\phi(2) = \frac{1}{2}v(2) + \frac{1}{2}(R - v(1))$ and it coincides with a^m. ■

When the number of players is bigger than two, the mood value does not coincide any longer with the Shapley value as it is shown in the following example.

■ **Example 3.2** Let $d_i = (6,2,5)$ and let $R = 10$. The mood value is $a^m = (4.875, 1.25, 3.875)$ and the Shapley value is $a^s(4.833, 1.333, 3.833)$. ■

It is important to note that the mood value solution for a resource allocation problem produces an interesting solution also in the case in which the sum of the demands is inferior to the resource. This is a desirable property with an application perspective to systems in which bankruptcy situations can dynamically alternate with situations that are not bankruptcy situations. In such cases, each user receives the demand d_i and the excess $R - \sum_{i=1}^n d_i$ is divided equally between them.

Proposition 3.2.5 Let (c,R) such that $\sum_{i=1}^n d_i \leq R$. The mood value solution for user i is $a_i = d_i + \dfrac{R - \sum_{i=1}^n d_i}{n}$.

Proof. In order to calculate a_i, it is necessary the value of $v(i)$ and $v(N\setminus i)$. It holds: $v(i) = R - \sum_{j\neq i} d_i$, $v(N\setminus i) = R - d_i$.
Using the formula (3.1) and (3.2), we have $m = (n-1)/n$ and $a_i = R - \sum_{j\neq i} d_i + \frac{n-1}{n}\left(\sum_{i=1}^n d_i - R\right) = d_i + \frac{R - \sum_{i=1}^n d_i}{n}$. ■

The socio-economical interpretation of the mood value is similar to the one of the proportional. If with the proportional allocation the same portion of resource is allocated to each users, with the mood value we allocate the same portion of a refined demand, that takes into account the minimal and maximal right to each user.

Concluding, we show how we can simply adapt the mood value allocation in the case users sign a contract with the resource provider that guarantees them a minimal allocation, i.e. a minimum quantity of resource. Let us call this quantity *minimal demand* d^m. Under the hypothesis that the sum of the minimal demands is inferior to the available resource, the mood value allocation can be calculated as follows:

$$a^m_i = min^*_i + m^*(max_i - min^*_i). \tag{3.4}$$

where: $min^*_i = \max\{d^m_i, v(i)\}$, $max_i = \max\{d_i, R\}$, $m^* = \frac{R - \sum_{i=1}^n min^*_i}{\sum_{i=1}^n max_i - \sum_{i=1}^n min^*_i}$

3.2.2 Analysis of cheating behaviors

Let us investigate the consequences of users' cheating behaviors and in particular the relationship with the mood value, which, while it allows cheating behaviors, limits the gain of the cheating user[2]. Figure 3.2 shows the proportional allocation and the mood value when users cheat on their demands. The figure refers to an allocation problem where the available resource is 10 and the real demands of the users are 6 and 8. For the proportional allocation, the value of each allocation is the intersection between the black line, that is the Pareto-efficient frontier, and the line with angular coefficient given by the ratio between the demand of user 2 and the demand of user 1; for the mood value, the value is the intersection between the frontier and the line connecting the minimum and the maximum allocation of the two users. We can notice that, with the proportional allocation, a user is stimulated in asking more in order to obtain a bigger allocation. The mood value does not avoid cheating behavior as well: asking more, users can receive more if their real demand is smaller than the available resource; nevertheless, when the demand goes beyond the available resource R, the mood value limits it at the available resource amount so that users have no incentive in asking more than E. In our example the first user can increase at most its allocation from 4 to 6 and the second one from 6 to 7. We formalize this aspect as follows:

Proposition 3.2.6 A user has no incentive in asking more than the available resource if the allocation rule is the Mood value.

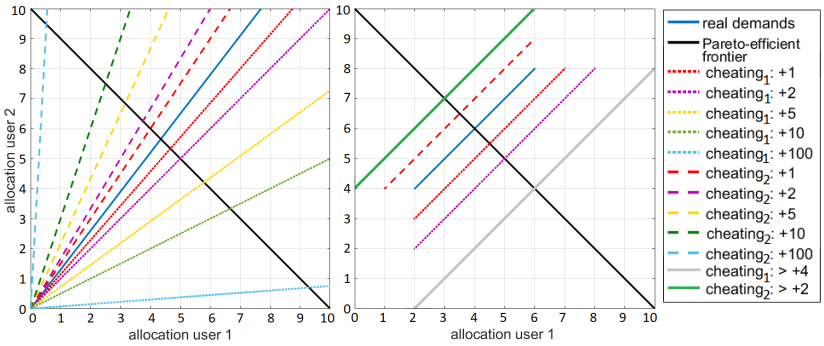
Proof. If a user i has a demand $d_i > R$ then the interval of value considered to calculate the mood value is $[min_i, R]$: increasing the demand the interval does not change because min_i depends only on R and on the demands of the other users. So it trivially follows that the mood value allocation for the user is not increasing. ■

We test now the gain of users in cheating for a 2-user allocation problem in which both users have a demand, expressing their real need, inferior to the available resource (d=$(6,8)$, R=10). In order to obtain a better allocation users can declare a need superior than how much they really need; in particular in our example from 10% to 400% more than the demand d_i.

Fig. 3.3 shows the heat map of the users gain as a function of the percentage of cheating of both users; we use the DFS satisfaction and the gain for user i is calculated as $Gain_i = \frac{a_i^c - a_i^r}{a_i^r}$, where a_i^c is i's allocation when there is cheating and a_i^r is the allocation when both users declare the true needs.

Being d_1 and d_2 bigger than $\frac{R}{2}$, the MMF allocation is always equal to $\frac{R}{2}$ for both users thus cheating brings no gain to users, otherwise a proportional or a mood value allocation allow users to gain or to lose. With a proportional allocation the gain or the loss of user can be very high (see Table 3.4), depending on the percentage (importance) of cheating, while the mood value allocation limits the gain or the loss. This follows from the property of equal treatment of greedy claimants, and from the fact that the mood value solution is close to the MMF fair allocation when the resource is scarce with respect to the demands of the users (see section 3.4.1 for more detail).

[2]It is worth mentioning that, in order to introduce mechanisms to guarantee truthful demands, a pricing scheme like the one proposed in [23] can be applied. Such a pricing scheme encourages the users to declare their truthful demands by maximizing their utilities for real declarations. See appendix A for the price implementation of the most well-known rules and for the mood value.



(a) Proportional (b) Mood value

Figure 3.2: Variation of the proportional and mood value allocations as function of the cheating. d=(6,8), R=10

	U.1-prop	U.2-prop	U.1-mood value	U.2-mood value
max gain	80%	50%	40%	12%
max lost	67%	60%	18%	27%

Table 3.4: Maximum gain and lost: comparison

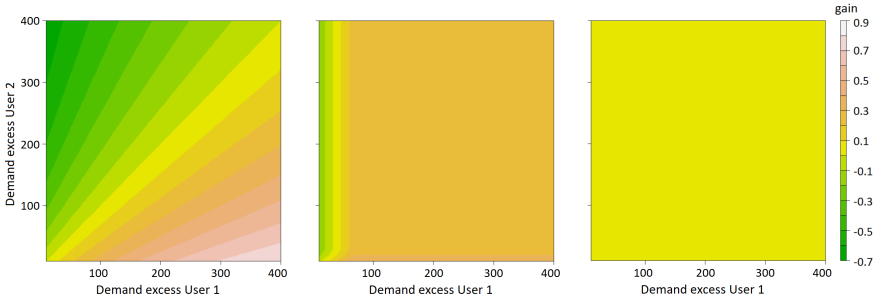
3.2.3 Mood Value Computation Complexity

Differently from the other allocation solutions inspired by game theory, in order to calculate this new allocation, only the value of $2n$ coalitions, i.e., the ones formed by the single players and the ones containing $n-1$ players, is needed. The time complexity of mood value computation is dominated by the complexity of computing $v(i)$ that is $\mathcal{O}(n)$. In dynamic situations, i.e., when the value of each of the n coalitions has to be updated at each slot of time, the complexity is therefore $\mathcal{O}(n^2)$, but it can be reduced to $\mathcal{O}(n)$ when $v(i)$ pre-computation is possible. This makes the mood value the best allocation rule in terms of time complexity together with the proportional allocation: the Shapley value has a time complexity of $\mathcal{O}(n!)$, while iterative algorithms for the computation of MMF and CEL allocations have a $\mathcal{O}(n^2 \log n)$ time complexity; the Nucleolus computation that in general is a NP-hard problem, in case of bankruptcy games can reduced to $\mathcal{O}(n \log n)$ [33, 54].

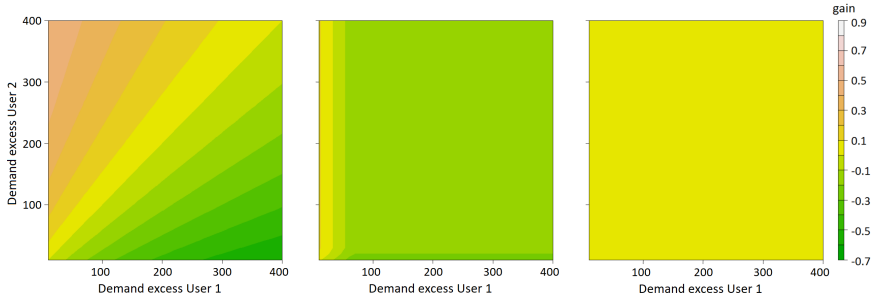
In terms of spatial complexity, the mood value, proportional, MMF and CEL allocations can be considered as equivalent and in the order of $\mathcal{O}(n)$. Instead, the Shapley value and the Nucleolus computations have a spatial complexity of $\mathcal{O}(2^n)$.

3.2.4 Interpretation with respect to traffic theory

The classical definition of proportional and weighted proportional allocations in network communications is done using as goal the maximization of a utility function. A typical application is the bandwidth sharing between elastic applications [5], i.e., protocols able to adapt the transmission rate upon detection of packet loss. In this context, we show how it is possible to revisit the mood value as a value resulting of the sum of the minimum allocation and the result of a weighted proportional allocation formulation where the weights are not the original demands, but new demands re-scaled accordingly to the maximum possible allocation knowing the available resource, and the minimum allocation under complete information sharing. More precisely, it holds the following proposition:



(a) User 1: Proportional (b) User 1: Mood V. (c) User 1: MMF



(d) User 2: Proportional (e) User 2: Mood V. (f) User 2: MMF

Figure 3.3: Users' gain in cheating environment (R=10, d=(6,8)).

Proposition 3.2.7 The mood value can be computed as the result of the following 4-step algorithm.

Step 1: We assign to each user the minimal right $v(i)$.
Step 2: We set the new value of the estate $R' = R - min = R - \sum_{i=1}^{n} v(i)$ and the new demands $d'_i = max_i - min_i$.
Step 3: We solve the following optimization problem

$$\begin{aligned} \underset{a}{\text{maximize}} \quad & \sum_{i=1}^{n} d'_i \log a_i \\ \text{subject to} \quad & a_i \leq d'_i,\ i = 1,\ldots,n \\ & a_i \geq 0,\ i = 1,\ldots,n \\ & \sum_{i=1}^{n} a_i = R' \end{aligned}$$

Step 4: The mood value coincides with the sum of the minimal right and the allocation given by step 3: $a^m_i = v(i) + a_i$.

Proof. We should prove that the result of the optimization problem is $a_i = md'_i$. The lagrangian of the problem is $L(a,\mu,\lambda) = \sum_{i=1}^{n} d'_i \log a_i - \mu^T(C - Aa) - \lambda(R' - \sum_{i=1}^{n} a_i)$, where the vector μ and λ are the lagrangian multipliers (or shadow prices), C is the vector of the demands $(d'_1,\ldots d'_n)$ and A is the identity matrix of dimension n. Then, $\frac{\partial L}{\partial a_i} = \frac{d'_i}{a_i} - \mu_i - \lambda$. The optimum is given by $a_i = \frac{d'_i}{\mu_i + \lambda}$ when $\mu \geq 0, Ay \leq C, \sum_{i=1}^{n} a_i = R'$ and $\mu^T(C - Aa) = 0$. This coincides with the case in which $\mu^T = 0$ and $\lambda \neq 0$. In fact, we have $\sum_{i=1}^{n} \frac{d'_i}{\lambda} = \frac{1}{\lambda}\sum_{i=1}^{n} d'_i = R'$. It follows that $\lambda = \frac{1}{R'}\sum_{i=1}^{n} d'_i$ is greater or equal to 1 and $a_i = \frac{d'_i}{\lambda}$ is less or equal to d'_i, that is an admissible solution. We can now notice that $\lambda = \frac{1}{R'}\sum_{i=1}^{n} d'_i = \frac{max - min}{R - min} = \frac{1}{m}$. It follows $a_i = md'_i$. ■

■ **Example 3.3** Let (d,R) be the allocation problem of Fig. 5.15b with d=$(3,2,13)$ and R=10. Following the algorithm we have:
Step 1: $v(i) = [0,0,5]$.
Step 2: $E' = 5$, $d'_i = [3,2,5]$.
Step 3: $x = [1.5,1,2.5]$
Step 4: $a^m = [1.5,1,7.5]$. ■

The algorithm shows that the mood value firstly assigns the minimal right (step 1) and secondly, considering the new allocation problem resulting after the first assignment (step 2), it allocates in a proportional way the resources (step 3). Then the proportion of allocated resource is the mood.

We provide two ways to compute the mood value: (3.2) and the 4-step algorithm of this section. It is clear that the computation of the mood value through the formula (3.2) is less complex than the one using the 4-step algorithm.

3.3 The Player fairness index

In our next analysis, we propose a modification of the Jain's index, introduced in Chapter 2. We remind that the *Jain's fairness index* is:

$$J = \left[\sum_{i=1}^{n}\left(\frac{a_i}{d_i}\right)\right]^2 \Big/ \left[n\sum_{i=1}^{n}\left(\frac{a_i}{d_i}\right)^2\right]$$

As we argued in section 3.1, the appropriate metric to rationally measure the satisfaction of the users, in complete information sharing settings, is the PS rate. Consequently, we replace in the Jain's index the DFS rate with the PS rate and we obtain a new measure of fairness, we call *Players fairness index*.

Definition 3.3.1 — Players fairness index. Given a problem (d,R) and an allocation a, the *players fairness index* is:

$$J_p = \left[\sum_{i=1}^{n}\left(PS_i\right)\right]^2 \Big/ n\sum_{i=1}^{n}\left(PS_i\right)^2$$

The resulting new fairness index we propose takes value 1 when all the users have the same satisfaction, i.e., when the allocation is the mood value.

Theorem 3.3.1 The players fairness index takes value in the interval $[\frac{1}{n},1]$ when the allocation belongs to the core.

Proof. From Theorem 3.1.1 follows that PS_i belongs to $[0,1]$ and that $\sum_{i=1}^{n} PS_i$ is always not negative. The maximum fairness is measured when all the users have the same PS rate, i.e.: $\left[\sum_{i=1}^{n}\left(PS_i\right)\right]^2 = \left(nPS_i\right)^2 \Rightarrow n\sum_{i=1}^{n}\left(PS_i\right)^2 = nn\left(PS_i\right)^2$. Thus $J_p = 1$.
The minimum fairness is measured when $\exists! k$ s.t. $PS_k \neq 0$ and $PS_j = 0\ \forall j \neq k$. In this case: $\left[\sum_{i=1}^{n}\left(PS_i\right)\right]^2 = \left(PS_k\right)^2 \Rightarrow n\sum_{i=1}^{n}\left(PS_i\right)^2 = n\left(PS_k\right)^2 \Rightarrow J_p = \frac{1}{n}$ ■

For core allocations, J_p takes value in the same interval of J making possible a comparison between the two indices. Furthermore, this index maintains all the good properties of the

Jain's index: the population size independence, the scale and metric independence, the boundedness and the continuity.

It is worth mentioning one more time that our proposed fairness index, as well as other indices from the literature that we recall in the background, are used in the context of resource allocation frameworks where the satisfaction rate of the users is not boolean (either satisfied or unsatisfied) and there are no strict service level agreements to be fully satisfied.

3.4 Numerical examples

We provide a numerical analysis of the proposed allocation and fairness index in 2 scenario: (i) when the resource to allocate is discrete, e.g. the Resource blocks (RBs) in the OFDMA scheduling use case and (ii) when the resource is divisible, e.g. cache or link bandwidths.

3.4.1 OFDMA scheduling use-case

In this section, we want to test the mood value and the new fairness index and to compare them with the classical allocations and the Jain's index. We run numerical simulations of the cellular OFDMA (Orthogonal Frequency-Division Multiple Access) spectrum scheduling problem.

In OFDMA scheduling, a base station unit or controller dynamically receives new users and decides which spectrum portion to allocate to which users, as a function of (i) their signal power and interference levels (aspects that characterize their demands), (ii) the other users to manage concurrently (i.e., users that arrive together during a OFDMA frame time or still in the scheduler queue) and (iii) the spectrum already allocated to existing users. The number of users to manage concurrently is basically limited to few (up to a dozen), except in high mobility environments. It is worth mentioning that in OFDMA, the unit of spectrum for the allocation is the Resource Block (RB).

We suppose that the maximum number of available resource blocks is equal to 100; this coincides, in LTE standard, with the number of resource blocks for a bandwidth of 20 MHz. Furthermore, we consider a range for demand generation between 0 and 100 RBs using two different distributions: (i) a uniform distribution between 0 and 100, and (ii) a Zipf's distribution $f(k,s,N) = \left[\frac{1}{k^s}\right] / \left[\sum_{i=1}^{N} \frac{1}{n^s}\right]$ where the parameters k and s are equal to 100 and 0.4, respectively. We choose these values for the two parameters of the Zipf's distribution because they permit to fit well a realistic demand distribution[3].

We run different instances varying the available resource (i.e., R) from 5 to 95, with the interpretation of being the available number of resource blocks at the instant the OFDMA scheduling problem is faced. We simulate 300 bankruptcy games with 3 and 10 users in the scheduler.

Fig. 3.4, 3.5, 3.6 and 3.7 show the results of the simulations. We consider six allocations discussed so far in the background and in this chapter: Proportional, Shapley, Nucleolus, Mood Value, MMF and CEL. We calculate the Jain's fairness index and the players fairness index and we plot, for each value of R and each index, the mean value in between the first and third quartile lines.

[3]Taking inspiration from cellular (OFDMA) resource allocation studies we emulated an indoor scenario of femtocells using the WINNER II channel model [55]: generating in a uniform way 10000 users around the cell station between 3 and 100 m, we associate resource blocks (RBs) to each of them with a transmit power between 1 and 100 dB. The resulting RB distribution is well fit by a Weibull distribution and the Zipf's distribution can be seen as a discrete variation of the Pareto distribution, that belongs to the same distributions family of the Weibull one.

In the 3-user cases (Fig. 3.4 and Fig. 3.6) the fairest allocation accordingly to the Jain's index is the proportional rule, and accordingly to the players' fairness index is the mood value. For both allocations, the value of the respective fairness index is equal to 1 for almost all the values of the available resource; only when the resource is scarce the value decreases due to the fact that the solutions are rounded. We can also notice that the mood value allocation has a behavior similar to the Shapley value and to the nucleolus and that it is close to the proportional allocation when the resource is between 50 and 80, and to the MMF allocation when the resource is scarce. For this last allocation the PF index has high value when the available resource is small (high congestion), i.e., when there are many greedy users. In fact, the MMF allocation and the mood value are close: in such cases, both have the property of treating equally the greedy claimant, giving them the same portion of resource, independently of their demands.

In the 10-user cases (Fig. 3.5 and Fig. 3.7), we can observe a similar trend for the two indices, but their values decreases, in particular in case of scarce resource, due to the discretization of the solution. Again, the mood value has a behavior similar to the Shapley value, but it is no more close to the nucleolus.

For each scenario, we can notice that the mood value solution gives a better performance in term of fairness, measured with both indices, with respect to the MMF allocation, that is the one mostly used in this type of problems. In particular, the difference in term of fairness between the two allocations increases when the number of users in the system increases.

3.4.2 Continuous allocation example

Differently from the previous analysis around the OFDMA scheduling use-case where to a user can be given a discrete and limited number of RBs, we now consider divisible resources as caches or link bandwidths (i.e., a quasi-continuum situation with the bit granularity but with millions of bits for a single allocation). In the appendix B, we provide the same type of results than the one in the previous section comparing rules and fairness indices, which lead to similar conclusions.

The continuous allocation allows us to better stress the situations in which different users fall in, as discussed in Prop. 3.1.2 and 3.1.3, and that as a function of the congestion level computed as the global demand over the available resource. Due to its non-informative nature, we consider a uniform distribution of the demands between 0 and 100 units of resource (e.g., Mega-bytes or Mega-bit/s) and we run different instances with a ratio of E (available resource) ranging from 5% to 95% of the global demand. We simulate 300 bankruptcy games with 3 and 10 users in the system waiting for an allocation.

Fig. 3.8 shows the users configuration as a function of the available resource. With 3 users (Fig. 3.8a), for low value of R almost all are greedy players (GG case) due to the fact that the resource is small; increasing R the number of moderate players (GM) increases but also some users in configuration MG appear. In fact, increasing R, some greedy players become moderate while the others remain greedy; some of them are greedy inside a group of greedy users (GG), while some others greedy inside a group of moderate ones (MG). When the available resource is higher than half of the global demand, greedy players GG disappear and the number of moderate players increases. In particular, users MM appear and they become the majority when the resource is large. With 10 users (Fig. 3.8b), we find a similar trend than with 3 users in the number of moderate players that increases increasing R. However, MG users disappear; in fact, it holds that it can exist at most one MG user in a game (see Prop. 3.1.3) and, due to the higher number of users in the system,

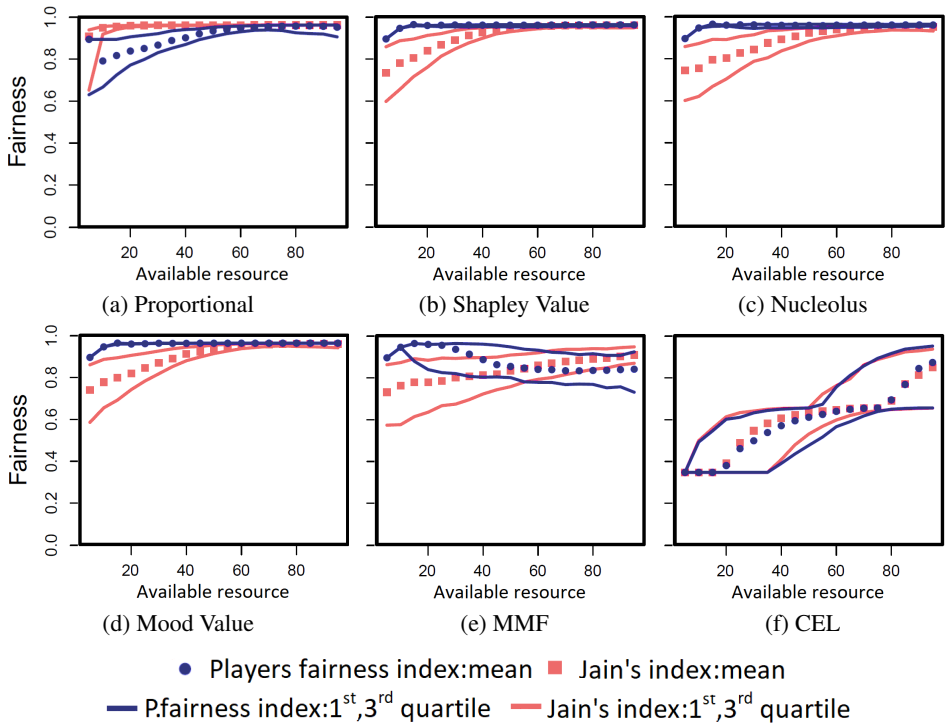


Figure 3.4: Fairness w.r.t. the available resource (3 users, uniform)

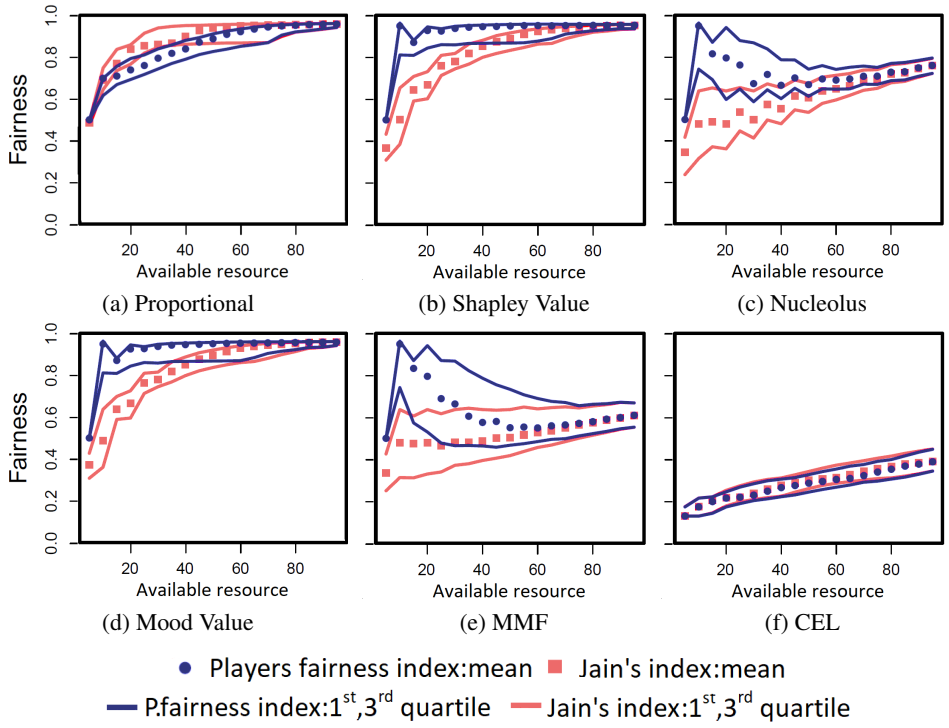


Figure 3.5: Fairness w.r.t. the available resource (10 users, uniform)

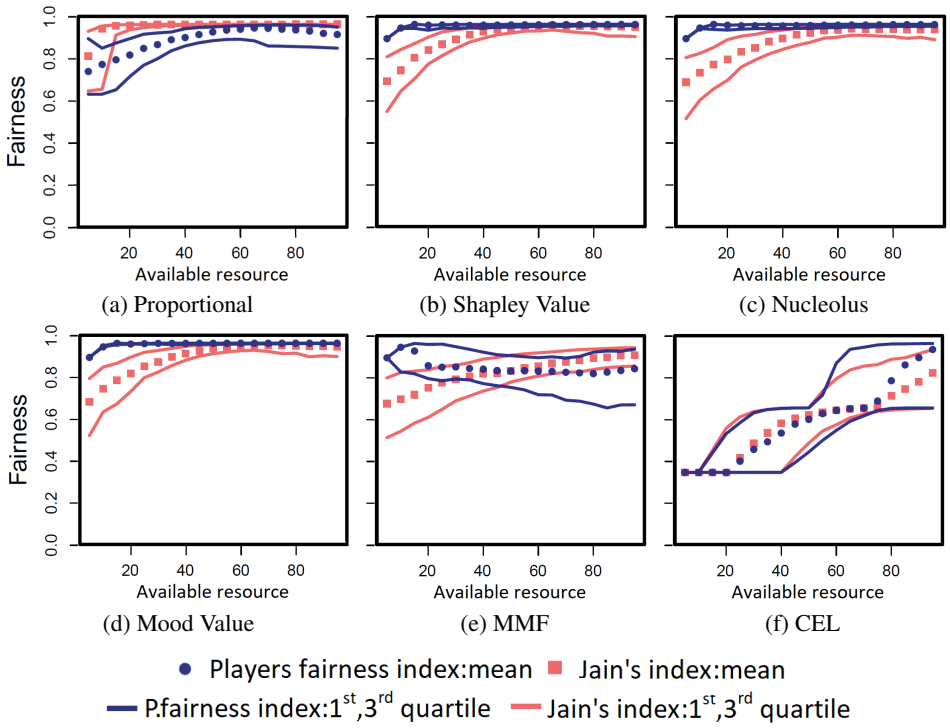


Figure 3.6: Fairness w.r.t. the available resource (3 users, Zipf)

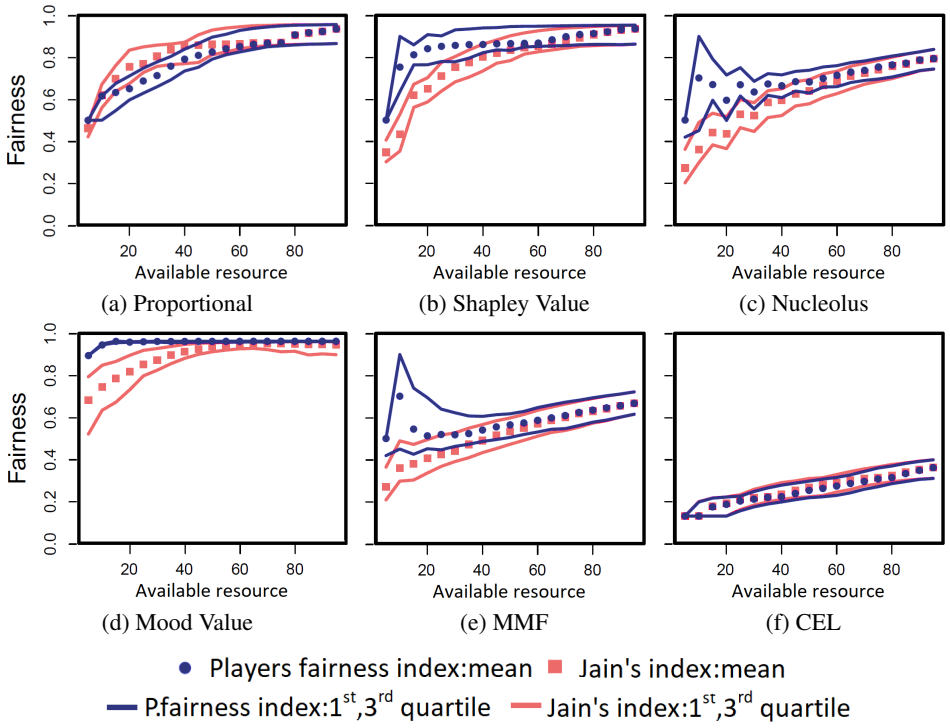
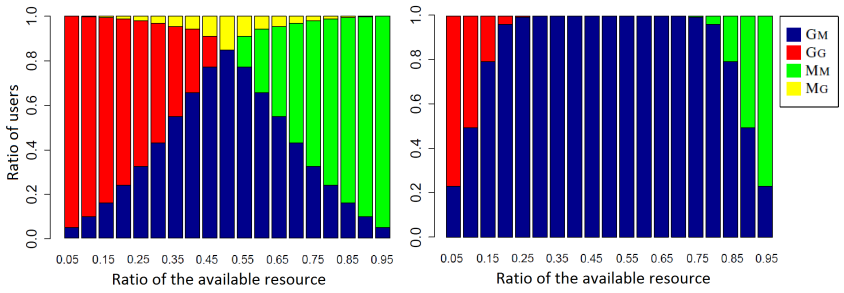


Figure 3.7: Fairness w.r.t. the available resource (10 users, Zipf)



(a) 3 users, uniform (b) 10 users, uniform

Figure 3.8: User cases distribution

it is very unlikely that there exists only a player MG in the system such that the sum of the demands of the other $n-1$ players exceeds R.

To support the analysis of the user cases distribution, we plot the ratio of the four user types increasing the number of users from 3 to 15 and setting the demands in a uniform way between 0 and 100 (Fig. 3.9). As we already noticed, the number of MG users is small and it becomes negligible starting from a number of users higher than 5 (Fig. 3.9d). Furthermore, increasing the number of users, the range of available resource in which all the users are of type GM increases. In fact, if in 3-users scenarios a user can be of each possible type, in 15-user scenarios we find users different from type GG only if the ratio of the available resource is less than 0.2 or higher than 0.8. When users are of type GG, their satisfaction is measured in the classical way with the DFS rate; it follows that with a sufficiently high number of users, the new proposed approach gives different results from the classical one only in case of high congestion or in case of low congestion. In order to capture all the possible scenarios, we choose a low number of users for the simulations.

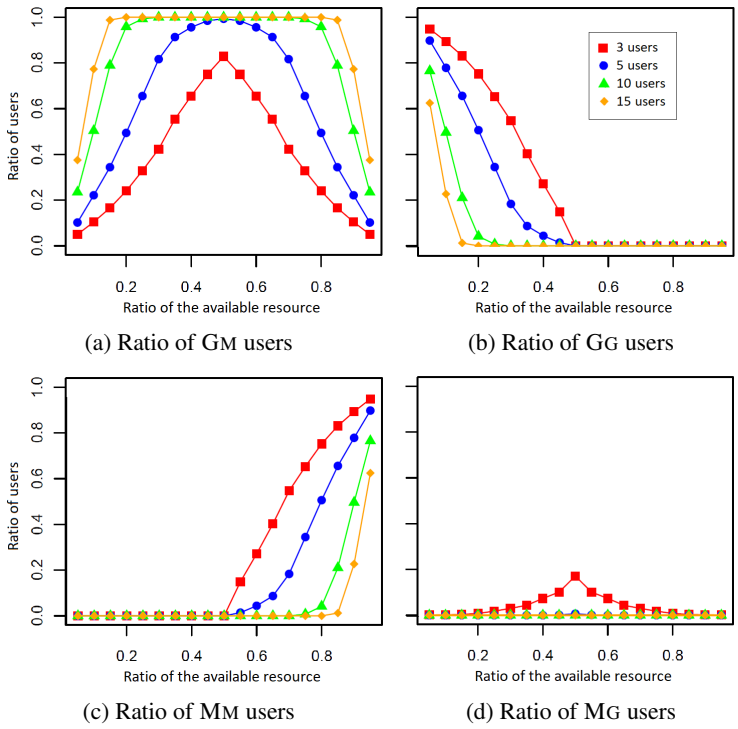
Summarizing, the simulations show how the proposed mood value produces different results with respect to the classical approach; in particular, in case of few users or, if the number of users is sufficiently high, in case of high or low congestion. The Mood Value is able to nicely weight the nature (greedy or moderate) of users; in particular, it is close to the MMF allocation when the resource is scarce and to the proportional allocation when the resource is close to the global demand. Furthermore, it is worth noticing that with respect the Shapley value, the results show that the Mood Value has a similar good behavior in terms of fairness, with the key advantage of having a much lower computation time complexity.

3.5 Dynamics in a multi-provider context

We test the behavior of the different resource allocation rules in a strategic context with multiple competing providers. We run this analysis to (i) study the global system efficiency under the different allocation rules, and to (ii) qualify the motivation in adopting the mood value for a network provider.

3.5.1 Impact on system efficiency

For the first analysis we consider two providers, provider 1 and provider 2, providing the same service on a competitive market. Each of them has its own capacity ($R1$, $R2$) and



(a) Ratio of GM users (b) Ratio of GG users

(c) Ratio of MM users (d) Ratio of MG users

Figure 3.9: Ratio of users as function of the users number

its own way to allocate resources. We consider only the MMF, the mood value and the proportional allocation rules.

Users are selfish and they have no binding agreements with the provider thus they can move from one provider to the other in order to reach a better satisfaction with respect of their allocation. The satisfaction is calculated using the demand fraction satisfaction rate (DFS rate) with the consequence that users prefer to move if their allocation is strictly bigger.

We set up a simulation in order to investigate the equilibrium configuration of the user to provider choices. We are particularly interested in the percentage of time in which the simulation produces 'agglomerated' configurations, i.e., when the equilibrium configuration coincides in having all the users served by only one provider. This is the worst configuration in terms of efficiency: the equilibrium is globally inefficient because the entire resource of one operator gets wasted.

In order to find the equilibrium configuration we randomly choose one of the two providers and we calculate the solution when all the users are served by this provider. Having the initial state, we calculate, for each player, the gain in moving to the other provider: if the gain is positive, it has propensity to move to the other provider, otherwise it prefers to stay in the currently provider. We choose randomly one user between the users that have positive gain and we move it in the other provider. We repeat the algorithm until we reach an equilibrium configuration (Algorithm 1).

For our simulations we generate $R1$ randomly between 0 and 20 units and we consider fixed ratios between $R1$ and $R2$ ($R2 = \frac{1}{10}R1$, $R2 = \frac{2}{10}R1$, ..., $R2 = R1$, ..., $R2 = 10R1$). For each scenario, we generate 200 resource allocation problem instances with 3 users, choosing the demands uniformly between 0 and $R1 + R2$ (CASE 1). We repeat the simulations adding the constraints that both $R1$ and $R2$ are bigger than the smaller demand,

Algorithm 1 Dynamic allocation in strategic context

Input: $R1, R2, d$
Output: a_1^{eq}, a_2^{eq}

$s \leftarrow 1$
Random selection between provider 1 and provider 2
if provider 1 is selected **then**
  $a_1(1) \leftarrow$ Allocation when users are in the provider 1
  $a_2(1) \leftarrow$ Null vector
else
  $a_2(1) \leftarrow$ Allocation when users are in the provider 2
  $a_1(1) \leftarrow$ Null vector
end if
repeat
  for all i in N **do**
    if i in provider 1 **then**
      $[a_2^*]_i \leftarrow$ Allocation of user i when it moves in provider 2
      $\mathscr{G}(s)_i \leftarrow [a_2^*]_i - [a_1(s)]_i$
    else
      $[a_1^*]_i \leftarrow$ Allocation of user i when it moves in provider 1
      $\mathscr{G}(s)_i \leftarrow [a_1^*]_i - [a_2(s)]_i$
    end if
  end for
  $j \leftarrow$ Random selection of a user with $\mathscr{G}(s)_j > 0$
  $s \leftarrow s+1$
  $a_1(s) \leftarrow$ New allocation when j is moving
  $a_2(s) \leftarrow$ New allocation when j is moving
until $\mathscr{G}(s) \leq 0$
$a_1^{eq} \leftarrow a_1(s-1)$
$a_2^{eq} \leftarrow a_2(s-1)$

i.e., we avoid situations in which all the demands are bigger than the estate (CASE 2). This second case makes more sense in some configurations and it follows the trivial idea that usually a provider owns enough resource to completely satisfy at least the user with the smallest demand.

For both cases, we plot the results in three scenarios:

- MMF-MOOD: the first provider allocates the resource using the MMF rule and the second with the mood value.
- MMF-PROP: the first provider allocates using the MMF rule and the second with the proportional rule.
- MOOD-PROP: the first provider allocates using the mood value rule and the second with the proportional rule.

Fig. 3.10 and Fig. 3.11 show the result of the analysis. In CASE 1 (Fig. 3.10) we can notice that in each scenario the percentage of agglomerated equilibria is high when the gap between the quantity of available resource in the two providers is considerable; for instance, if one provider's resource is four times higher than the one of the other provider. In these cases, there is a high probability that all the users, including the one with the smaller demand, reach a better allocation choosing the provider with the widest

resource. In this case, the percentage of agglomerated equilibria slightly differs from one allocation to the other and in particular it is slightly higher when the provider allocates using the MMF rule; differently, in CASE 2 (Fig. 3.11) the percentage of agglomerated equilibria differs a lot with respect to the allocation that the providers adopt. In particular, we can notice that the number of agglomerated equilibria produced by the MMF allocation slightly decreases with respect to CASE 1, while the number of the ones produced by the proportional and mood value solution drastically decreases. We can report that in this case there is a resource waste that goes up to 26% (case $R2 = \frac{3}{10}R1$) of the global resource with the MMF allocation, and it does not exceed 1,7% (case $R2 = \frac{2}{10}R1$) with the mood value allocation.

3.5.2 Impact on user retention

In a second analysis, we aim to assess which type of users are attracted by which allocation rule. In this case we consider that operators have equal resources to avoid the presence of inefficient equilibria and we set two scenarios; we randomly generate 200 times R1 equal to R2 and 10 users such that in average the ratio of available resource in first scenario is 10% (high congestion) and in second is 90% (low congestion).

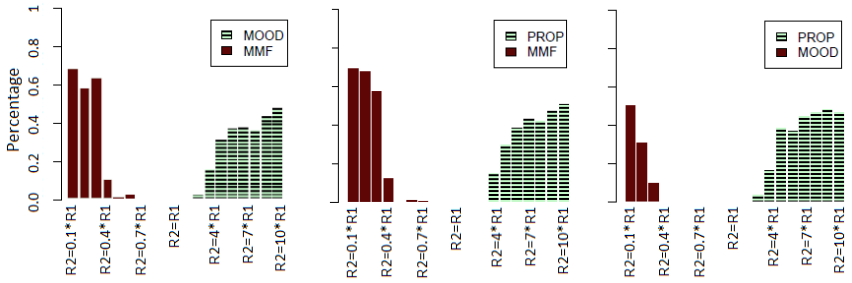
Fig. 3.12 and Fig. 3.13 show the distribution of the four types of users previously discussed, for the three different pairs of allocation rules among the two providers, and for the two congestion scenarios[4]. We can notice that in case of high congestion there are only GM and GG users, while without congestion there are GM and MM users. In the former case, the mood value and the proportional allocation attract the users with high demand when the allocation of the other provider is MMF, while in the MOOD-PROP case there is a symmetric distribution in the users' type. We can also notice that in the MMF-MOOD case the mood value gives a median number of users 20% higher than with the MMF allocation. Moreover, in the high congestion scenario (Fig. 3.13), the MMF mostly attracts MM users, i.e., users with a demand lower than the available resource and such that the sum of other users demands is less than E; this means that if one of them leaves the provider, there is no more congestion on that provider and there is an excess of resource that gets wasted.

Therefore, the mood value and the proportional allocation have a similar impact on user retention from a provider perspective: they appear better than the MMF allocation in a multi-provider strategic context because they can better use the resource of the providers, avoid resource waste. In particular, the gain of using these two allocations is conspicuous when we avoid (unlikely) situations in which all the users ask more than the resource available in one provider. Furthermore, in case of high congestion, the mood value attracts more users and users of higher demands, with respect to the MMF; in case of low congestion, similarly to the proportional allocation, it reduces the resource waste due to provider change.

3.6 Summary

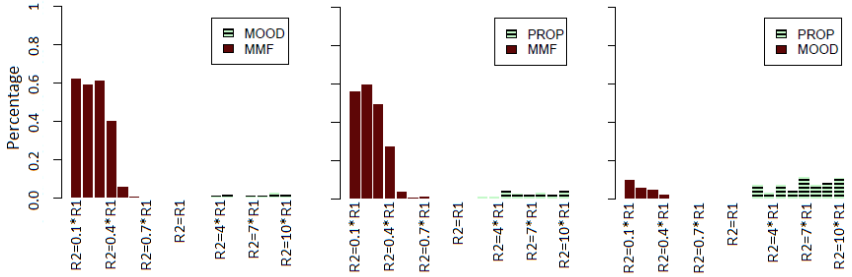
In this chapter we proposed a game-theoretical approach to analyze and solve resource allocation problems, going beyond classical approaches that do not explore the setting where users can be aware of other users' demand and the available resource.

[4] e.g., in Fig. 3.12a the first provider uses the MMF rule and the user distribution is given by the first four boxplots, the fifth one giving the sum; the remaining boxplots give the same numbers for the second provider. Each boxplot reports, from bottom to top, the minimum, first quartile, median, third quartile, and maximum.



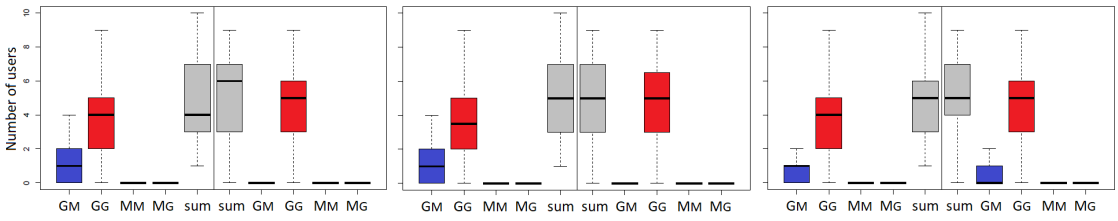
(a) MMF-MOOD (b) MMF-PROP (c) MOOD-PROP

Figure 3.10: Percentage of agglomerated equilibria in CASE 1.



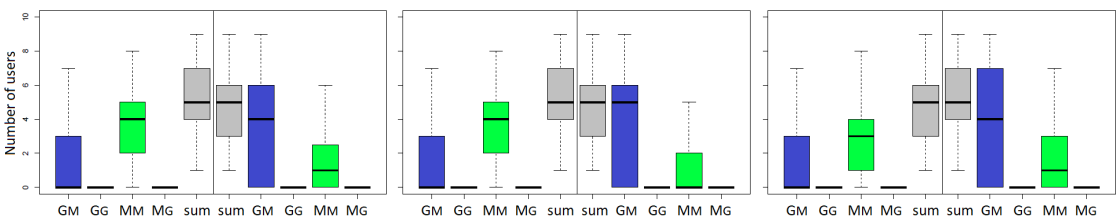
(a) MMF-MOOD (b) MMF-PROP (c) MOOD-PROP

Figure 3.11: Percentage of agglomerated equilibria in CASE 2.



(a) MMF-MOOD (b) MMF-PROP (c) MOOD-PROP

Figure 3.12: Distribution of the four type of users - average level of $\rho = 10\%$



(a) MMF-MOOD (b) MMF-PROP (c) MOOD-PROP

Figure 3.13: Distribution of the four type of users - average level of $\rho = 90\%$

In particular, we proposed a new way of quantifying the user satisfaction taking into account the deeper knowledge of the users with respect to the resource allocation problem, and a new fairness index as enhancement of the family of fairness measures, describing and comparing their mathematical properties in detail. According to these new concepts, we propose a new resource allocation rule called the 'Mood Value' that meets the goal of providing the fairest resource allocation and we position it with respect to game theory metrics as well the common theory of fair allocation in networks.

Finally, we tested our ideas via numerical simulations of representative demand distributions and we provide two further analysis showing the advantages of the mood value allocation in a strategic multi-provider context and in the presence of cheating users. Besides the properties we analytically prove, the results of our simulations and of our analysis can be summarized as follows:

- the mood value allocation is able to take into account the nature of the users and the level of congestion of the system and consequently to choose the fairest solution;
- in case of high congestion, the mood value allocates the resources in way similar to the MMF allocation, while in case of low congestion similarly to the proportional allocation; this implies that if users cheat on their demand, they have a limited gain because the mood value converges to a MMF allocation under high congestion;
- the mood value has lower computational complexity than other game theoretical solutions as the Shapley value;
- in case of strategic contest, the mood value guarantees the efficiency of the equilibrium, except, with a low percentage, in case of strong resource imbalance between the two providers and it attracts more users and with higher demands in case of high system congestion.

In the following chapter we estimate the impact of the inaccurate information on the allocation and we provide some fairness consideration when the network setting are such that there is an inaccurate information about the users demands and the available resource.

4. Resource allocation with inaccurate information sharing

Classically, the network setting is such that users have little information about the available resources and demands of other users. Nonetheless, with the emergence of new networking features such as 5G infrastructure sharing and programmability in SDN, and for auditability requirements (i.e., to ensure tenants fair sharing), we explain in chapter 3 how network setting is evolving toward a complete information sharing situation so that all users can be aware of the demands of the other users and of the available resources for resource allocation systems.

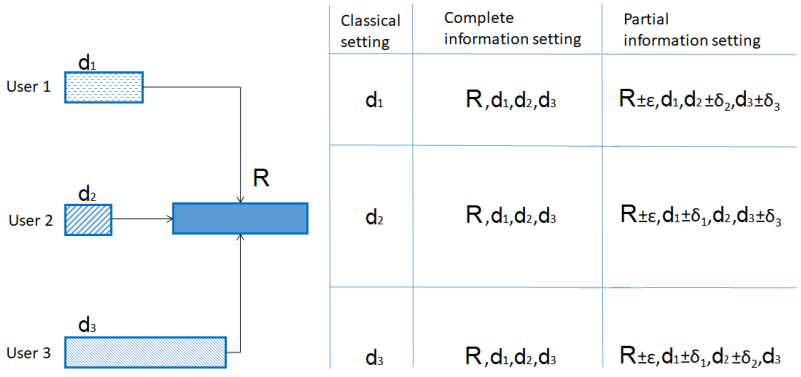


Figure 4.1: Information sharing contexts in resource allocation.

As an intermediate context between the classical no information sharing and the complete information sharing, we consider the scenario in which the information is inaccurate. In this chapter, we study the behavior of resource allocation rules under this scenario that we call *inaccurate information sharing context*, where the amount of available resource is known up to a constant (Fig. 4.1); we also highlight the impact of

The work presented in this chapter is partially presented in [56]

Allocation rule		ERR_i
Weighted proportional		$\pm\frac{d_i}{\sum_{i=1}^{n} d_i}\varepsilon$
MMF		$\begin{cases} 0, & \text{if } i=1,..,k \\ \frac{\pm\varepsilon}{n-k}, & \text{if } i=k+1,...,n. \end{cases}$
Mood value	GM	$\pm\frac{d_i}{\sum_{i=1}^{n} d_i}\varepsilon$
	GG	$\pm\varepsilon/n$
	MM	$\pm\varepsilon/n$
	GM-GG	$\begin{cases} \pm\frac{\sum_{i\in N_1} d_i\varepsilon d_i}{(\sum_{i\in N_1} d_i+n_2R)(\sum_{i\in N_1} d_i+n_2R\pm n_2\varepsilon)}, & i\in N_1 \\ \left(\frac{\varepsilon\pm 2R}{c\pm n_2\varepsilon}\mp\frac{R^2n_2}{c(c\pm n_2\varepsilon)}\right)\varepsilon, & i\in N_2 \end{cases}$
	GM-MM	$\begin{cases} \pm\sum_{i\in N_1} d_i\varepsilon d_i/(eb), & i\in N_1 \\ \pm\left(\frac{\sum_{i=1}^{n} d_i-R\mp\varepsilon}{b}+\frac{(\sum_{i=1}^{n} d_i-R)\sum_{i\in N_1} d_i}{eb}\right)\varepsilon, & i\in N_2 \end{cases}$
	GM-MG	$\begin{cases} 0, & i\in N_1 \\ \pm\varepsilon, & i\in N_2 \end{cases}$

Table 4.1: Evaluation errors with misknowledge on the available resource. N_1=set of users GM, N_2=set of users of the other type. $n_2(\sum_{i\in N_2} d_i - R)+(n_2+1)(\sum_{i\in N_1} d_i)=e$, $n_2(\sum_{i\in N_2} d_i - R\mp\varepsilon)+(n_2+1)(\sum_{i\in N_1} d_i)=b$, $\sum_{i\in N_1} d_i+n_2R=c$.

inaccurate information sharing on the demand of the other users. Indeed, in certain practical situations, such as in radio resource availability or in systems over/under provisioned by the infrastructure provider, it is likely to suffer from inaccurate information on the available resources to be shared. Furthermore, complete sharing may not be possible since this may require a lot of exchanges of updates causing a large overhead. While the problem of inaccurate information in networks has been studied before [57], a formal treatment on the error estimate and fairness has not been studied for different allocations schemes.

We are interested in evaluating the error on the allocation when users are in a inaccurate information context (Fig. 4.1). We treat first the case where there is an error only on the available resource and later the case in which there is an error on the users demand. For both the cases we provide some fairness considerations.

4.1 Error on the available resource

4.1.1 Error estimate

For each user $i \in N$ the error ERR_i is defined as $|\hat{a}_i - a_i|$, where a_i is the share obtained by i when the resource is R and $\hat{a}_i$ is the allocation for user i when resource is $R\pm\varepsilon$.

Table 4.1 summarizes the value of the error, when we consider the weighted proportional allocation rule, the MMF allocation and the mood value. The errors are calculated as follows:

Weighted proportional allocation

This allocation coincides with the allocation that assigns the resource proportionally to the demand, i.e., $a_i^p = d_i R / \sum_{i=1}^{n} d_i$ when the resource is R. If users believe that the available resource is $R \pm \varepsilon$ then the allocation is $\hat{a}_i^p = d_i(R \pm \varepsilon) / \sum_{i=1}^{n} d_i$, which implies that the error on the allocation for each user is:

$$ERR_i = \pm \frac{d_i}{\sum_{i=1}^{n} d_i} \varepsilon \tag{4.1}$$

The error ε is divided between the users proportionally to their demands.

MMF allocation

We consider the hypothesis that ε is small enough not to change the nature of the user. This means that if $d_i < \frac{R}{n}$, it holds also that $d_i < \frac{R-\varepsilon}{n}$. It follows that the users with small demands receive the same amount of resource (i.e., their demand), while the excess ε is equally divided between the users that receive less than their demand. When the first k users receive their demands, the error is:

$$ERR_i = \begin{cases} 0, & \text{if } i = 1,..,k \\ \frac{\pm\varepsilon}{n-k}, & \text{if } i = k+1,...,n. \end{cases} \tag{4.2}$$

Mood value

For the mood value allocation, we need to consider that we have four types of users when we take into account the minimum and the maximum value they can get and six combinations of users (Table 3.2 and Prop. 3.1.3). We again consider the hypothesis that ε is small enough not to change the nature of the user. This means that, e.g., if $d_i \geq R$ for a user $i \in N$ it also holds $d_i \geq R + \varepsilon$, if $\sum_{j \neq i} d_j < R$ it holds also $\sum_{j \neq i} d_j < R - \varepsilon$, and so on.

- Case GM
  This case coincides with the weighted proportional allocation. For each user i the error is given by (4.1).
- Case GG
  In this case, if the resource is R it holds that $min_i = 0$, $max_i = R$ for each user i. The value of the mood is $m = \frac{R-0}{nR-0} = \frac{1}{n}$, and the mood value is $a_i^m = \frac{R}{n}$. If the value of the available resource is $R \pm \varepsilon$ the value of the mood $\hat{m}$ is again equal to $\hat{m} = \frac{1}{n}$ and the mood value is $\hat{a}_i^m = \frac{R \pm \varepsilon}{n}$. It follows that for each user i the error is equal to:

  $$ERR_i = \pm \varepsilon / n. \tag{4.3}$$

  In this case, the error is divided equally between the users without considering the value of their demands.
- Case MM
  In this case, if the resource is E it holds that $min_i \neq 0$, $max_i = d_i$ for each user i. The value of the mood is $m = \frac{R - n(R) + (n-1)\sum_{i=1}^{n} d_i}{\sum_{i=1}^{n} d_i - n(R) + (n-1)\sum_{i=1}^{n} d_i} = \frac{n-1}{n}$, and the mood value is $a_i^m = R - \sum_{j \neq i} d_j + \frac{n-1}{n}(\sum_{i=1}^{n} d_i - R)$. If the value of the available resource is $R \pm \varepsilon$, the

value of the mood $\hat{m}$ is given by: $\hat{m} = \frac{R\pm\varepsilon - n(R\pm\varepsilon) + (n-1)\sum_{i=1}^{n} d_i}{\sum_{i=1}^{n} d_i - n(R\pm\varepsilon) + (n-1)\sum_{i=1}^{n} d_i} = \frac{n-1}{n}$ and the mood value is $\hat{a}_i^m = R\pm\varepsilon - \sum_{j\neq i} d_j + \frac{n-1}{n}(\sum_{i=1}^{n} d_i - R\mp\varepsilon)$. It follows that for each user i the error is:

$$ERR_i = \pm\varepsilon/n. \tag{4.4}$$

The error is equally divided between the users also here.

- Case GM-GG
  Let $N = N_1 \cup N_2$ be partitioned into two disjoint sets N_1 and N_2 representing the set of user of type GM and GG, respectively. When the resource is R, the value of the mood is $m = \frac{R}{\sum_{i\in N_1} d_i + n_2 R}$, and the mood value is $a_i^m = \frac{R}{\sum_{i\in N_1} d_i + n_2 R} d_i$ if $i \in N_1$ and $a_i^m = \frac{R^2}{\sum_{i\in N_1} d_i + n_2 R}$ if $i \in N_2$.
  If the value of the available resource is $R\pm\varepsilon$, only the maximum value for the user GG is changing. The value of the mood is $\hat{m} = \frac{R\pm\varepsilon}{\sum_{i\in N_1} d_i + n_2 R \pm n_2\varepsilon}$, and the mood value is $\hat{a}_i^m = \frac{R\pm\varepsilon}{\sum_{i\in N_1} d_i + n_2 R \pm n_2\varepsilon} d_i$ if $i \in N_1$ and $\hat{a}_i^m = \frac{(R\pm\varepsilon)^2}{\sum_{i\in N_1} d_i + n_2 R \pm n_2\varepsilon}$ if $i \in N_2$. Called c the denominator of m, the error is :

$$ERR_i = \begin{cases} \pm\frac{\sum_{i\in N_1} d_i \varepsilon d_i}{(\sum_{i\in N_1} d_i + n_2 R)(\sum_{i\in N_1} d_i + n_2 R \pm n_2\varepsilon)}, & i \in N_1 \\ \left(\frac{\varepsilon \pm 2R}{c \pm n_2\varepsilon} \mp \frac{R^2 n_2}{c(c\pm n_2\varepsilon)}\right)\varepsilon, & i \in N_2 \end{cases} \tag{4.5}$$

- Case GM-MM
  Let $N = N_1 \cup N_2$ be partitioned into two disjoint sets N_1 and N_2 representing the set of user of type GM and MM, respectively. When the resource is R, the value of the mood is $m = \frac{(n_2-1)(\sum_{i\in N_2} d_i - R) + n_2(\sum_{i\in N_1} d_i)}{n_2(\sum_{i\in N_2} d_i - R) + (n_2+1)(\sum_{i\in N_1} d_i)}$, and the mood value is $a_i^m = m d_i$ if $i \in N_1$ and $a_i^m = R - \sum_{j\neq i} d_j + m(\sum_{i\in N} d_i - R)$ if $i \in N_2$. When the available resource is $R\pm\varepsilon$ the mood and the mood value are, respectively: $\hat{m} = \frac{(n_2-1)(\sum_{i\in N_2} d_i - R\mp\varepsilon) + n_2(\sum_{i\in N_1} d_i)}{n_2(\sum_{i\in N_2} d_i - R\mp\varepsilon) + (n_2+1)(\sum_{i\in N_1} d_i)}$, $\hat{a}_i^m = \hat{m} d_i$ if $i \in N_1$, $\hat{a}_i^m = R\pm\varepsilon - \sum_{j\neq i} d_j + \hat{m}(\sum_{i\in N} d_i - R\mp\varepsilon)$ if $i \in N_2$. Called e the denominator of m and b the one of $\hat{m}$, the error is:

$$ERR_i = \begin{cases} \pm\sum_{i\in N_1} d_i \varepsilon d_i/(eb), & i \in N_1 \\ \pm\left(\frac{\sum_{i=1}^{n} d_i - R\mp\varepsilon}{b} + \frac{(\sum_{i=1}^{n} d_i - R)\sum_{i\in N_1} d_i}{eb}\right)\varepsilon, & i \in N_2 \end{cases} \tag{4.6}$$

- Case GM-MG
  Let $N = N_1 \cup N_2$ be partitioned into two disjoint sets N_1 and N_2 representing the set of user of type GM and the only one MG user, respectively. When the resource is R,

the value of the mood is $m = \frac{R-R+\sum_{i\in N_1} d_i}{\sum_{i\in N_1} d_i+R-R+\sum_{i\in N_1} d_i} = \frac{1}{2}$, and the mood value is $a_i^m = \frac{1}{2}d_i$ if $i \in N_1$ and $a_i^m = R - \sum_{i\in N_1} d_i + \frac{1}{2}(R-R+\sum_{i\in N_1} d_i) = R - \frac{1}{2}(\sum_{i\in N_1} d_i)$ if $i \in N_2$.
If the value of the available resource is $R \pm \varepsilon$, due to the hypothesis that we consider, only the minimum value for the user MG is changing. The value of the mood $\hat{m}$ is again equal to $\frac{1}{2}$ and the mood value is $\hat{a}_i^m = \frac{1}{2}d_i$ if $i \in N_1$ and $\hat{a}_i^m = R \pm \varepsilon - \frac{1}{2}(\sum_{i\in N_1} d_i)$ if $i \in N_2$. It follows:

$$ERR_i = \begin{cases} 0, & i \in N_1 \\ \pm\varepsilon, & i \in N_2. \end{cases} \tag{4.7}$$

Concerning the boundness of the error in case of the three allocation policy we can state the following proposition.

Proposition 4.1.1 If the allocation rule is proportional, MMF or mood value, the error to the users is less than or equal to ε.

Proof. The error boundness in case of proportional and MMF allocation is easily proof from the error formulas (4.1), (4.2). The mood value corresponds to the τ-value solution of bankruptcy games as proved in Chapter 3 (Theorem 3.2.2) and satisfies the monotonicity property as proved in [32]. We show that $|\hat{a}_i - a_i| \leq \varepsilon$. When the resource is $R+\varepsilon$ due to the monotonicity it holds:

$$a_i^m(R,d) \leq a_i^m(R+\varepsilon,d), \forall i \in N \tag{4.8}$$

and due to the efficiency it holds:

$$\sum_{i=1}^{n} a_i^m(R,d) = R, \sum_{i=1}^{n} a_i^m(R+\varepsilon,d) = R+\varepsilon \tag{4.9}$$

From (4.8) and (4.9) follows that $a_i^m(R+\varepsilon,d) - a_i^m(R,d) \leq \varepsilon, \forall i \in N$. In similar way when the resource is $R-\varepsilon$ due to the monotonicity it holds:

$$a_i^m(R-\varepsilon,d) \leq a_i^m(R,d), \forall i \in N \tag{4.10}$$

and due to the efficiency it holds:

$$\sum_{i=1}^{n} a_i^m(R,d) = R, \sum_{i=1}^{n} a_i^m(R-\varepsilon,d) = R-\varepsilon \tag{4.11}$$

Given (4.10), (4.11) then $a_i^m(R\text{-}\varepsilon,d) - a_i^m(R,d) \leq \varepsilon, \forall i \in N$. ■

When each user has the same misknowledge of the available resource (i.e., the same ε) equations (4.1)-(4.7) explain how the error ε is distributed among them. As already noticed, the error, for each allocation, is bounded by ε, i.e., the error is split between the users without anyone being severely disadvantaged. Furthermore, considering the fairness policy behind each error allocation, we can notice that it is close to the one of the resource allocation. In fact:

- the *weighted proportional allocation rule* splits the error proportionally to the users demands;
- the *MMF allocation* protects weak users, i.e., users with a smaller demand compared to the other users, not allocating them the error. No differences exist between the other users, receiving the same proportion of the error;

	a_i	S	$\hat{a}_i$	$\hat{PS}$	ΔPS $(\hat{PS}-PS)$
User 1	1.3333	0.66665	1.3335	0.66675	10^{-4}
User 2	4	0.6	4.0004	0.6	0
User 3	4.6667	0.53334	4.6671	0.53333	-10^{-5}

Table 4.2: Error on user satisfaction with the proportional allocation - a_i and $\hat{a}_i$ are the allocation when $R = 10$ and $R = 10+\varepsilon$, PS and $\hat{PS}$ are the satisfaction when the allocations are a_i and $\hat{a}_i$, $\varepsilon = 10^{-3}$, $d = (2,6,7)$.

- the *mood value* takes into account the nature of each user and of the others.

In particular the mood value in the GM case allocates the error as the proportional rule does; if the users are all of type GG, or all of type MM, it does not make difference between the user and that is a good property due to the fact that they have close demands; in the case of mixed users, it assigns the error considering the group to which a user belongs.

When we have misknowledge on the available resources, but the error is not equal among the users, interestingly (4.1)-(4.7) still provide the evaluation of the error for an user i, but clearly it depends on ε_i. We can notice that, for each allocation and for each group of user, again the error depends linearly on the value of the error, but compared to the case analyzed in which the error coincides for all the users the error ε is not shared between the users so that the sum of the users error is equal to ε. The coefficient of dependency varies between the users, taking into account the nature of the user, i.e., the absolute value of the demand and the demand compared to the other users. Because of the error of each user depends on different variables, we can not compare in general the allocations errors but from (4.1)-(4.7) we can see that the error is always limited by ε_i, so that each of the allocation considered does not strongly advantage/disadvantage an user.

4.1.2 Fairness considerations

We now look at the variation of the user satisfaction between the two scenarios with and without misknowledge on the available resource value using the three different allocation rules[1]. As we exhaustively explained in Chapter 3, the user satisfaction, when users can collect information about other users' demands and the available resource, has to be measured as $PS_i = \frac{a_i - min_i}{max_i - min_i}$ where min_i and max_i are equal to are the smallest and the biggest possible allocation for the user i. We can state the following:

Theorem 4.1.2 If each user has a full knowledge of the other user demands and the same misknowledge on the available resource ($R \pm \varepsilon$ instead of R), the mood value is the only scheme that:

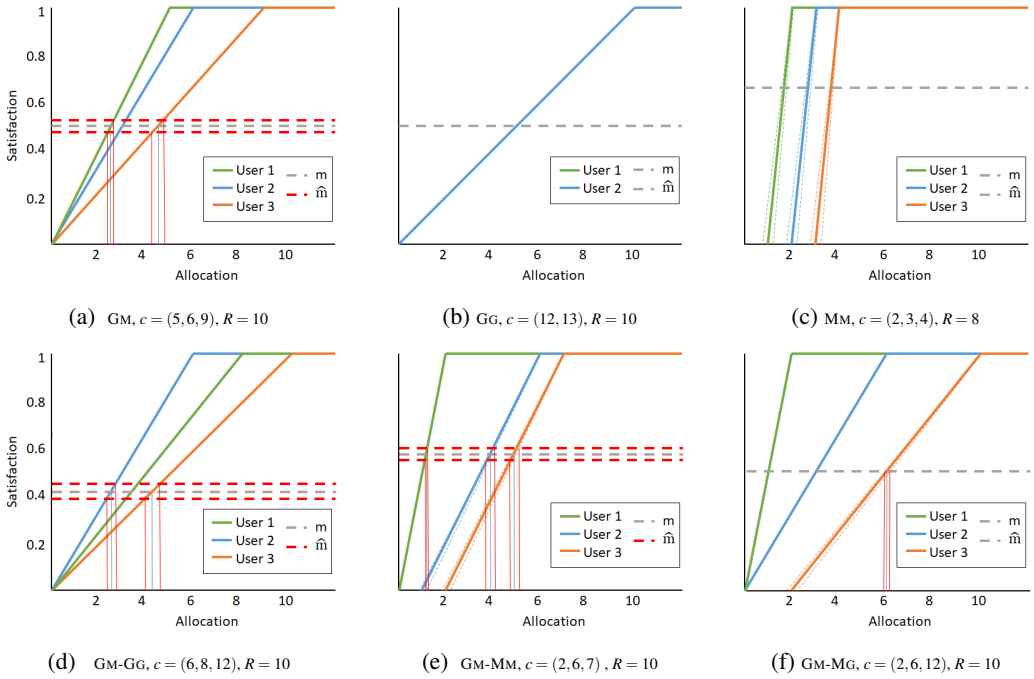
1. equalizes the satisfaction of the users,
2. equalizes the error on the user satisfaction.

Proof. The proof of the first part is in Chapter 3. Due to the fact that the value of the satisfaction for all the user is the same for both the case in which the resource is R and $R \pm \varepsilon$, the error on the satisfaction, i.e., the difference between the satisfaction in case without and with the misknowledge on the available resource, is the same. ■

[1] In this analysis we consider the fairness concept linked to the users satisfaction but other fairness properties can be analyzed, such as the envy-freeness or other generalized measure of fairness, not strictly linked to the concept of satisfaction [39] can be used.

	a_i^{MMF}	S	$\hat{a}_i^{MMF}$	$\hat{PS}$	$\Delta PS\ (\hat{PS} - PS)$
User 1	2	1	2	1	0
User 2	4	0.6	4.0005	0.60002	$2 \cdot 10^{-5}$
User 3	4	0.4	4.0005	0.39998	$-2 \cdot 10^{-5}$

Table 4.3: Error on user satisfaction with the MMF allocation - a_i and $\hat{a}_i$ are the allocation when $R = 10$ and $R = 10 + \varepsilon$, PS and $\hat{PS}$ are the satisfaction when the allocations are a_i and $\hat{a}_i$, $\varepsilon = 10^{-3}$, $d = (2,6,7)$.



(a) GM, $c = (5,6,9)$, $R = 10$ (b) GG, $c = (12,13)$, $R = 10$ (c) MM, $c = (2,3,4)$, $R = 8$

(d) GM-GG, $c = (6,8,12)$, $R = 10$ (e) GM-MM, $c = (2,6,7)$, $R = 10$ (f) GM-MG, $c = (2,6,12)$, $R = 10$

Figure 4.2: Users satisfaction with and without misknowledge on the available resource - mood value case.

Tables 4.2 and 4.3 show two counterexamples where proportional allocation and MMF allocation do not allocate the same satisfaction and the same error on the user satisfaction.

Figure 5.5 shows the variation of the satisfaction for the 6 possible cases. We can notice, as the theorem states, that the value of the satisfaction is the same for each user and for each value of R because it coincides with the mood m and $\hat{m}$. Furthermore, in Figures 4.2a, 4.2d, 4.2e, we clearly see that the gap between the satisfaction value when the resource is R (i.e., m) and when the resource is $\hat{R}$ (i.e., $\hat{m}$) is the same for each user. In addition to the two properties stated in the theorem, we can see that in Figures 4.2b, 4.2c, 4.2f the value of the satisfaction does not increase or decrease when we consider the error on the available resource. In this case the satisfaction of the users, called also mood, depends only by the number of users and not by the value of the demands. That are in fact situations (i) in which each single user has the same nature of the coalition of the other ones, or (ii) in which there is only one greedy user. As already explained, in the first case the the error is split uniformly between the users and in second one the greedy users keeps all the error. Another interesting fact is that the slope of the satisfaction line for users with smaller demands is not smaller than the one of users with bigger demands. This imply that

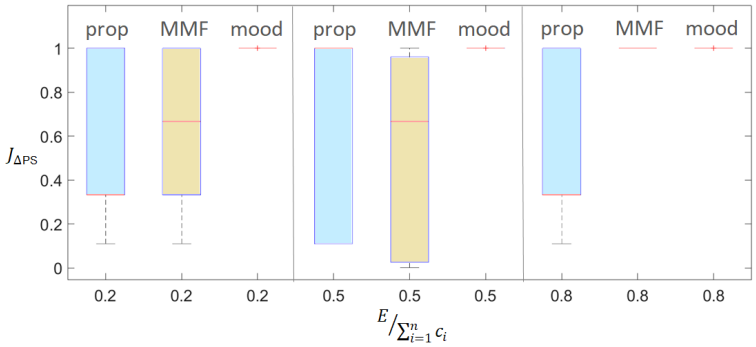


Figure 4.3: $J_{\Delta PS}$ for three ratio of available resource.

for these users the allocated error cannot be bigger than the other ones.

We now look at a global measure of fairness. We remind that the fairness is maximized using the mood value allocation, when we define the index as $J_{PS} = \left[\sum_{i=1}^{n}(PS_i)\right]^2 \Big/ n\sum_{i=1}^{n}(PS_i)^2$, where $PS_i = \frac{x_i - min_i}{max_i - min_i}$. We are then interested into evaluate the global fairness on the error. In particular we can re-define Jain index as follows:

$$J_{\Delta PS} = \left[\sum_{i=1}^{n}(\Delta PS_i)\right]^2 \Big/ n\sum_{i=1}^{n}(\Delta PS_i)^2 \tag{4.12}$$

where ΔPS_i is the difference of the satisfaction calculated when the resource is R and when the resource is $R \pm \varepsilon$. The redefinition of the index is necessary to evaluate the fairness on the satisfaction error. We can derive the following theorem.

Theorem 4.1.3 J_{PS} and $J_{\Delta PS}$ are maximized when the resource allocation is based on the mood value.

Proof. The maximization of J_{PS} happens when each user receives the same satisfaction PS_i, and the maximization of $J_{\Delta PS}$ happens when each user receives the same ΔPS_i (see chapter 3). From Theorem 5.3.1, it follows that the mood value is the allocation that maximizes the two indices of fairness. ■

We now test the behavior of the three considered allocation schemes in term of $J_{\Delta PS}$ by simulating 100 resource allocation problems with random demands belonging to [1,10] while varying the value of R. We set the error equal to 10^{-2}. Fig. 4.3 shows the boxplot of $J_{\Delta PS}$ when ρ is 0.2, 0.5 and 0.8. We note that the mood value maximizes the fairness on the satisfaction error. In fact, the index takes value in $[0,1]$, and the higher is its value, the higher is the fairness. The proportional and the MMF allocation can produce inequality between users, presenting median values different from one and high variability.

In summary, the resource allocation that users in the analyzed inaccurate information scenario prefer is the mood value allocation because:

- it equalizes the user satisfaction and the satisfaction error and it maximizes the Jain index of fairness on the allocation and on the error;
- it takes into account the user nature in error splitting;
- in some cases, it allocates a portion of resource that provides exactly the expected satisfaction.

	Allocation rule	ERR_i
	weighted proportional	$\mp\frac{(n-1)d_i\delta R}{(\sum_{i=1}^n d_i \pm (n-1)\delta)(\sum_{i=1}^n d_i)}$
	MMF	$\begin{cases} 0, & i=1,..,k \\ \frac{\mp k\delta}{n-k}, & i=k+1,...,n \end{cases}$
Mood value	GM	$\mp\frac{(n-1)d_i\delta R}{(\sum_{i=1}^n d_i \pm (n-1)\delta)(\sum_{i=1}^n d_i)}$
	GG	0
	MM	$\mp\frac{n-1}{n}\delta$
	GM-GG	$\begin{cases} \mp\frac{\delta(n_1-1)Rd_i}{((n_1-1)\delta+c)c}, & i\in N_1 \\ \mp\frac{\delta n_1 R^2}{(n_1\delta+c)c}, & i\in N_2 \end{cases}$
	GM-MM	$\begin{cases} \mp\frac{[n_2 b+f(1-n_1)]\delta}{e(e\pm n_2^2\delta\pm(n_2+1)(n_1-1)\delta)}, & i\in N_1 \\ \pm\frac{[b(3n_2-1-2n_2^2)-n_1 f]\delta}{e(e\pm n_2(n_2-1)\delta\pm(n_2+1)n_1\delta)}, & i\in N_2 \end{cases}$
	GM-MG	$\begin{cases} 0, & i\in N_1 \\ \mp\frac{(n-1)}{2}\delta, & i\in N_2 \end{cases}$

Table 4.4: Evaluation errors with misknowledge on the users demands. N_1=set of users GM, N_2=set of users of the other type. $\sum_{i\in N_2} d_i - R = f$, $\sum_{i\in N_1} d_i = b$, $\sum_{i\in N_1} d_i + n_2 R = c$, $e = n_2 f + (n_2+1)b$.

4.2 Error on the users demand

4.2.1 Error estimate

In this section we analyze the error on the allocation when the error is on the users demand. For each user $i \in N$ the error ERR_i is defined as $|\hat{a}_i - a_i|$, where a_i is the share obtained by i when the demand vector is d and $\hat{a}_i$ is the allocation of user i when each user $j \neq i$ has a demand of $d_j \pm \delta$. Table 4.4 summarizes the value of the error, when we consider the weighted proportional allocation rule, the MMF allocation and the mood value. The errors are calculated as follows:

Weighted proportional allocation

The allocation for user i is $a_i^p = d_i R / \sum_{i=1}^n d_i$ when the demand vector is is c. When the other users demand is $d_j \pm \delta$ then the allocation is $\hat{a}_i^p = d_i R / (\sum_{i=1}^n d_i \pm (n-1)\delta)$. The error on the allocation is for user i is:

$$ERR_i = \mp\frac{(n-1)d_i\delta R}{(\sum_{i=1}^n d_i \pm (n-1)\delta)(\sum_{i=1}^n d_i)} \tag{4.13}$$

Since $\frac{d_i}{\sum_{i\in N} d_i \pm (n-1)\delta}$ and $\frac{R}{\sum_{i\in N} d_i}$ are less than 1, the error is limited by $(n-1)\delta$.

MMF allocation

As done in the previous scenario, we consider the hypothesis that δ is enough small to not change the nature of the user. This means that if the users i is in the first k users with

smaller demands, it should have an allocation equal to d_i also when the error is considered. On contrary, in the other case, it receives $\frac{R-\sum_{i=1..k}(d_i\pm\delta)}{n-k}$.The error is:

$$ERR_i = \begin{cases} 0, & \text{if } i = 1,..,k \\ \frac{\mp k\delta}{n-k}, & \text{if } i = k+1,...,n. \end{cases} \tag{4.14}$$

The error can be greater than δ but it is limited because it is bounded by $k\delta$.

Mood value

- Case GM
  This case coincides with the weighted proportional allocation. The error is given by (4.13).
- Case GG
  In this case it holds that $min_i = 0$ and $max_i = R$ for each user i. It follows that, even if the information received about other users demands is not correct, the allocation is exactly the one expected from the users: $a_i = \frac{R}{n}$. Thus the error is:

$$ERR_i = 0 \tag{4.15}$$

- Case MM
  In this case, if the demand vector is c the value of the mood is $m = \frac{R-n(R)+(n-1)\sum_{i=1}^{n} d_i}{\sum_{i=1}^{n} d_i - n(R)+(n-1)\sum_{i=1}^{n} d_i} = \frac{n-1}{n}$, and the mood value is $a_i^m = R - \sum_{j\neq i} d_j + \frac{n-1}{n}(\sum_{i=1}^{n} d_i - R)$. If we introduce an error on the demand vector, the value of the mood is $\hat{m}$ is given by: $\hat{m} = \frac{R-n(R)+(n-1)\sum_{i=1}^{n} d_i\pm(n-1)(n-2)\delta+(n-1)\delta}{\sum_{i=1}^{n} d_i\pm(n-1)\delta-nR+(n-1)\sum_{i=1}^{n} d_i\pm(n-1)(n-2)\delta+(n-1)\delta} = \frac{n-1}{n}$ and the mood value is $\hat{a}_i^m = R - \sum_{j\neq i} d_j \mp (n-1)\delta + \frac{n-1}{n}(\sum_{i=1}^{n} d_i - R \pm (n-1)\delta)$. It follows that for each user i the error is:

$$ERR_i = \mp\frac{n-1}{n}\delta \tag{4.16}$$

  and bounded by δ.
- Case GM-GG
  Let $N = N_1 \cup N_2$ be partitioned into two disjoint sets N_1 and N_2 representing the set of user of type GM and GG, respectively. The value of the mood is $m = \frac{R}{\sum_{i\in N_1} d_i + n_2 R}$, and the mood value is $a_i^m = \frac{R}{\sum_{i\in N_1} d_i + n_2 R} d_i$ if $i \in N_1$ and $a_i^m = \frac{R^2}{\sum_{i\in N_1} d_i + n_2 R}$ if $i \in N_2$. If there is a misknowledge of the demand vector and $i \in N_1$, the value of the mood is $\hat{m} = \frac{R}{\sum_{i\in N_1} d_i + (n_1-1)\delta + n_2 R}$, and the mood value is $\hat{a}_i^m = \frac{R}{\sum_{i\in N_1} d_i + (n_1-1)\delta + n_2 R} d_i$. If $i \in N_2$, the mood is $\hat{m} = \frac{R}{\sum_{i\in N_1} d_i + n_1\delta + n_2 R}$, and the mood value is $\hat{a}_i^m = \frac{R^2}{\sum_{i\in N_1} d_i + n_1\delta + n_2 R}$. Called

$\sum_{i\in N_1} d_i + n_2 R = c$, it follows that:

$$ERR_i = \begin{cases} \mp\frac{\delta(n_1-1)Rd_i}{((n_1-1)\delta+c)c}, & i \in N_1 \\ \mp\frac{\delta n_1 R^2}{(n_1\delta+c)c}, & i \in N_2 \end{cases} \tag{4.17}$$

For users GM, since $\frac{R}{\sum_{i\in L} d_i+(l-1)\delta+kR}$ and $\frac{d_i}{\sum_{i\in L} d_i+kR}$ are less than one, the error is inferior to $(l-1)\delta$. For users GG, since $\frac{R}{\sum_{i\in L} d_i+l\delta+kR}$ and $\frac{R}{\sum_{i\in L} d_i+kR}$ are less than one, the error is inferior to $l\delta$.

- Case GM-MM
  Let $N = N_1 \cup N_2$ be partitioned into two disjoint sets N_1 and N_2 representing the set of user of type GM and MM, respectively. When the demand vector is d, the value of the mood is $\frac{(n_2-1)f+n_2 b}{n_2 f+(n_2+1)b}$, where $\sum_{i\in N_2} d_i - R = f$ and $\sum_{i\in N_1} d_i = b$. The mood value is $a_i^m = md_i$ if $i \in N_1$ and $a_i^m = R - \sum_{j\neq i} d_j + m(d_i - R + \sum_{j\neq i} d_j)$ if $i \in N_2$. If there is a misknowledge of d and $i \in N_1$ the mood is $\hat{m} = \frac{(n_2-1)(f\pm n_2\delta)+n_2(b\pm(n_1-1)\delta)}{n_2(f\pm n_2\delta)+(n_2+1)(b\pm(n_1-1)\delta)}$ and the mood value is $\hat{a}_i^m = \hat{m}_1 d_i$. If there is a misknowledge of d and $i \in N_2$ the mood is $\hat{m} = \frac{(n_2-1)(f\pm(n_2-1)\delta)+n_2(b\pm n_1\delta)}{n_2(f\pm(n_2-1)\delta)+(n_2+1)(b\pm n_1\delta)}$ and the mood value is $\hat{a}_i^m = R - \sum_{j\neq i}(d_j \pm \delta) + \hat{m}_2(d_i - R + \sum_{j\neq i}(d_j \pm \delta))$. It follows that the error is:

$$ERR_i = \begin{cases} \mp\frac{[n_2 b+f(1-n_1)]\delta}{e(e\pm n_2^2\delta\pm(n_2+1)(n_1-1)\delta)}, & i \in N_1 \\ \pm\frac{[b(3n_2-1-2n_2^2)-n_1 f]\delta}{e(e\pm n_2(n_2-1)\delta\pm(n_2+1)n_1\delta)}, & i \in N_2 \end{cases} \tag{4.18}$$

  where $e = n_2 f + (n_2+1)b$.
- Case GM-MG
  Let $N = N_1 \cup N_2$ be partitioned into two disjoint sets N_1 and N_2 representing the set of user of type GM and MG, respectively. When the demand vector is R, the value of the mood is $m = \frac{1}{2}$, and the mood value is: $a_i^m = \frac{1}{2}d_i$ if $i \in N_1$ and $a_i^m = R - \frac{1}{2}(\sum_{i\in N_1} d_i)$ if $i \in N_2$. The value of the mood $\hat{m}$, when the user knows the demand vector with a small error is again equal to $\frac{1}{2}$, both in the case in which user i is of type GM or MG. The mood value is $\hat{a}_i^m = \frac{1}{2}d_i$ if $i \in N_1$ and $\hat{a}_i^m = R - \sum_{i\in N_1} d_i \pm (n-1)\delta + \frac{1}{2}(R - R + \sum_{i\in N_1} d_i \pm (n-1)\delta) = R - \frac{1}{2}(\sum_{i\in N_1} d_i \pm (n-1)\delta)$ if $i \in N_2$. It follows:

$$ERR_i = \begin{cases} 0, & i \in N_1 \\ \mp\frac{(n-1)}{2}\delta, & i \in N_2. \end{cases} \tag{4.19}$$

In contrast from the first analyzed case, when there is an error on the users demand the error is not always limited by δ and most of the time increases with the number of users. For example, in the MMF case, the error can be greater than δ but bounded by $k\delta$. Furthermore, we can notice one more time that the mood value assigns an error that depends on the nature of the problem: it assigns the same error to users belonging to cases GG and MM, while it differentiates users belonging the group GM.

Scenario	User type	PS	$\hat{PS}$
GM	GM	$\frac{R}{\sum_{i=1}^{n} d_i}$	$\frac{R}{\sum_{i=1}^{n} d_i \pm (n-1)\delta}$
GG	GG	$1/n$	$1/n$
MM	MM	$(n-1)/n$	$(n-1)/n$
GM-GG	GM	$\frac{R}{\sum_{i\in N_1} d_i + n_2 R}$	$\frac{R}{\sum_{i\in N_1} d_i + n_2 R \pm (n_1-1)\delta}$
	GG	$\frac{R}{\sum_{i\in N_1} d_i + n_2 R}$	$\frac{R}{\sum_{i\in N_1} d_i + n_2 R \pm n_1 \delta}$
GM-MM	GM	$\frac{(n_2-1)f+n_2 b}{n_2 f+(n_2+1)b}$	$\frac{(n_2-1)(f\pm n_2\delta)+n_2(b\pm(n_1-1)\delta)}{n_2(f\pm n_2\delta)+(n_2+1)(b\pm(n_1-1)\delta)}$
	MM	$\frac{(n_2-1)e+n_2 b}{n_2 e+(n_2+1)b}$	$\frac{(n_2-1)(e\pm(n_2-1)\delta)+n_2(b\pm n_1\delta)}{n_2(e\pm(n_2-1)\delta)+(n_2+1)(b\pm n_1\delta)}$
GM-MG	GM	$1/2$	$1/2$
	MG	$1/2$	$1/2$

Table 4.5: Evaluation of $\hat{PS}$ and PS in case of full knowledge of the available resource and the same misknowledge on the other users demand. N_1=set of users GM, N_2=set of users of the other type. $\sum_{i\in N_2} d_i - R = f$, $\sum_{i\in N_1} d_i = b$.

4.2.2 Fairness considerations

Looking at the satisfaction PS, we can state the following theorem for the scenario in which there is an error on the users demands.

Theorem 4.2.1 If each user has a full knowledge of the available resource and the same misknowledge on the other users demand, the mood value is the only scheme that equalizes the error on the satisfaction for the same type of user.

Proof. We calculate the value of the ΔS in each case for each type of user. We report the evaluation in Table 4.5, where we can see that for users of same type the error on the satisfaction, i.e., ΔPS $(\hat{PS} - PS)$ is the same. ■

From Table 4.5 we can also see that in some resource allocation problem types (e.g. GG, MM,GM-MG), as it was happening in case of misknowledge on the available resource, the users receive an allocation that satisfies them at the same level of the case of complete information.

4.3 Summary

In this chapter, we analyze resource allocation with inaccurate information sharing, providing an estimation of the error on the resource allocation. We then present three theorems showing how the mood value allocation is superior to the proportional and MMF allocation in terms of fairness.

In the previous and current chapter we focus the analysis on the single-resource allocation problem, formalizing an appropriate measure of fairness and resource allocation for the complete information sharing and the inaccurate information sharing context. In the

following chapter, we deal with the multi-resource allocation problem, providing a general framework where the decision-making can select the most appropriate allocation, based on the fairness goal it wants to follow. This framework is suitable for the network slicing resource allocation problem, where a set of heterogeneous resources has to be provided to each tenant.

5. Multi-resource allocation for network slicing

In this chapter we analyze the allocation of multi resources in the network slicing. In particular, we address the following research questions: are the multiple resources called by a slice to be allocated one after the other independently of each other, or shall one take the multi-resource allocation as a joint allocation problem to increase system efficiency? If the request for at least one resource is bigger than the available one, we revisit how fairness in resource usage can be measured, and ensured by means of resource allocation algorithms. We propose a unified mathematical framework able to generalize some of the classical solutions for single and multi-resource allocation problems from the literature. This framework takes into account both user satisfaction and system efficiency objectives, meeting different degrees of fairness. Moreover, we compare multiple allocation rules and evaluate them in terms of wasted resource (i.e., resource allocated but eventually not used) and idle resource (i.e., resource left available for future allocations), running evaluations against the network slicing use-case.

5.1 Network slicing

While the fourth generation (4G) of networks was designed for improving the smartphone experience mostly in terms of network throughput, the fifth generation (5G) is instead being designed with a much broader goal. 5G networks need to provide end-to-end connectivity, directly supporting verticals, including radio connectivity, wired connectivity and computing resource delivery and orchestration, exploiting system and network virtualization technologies [60]. 5G verticals include, e.g., e-health services, public safety systems, smart office, and connected vehicles, trains and aircrafts [61]. According to the International Telecommunication Union (ITU) services are classified in three categories [62]: enhanced mobile broadband (eMBB), Ultra Reliable Low Latency Communications (URLLC) and massive machine type communications (mMTC) - characterized by bandwidth, latency,

The work presented in this chapter is partially presented in [58] and [59]

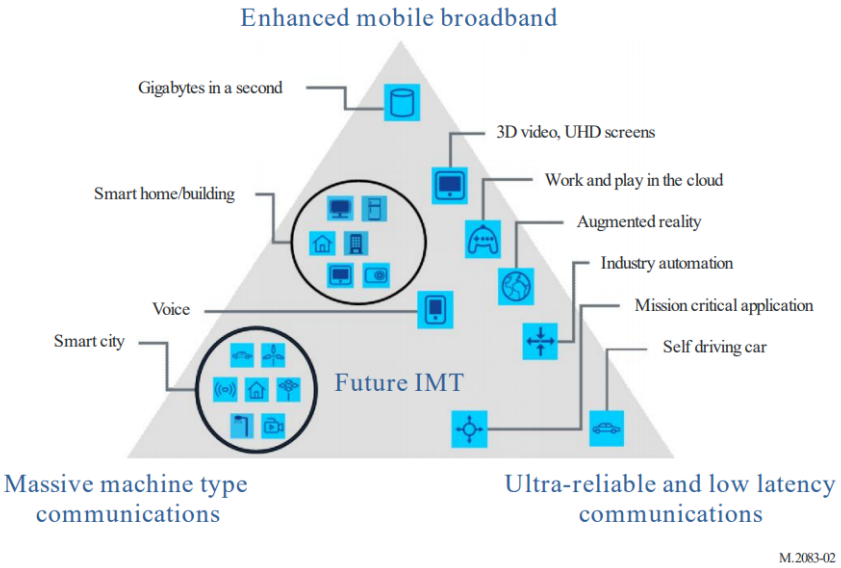


Figure 5.1: Usage scenarios of International Mobile Telecommunications for 2020 and beyond (source [62])

frequency and reliability requirements (see Figure 5.1).

Enhanced Mobile Broadband (eMBB) services are the evolution of 4G LTE enabled services toward high speed connections with high data rate supporting, they include 3D and ultra-high-definition video transmission and virtual reality applications; massive Machine Type Communication (mMTC) services are characterized by very high density of connected devices typically transmitting at low data rate, they satisfy requirements of sensor networks employed in smart cities, Internet of Things (IoT) and wearable device networks; Ultra-Reliable Low Latency Communications (URLLC) services require high reliability and low latency as it is necessary for wireless control of industrial manufacturing or production processes, transport safety control, remote medical surgery services, etc.

The provisioning abstraction being formalized by 5G activities is the so-called 'network slice' [60]. A network slice is an independent and logically-isolated end-to-end virtual network running on a shared physical infrastructure aiming to provide the customers required service or vertical corresponding to different business domains. It follows that a network slice spans different parts of the network as the access, transport, core and data-center segments, combining networking, computing and storage programmable resources [63]. The interest towards network slicing is motivated by the increasing programmability of the Radio Access Networks (RANs) and of the core elements, also thanks to the novel technologies such as Software-Defined Networking (SDN) and Network Function Virtualization (NFV) [60].

Provisioning resources along an end-to-end path is therefore a multi-resource allocation problem (Figure 5.2). In the literature, different multi-resource allocation techniques targeting forms of fairness are proposed (see Section 2.4) and recent works address resource allocation problem in network slicing from many point of view. They use different approaches to model and solve the resource allocation problem in network slicing. Different perspectives are adopted, considering various resource types, alternative mathematical tools and different objectives.

A recurrent approach is to integrate multi-resource considerations within the Virtual

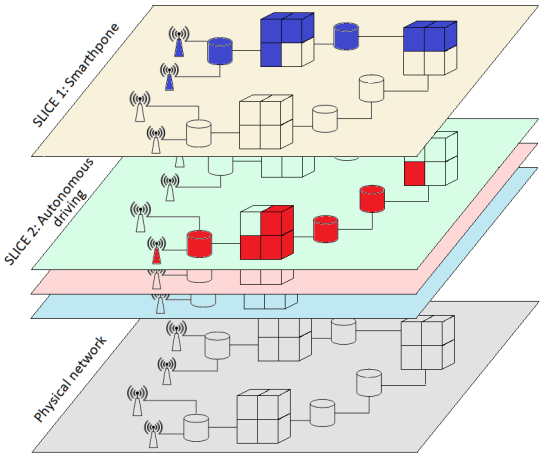


Figure 5.2: A representation of network slices and resource sharing.

Network Function (VNF) placement algorithm. For example, authors in [64] address the slicing of radio access network proposing a multi-operator resource allocation rule, able to assign to each user a single base station and able to quantify a fair portion of the resource to assign to each user. Similarly, authors in [65] model the infrastructure with a direct graph and the slice as a simple source-destination pair, solving both the placement and the resource allocation as a unique problem. In [66] a model to place VNFs while selecting links is proposed using a complex network analysis.

Another approach that can be found in the literature is the one concerning the maximization of the slice costumer's profit [67] or the one considering the network revenue [68].

Modeling the problem as a competition between tenants sharing the same infrastructure, it is also possible to adopt a game-theoretic approach. In [69], authors propose a network slicing game in which users react to other tenants allocation and maximize their utility, converging to a Nash equilibrium.

Distributed approaches are used in [70], where a cooperative game is introduced that, to avoid revealing mobile operators private information, uses a distributed algorithm to solve the allocation problem. Instead, in [71], both collaborative and non-collaborative approaches are analyzed and solved, using auctions between slices and datacenter providers for the former one, and a distributed approach for the latter one.

The approach of this chapter differs from the above mentioned ones in several aspects:

- we take into consideration only the problem of multi-resource allocation producing a solution giving an amount of each resource to allocate to each tenant, independently of the infrastructure, whereas the actual embedding of each resource into a final resource partitioning – taking into consideration the geographical distribution and interconnection links of computing servers – is considered as a separate, successive, problem;
- in our network model tenants express a demand for each resource and there is an actual problem when there is at least one congested resource, i.e., at least one resource cannot satisfy all the tenants;
- we consider resource dependency between resources, as done for instance in [65] (and elaborated hereafter);
- the possible allocation rules we propose span different concepts of fairness, namely, considering or not the awareness of the tenant with respect to the available resource

and the other tenants demands.

5.1.1 Resource dependency and depletion

Virtualized network systems are evolving so that network functions nodes can be given computing power elastically and as a function of the load (i.e., virtual link bitrate), and that the spectrum allocated via medium access protocols can be flexibly adapted to the requested bitrate. There is indeed a dependency among different types of resources in such systems 5G networks leverage on. For example, for the computing resource to traffic bit-rate dependency, it is typically a linear [45, 65, 72, 73] or step-linear or piece-wise linear relationship with few deflection points, as seen in [11, 74]; for the bitrate to spectrum one, a step-linear relationship can be inferred from slice specifications such as [75]. Taking such a behavior into account in network models is challenging. In the model proposed in this chapter, we assume a linear relationship that can provide a good approximation to such step and piece-wise linear relationships.

In our analysis, we consider two aspects to assess an allocation solution when some resources are not enough to fully satisfy tenants' requests. Firstly, each slice demand expresses an inter-resource linear relationship that has to be satisfied; e.g., the number of cores for a virtual machine in a slice can vary as a function of the bitrate and hence the link bandwidth allocated to the slice – i.e., one core needed every given amount of traffic: hence if less traffic is granted, a number or a proportion of core capacity can be saved. We refer to this aspect as *inter-resource dependency*, which can lead to wasted resource, i.e., allocated but not useful resource[1]. Secondly, we consider the *resource depletion*: a resource is depleted if it is fully distributed to slices. In the case of a single-resource system one aims at fully allocating the resource in order to provide an efficient solution, i.e., the resource is depleted, there is no idle resource left. In a multi-resource context, a multi-resource allocation rule taking into consideration inter-resource dependency can lead to idle resource, i.e., the resources may not be depleted. In [76] the resource depletion is measured as distance from the system efficiency obtained when all the resource available is distributed to users.

Fig. 5.3 depicts a basic resource allocation problem example with 2 users and 2 resources, representing in a bi-dimensional space (i.e., the resource space) the users demand and the available resource. A single-resource approach considers a number of problems equal to the number of resources needed by the slice, producing allocations that do not take into consideration resource dependency (linear in the figure). In fact, we can notice that for both the users a portion of resource is allocated even if it cannot be used by the tenant. Contrarily, with a multi-resource approach, resources and demands are multi-dimensional and take into account the resource dependency. A multi-resource allocation rule may create idle resource, hence respecting resource dependency while meeting allocation goals such as fairness.

[1] Under the hypothesis of linear dependency of resources, if a user decreases its demand for one resource, automatically it decreases its demand for all the other resources. The wasted resource can be a problem from both the user and the provider point of view. In fact the first one pays for a resource that is not able to use while the second is providing a resource that is not used and that could be held back for itself to serve someone else. The waste of resource is automatically nullified when we consider a multi-resource approach respecting the linear relationship. So the proposed approach is able to meet two objectives: to avoid resource waste while ensuring fairness.

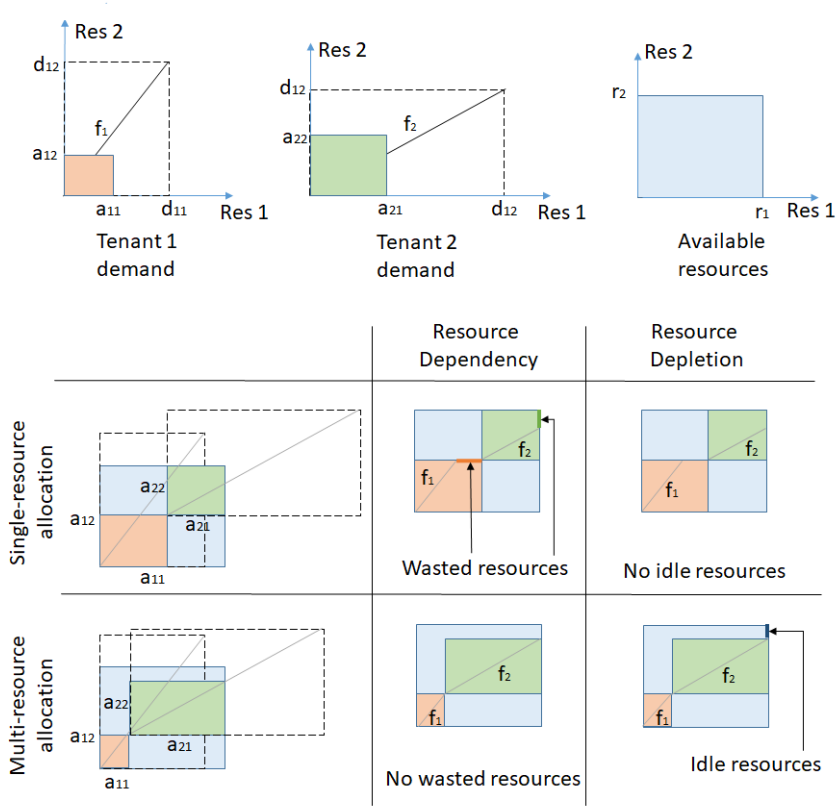


Figure 5.3: Behavior of single and multi-resource allocations in terms of inter-resource dependency and resource depletion. f_i, $i = 1,2$, is the relation between the resources for tenant i, whereas d_{ij} is the demand of the i^{th} user for the j^{th} resource and r_j, $j = 1,2$, is the available amount for the j^{th} resource. a_{ij} is the allocation of the i^{th} user for the j^{th} resource. The horizontal axis represents the resource 1, the vertical one the resource 2.

5.2 MUlti-Resource Allocation for NEtwork Slicing (MURANES)

In order to solve the network slicing resource allocation problem we propose a framework based on an aggregation technique we name MURANES (MUlti-Resource Allocation for NEtwork Slicing). Our objective is to propose a general framework to allocate multiple distinct resources. In this direction we consider two factors: an individual satisfaction of the tenants and a system efficiency utility. The main idea underlying our approach is depicted in Figure 5.4. We need to consider a utility function $F(y)$ that summarizes the information about users demands and the available resources. To obtain this function we can follow two methodological 'paths':

- we firstly aggregate the users, and then the resources;
- we firstly aggregate the resources, and then the users.

In network slicing, an important requirement is to provide a fair allocation matrix, thus it is necessary that the input vector of the function to optimize summarizes the information related to the user satisfaction. For this reason we follow the second path, depicted with a red arrow in the figure, aggregating firstly the information related to the different resources for each user, i.e., considering the user satisfaction, and secondly aggregating the users, i.e., considering the system efficiency objective. In particular, we propose to use as aggregation function F the Ordered Weighted Averaging (OWA) operator [77], that is detailed in the

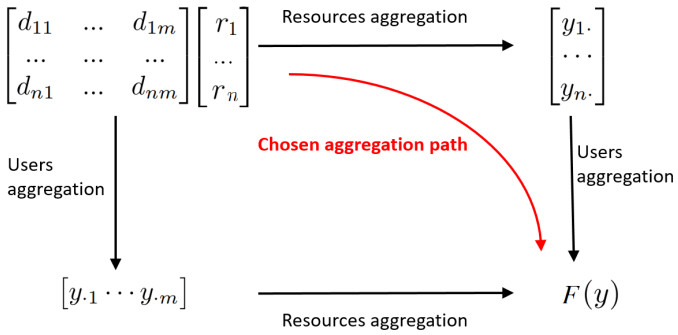


Figure 5.4: User and resource aggregation paths. The vector $y_{i\cdot}$ combines m data to provide a single aggregated variable for each user. The vector $y_{\cdot i}$ combines n data to provide a single aggregated variable for each resource. $F(y)$ is the aggregated function to optimize.

following subsection[2].

In the following, after explaining the OWA operator and its flexibility in covering different objectives, we discuss how to properly select OWA input vectors, related to different users satisfaction concepts. The OWA framework we formalize can so incorporate some of the existing multi-resource allocations rules, and permits also to transpose some of the existing single-resource allocation rules to the multi-resource context.

5.2.1 Ordered Weighted Averaging (OWA) operators

The Ordered Weighted Averagin (OWA) function is introduced in [77] and it is defined as follows.

Definition 5.2.1 An *OWA* is a scalarizing function $F : \mathbb{R}^n \to \mathbb{R}$ parametrized by a weighting vector $w \in {\mathbb{R}_+}^n$ of the form $F(v_1, ...v_n) = \sum_{j=1}^{n} w_j v_{(j)}$, where $v_{(j)}$ is the j-th smallest element of $(v_1, ..., v_n)$.

Contrary to the case of the weighted sum, the weights in an OWA function are not used to assign more importance to a component than to another one, but to control the importance attached to good or bad components in the value aggregation process.

The F aggregator encompasses many well known aggregators such as max, min, median and sum, as special cases. It is well known in the Social Choice area, to model an idea of fairness in the social evaluation function. In this case, F is often referred to as the *Generalized Gini social-evaluation Function* [78, 79]. Such a function used with a decreasing weighting vector w, i.e., $w_i \geq w_{i+1}$ for all $i < n$ allows to model a wide range of 'fair' attitudes going from the egalitarianism to the utilitarianism. An *egalitarian* solution is based on the notion of fairness, described in political philosophy by Rawls [17] aiming to protect weaker users, i.e., the less satisfied ones. It is obtained when we maximize the minimum component so when we choose only the first weight w_1 different from zero. An *utilitarian* solution, under the classical utilitarian principle, is obtained when the decision maker maximizes the sum of the utilities of the players. It is obtained choosing the same value for each weight ($w_i = w_j, \forall i \neq j$). Changing the OWA weights, choosing decreasing value of the weight, we can obtain trade-off solutions between egalitarianism and utilitarianism.

[2] In the case we consider users with different priority we can use the WOWA operator, a generalization of the OWA operator, where we can attach to different users a weight depending on its priority.

More precisely, one common way of formally introducing a fairness property in the aggregation is to require that the value of a vector is improved by any *mean preserving transfer reducing inequalities* (a.k.a. Pigou-Dalton transfers) [80]. Given a performance vector $v = (v_1, \ldots, v_n)$, any modification of v leading to a vector of the form $(v_1, \ldots, v_i - \varepsilon, \ldots, v_j + \varepsilon, \ldots, v_n)$ for some i, j, ε such that $v_i - v_j > \varepsilon > 0$ should make decision maker better off. Under the Pareto principle – requiring monotonicity in every component – and some other mild requirements such as completeness – requiring this fairness condition – it is possible to define the social utility as an OWA function using a weighting vector w with decreasing components. The described potential of F is illustrated in the following:

■ **Example 5.1** Consider a simple case with three users. A solution with utility vector (1, 0, .3) is less preferable than (.5, .5, .3) because there exists a transfer (-.5,+.5) between the two first agents to pass from the former solution to the latter. Consistently, we have $F(1,0,.3) = .3w_2 + w_3$ and $F(.5,.5,.3) = .3w_1 + .5w_2 + .5w_3$ and therefore $F(1,0,.3) - F(.5,.5,.3) = .3(w_2 - w_1) + .5(w_3 - w_2) \leq 0$ because $w_1 \geq w_2 \geq w_3$. We obtain the desired preference. Now if we compare $(1,0,.5)$ to $(.3,.3,5)$ the preference is less clear. In particular, no Pigou-Dalton transfer holds. Moreover, in such a situation, one may want to relax the desire of equity to hold average efficiency. Consistently, we have $F(1,0,.3) - F(.3,.3,.3) = .7w_3 - .3w_1$ which may be positive or negative depending on w given that $w_1 \geq w_3$. This illustrates the role of vector w that can lead to different choices depending on the importance attached to the least satisfied users. ■

The F function is also widely used in multi-objective optimization to generate solutions with well-balanced utility profiles [81, 82]. $F(v)$ is not linear in v due to the permutation of components, but smart linearization are available, see, e.g., [81].

The MURANES framework we propose is based on the optimization of OWA operators. It is designed for continuous resources, i.e., resources that can be partitioned indefinitely but that – with straightforward model variations – can also be applied to the case of discrete resources, or to the case in which the allocation must be selected from prefixed templates.

5.2.2 The general framework

As above introduced, the framework we propose is considering two axes: the system and the individual utility. About the former, subsection 5.2.1 shows that the maximization of an OWA function is a good candidate to obtain fair allocations, where fairness goes from the pure egalitarianism to the pure utilitarianism. About the latter, as we anticipated, the input vector of the OWA must depend on the user satisfaction vector, i.e., a vector containing the measure of the satisfaction of each user respect to the m resources. We describe now the four proposed inputs:

- classical satisfaction: Classically the satisfaction is measured as the percentage of resource allocated to a user, i.e., as the ratio between the allocated resource and the demanded one. In our model, for each user this ratio is the same for each resource and it is equal to x.
- weighted classical satisfaction: We can consider a weighted version of the classical satisfaction. Taking inspiration from the DRF allocation rule the satisfaction of each user i we choose a weight equal to the dominant share (i.e., $ds_i = \max_j\{\frac{d_{ij}}{r_j}\}$).
- player satisfaction (ps): As already explained in Chapter 3, in case of complete information, the correct way to measure the satisfaction is using the ps rate. If in the case of the classical satisfaction, given a user, the satisfaction coincides for each resource, here we need to find which satisfaction summarizes the information about

		System		
		w=$(1,0,\cdots,0)$	$\cdots$	w=$(1,1,\cdots,1)$
Individual	x	$\max\min x_i$	$\cdots$	$\max\sum_{i=1}^{n} x_i$
	$ds\cdot x$	$\max\min ds_i x_i$	$\cdots$	$\max\sum_{i=1}^{n} ds_i x_i$
	ps	$\max\min ps_i$	$\cdots$	$\max\sum_{i=1}^{n} ps_i$
	$ds\cdot ps$	$\max\min ds_i ps_i$	$\cdots$	$\max\sum_{i=1}^{n} ds_i ps_i$

Table 5.1: Objective function of the MURANES framework.

all the resources. For this purpose, we use the dominant resource for each tenant, because it is the more critical one and, realistically, the one that the tenant would consider to measure its satisfaction.

- weighted player satisfaction: in a dual way to the classical satisfaction, we can again consider the dominant share to weight the ps satisfaction.

The general problem to solve is[3]:

$$\begin{aligned} &\text{maximize} && OWA(v) \\ &\text{subject to} && x \in \mathscr{F} \\ & && 0 \le x_i \le 1, \forall i \in N \end{aligned} \tag{5.1}$$

where $\mathscr{F}$ is the admissible region s.t. $\sum_{i\in N} a_{ij} \le r_j$, $\forall j \in M$ and v can be equal to: (i) the vector x, (ii) the vector $ds\cdot x = \begin{bmatrix} ds_1\cdot x_1 & \dots & ds_n\cdot x_n \end{bmatrix}$, (iii) the vector ps, with the satisfaction calculated for each user respect to the dominant resource or (vi) the vector $ds\cdot ps = \begin{bmatrix} ds_1\cdot ps_1 & \dots & ds_n\cdot ps_n \end{bmatrix}$. We summarize in Table 5.1 the value objective function in the general framework we propose for two extreme OWA weights configurations.

5.3 MURANES properties

We describe some important properties of the allocations we obtain using the MURANES framework.

5.3.1 Generalization of well known-solutions

The unified framework uses a general class of utility functions that captures different fairness criteria, and between them we can find some already well-known ones. In fact for special combinations of OWA inputs and weights, the allocation coincides with an allocation known in literature. We can state the following theorems.

Theorem 5.3.1 The MURANES framework with $w = (1,0,\cdots,0)$ and input x generalizes to the multi-resource context the weighted proportional allocation rule.

Proof. The MURANES in case in which $w = (1,0,\cdots,0)$ and the input is x coincides with the solution of:

$$\begin{aligned} &\text{maximize} && min(x) \\ &\text{subject to} && x \in \mathscr{F} \\ & && 0 \le x_i \le 1, \forall i \in N \end{aligned} \tag{5.2}$$

[3]See Appendix C to check how it is possible to refine the model.

but (5.2) coincides with:

$$\begin{aligned} &\text{maximize} && x \\ &\text{subject to} && x \in \mathscr{F} \\ &&& x_i = x_j, \forall i, j \in N \\ &&& 0 \leq x_i \leq 1, \forall i \in N \end{aligned} \tag{5.3}$$

In fact, the constraints of (5.3) imply that the optimal solution is the Pareto efficient solution that belongs to the line produced by the constraints $x_i = x_j$, $\forall i, j \in N$. This follows from the fact that all the other Pareto efficient solutions are such that the variable with the minimum value can be increased. The constraint $x_i = x_j$ implies that the satisfaction of each user is equal and this property characterizes, in the case of single resource allocations, the weighted proportional allocation when we choose the weights equal to the user demand. ■

Theorem 5.3.2 The MURANES framework with $w = (1, 0, \cdots, 0)$ and input $ds \cdot x$ coincides with the DRF allocation rule.

Proof. Similarly to the proof of Theorem 5.3.1, the considered optimization problem can be rewritten as:

$$\begin{aligned} &\text{maximize} && x \\ &\text{subject to} && x \in \mathscr{F} \\ &&& ds_i \cdot x_i = ds_j \cdot x_j, \forall i, j \in N \\ &&& 0 \leq x_i \leq 1, \forall i \in N \end{aligned} \tag{5.4}$$

that is exactly the DRF allocation rule described in Chapter 2. ■

Theorem 5.3.3 The MURANES framework with $w = (1, 0, \cdots, 0)$ and input ps generalizes to the multi-resource context the mood value.

Proof. Again, similarly to the proof of Theorem 5.3.1, the considered optimization problem can be rewritten as:

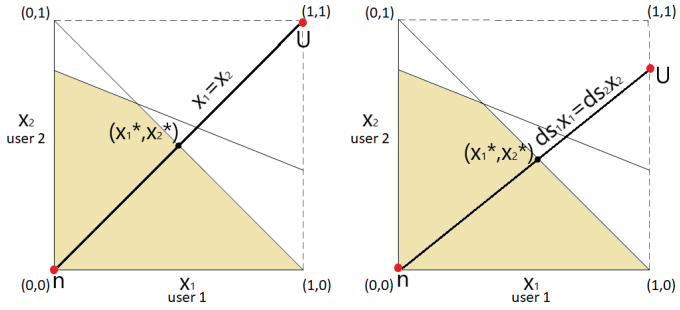
$$\begin{aligned} &\text{maximize} && ps \\ &\text{subject to} && x \in \mathscr{F} \\ &&& ps_i = ps_j, \forall i, j \in N \\ &&& 0 \leq x_i \leq 1, \forall i \in N \end{aligned} \tag{5.5}$$

So, the single resource allocation that equalizes the user satisfaction calculated using the *PS* is the mood value. ■

To sum up, the previous theorems show that MURANES allows to capture and generalize classical allocation rules. For the following, let us assign a name to the corresponding allocation rules obtained as a function of the OWA input:

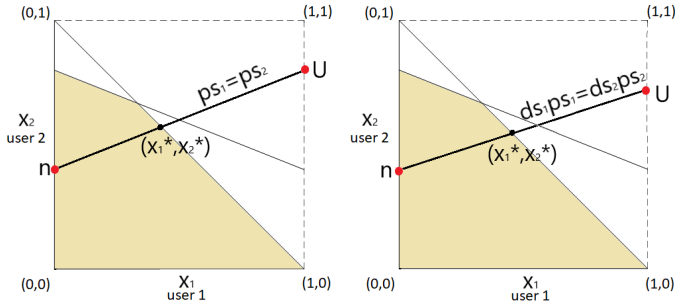
- generalized weighted proportional allocation (g-prop) when the input is x,
- generalized DRF allocation (g-drf) when the input is $ds \cdot x$,
- generalized mood value (g-mood) when the input is ps,
- moodified DRF (gm-drf) when the input is $ds \cdot ps$[4].

[4]The word 'moodified' comes from the fusion of the word 'mood' and 'modified', justified by the fact that the allocation considers the satisfaction rate typical of the mood value allocation but also the dominant share typical of the DRF allocation.



(a) Generalized proportional (g-prop)

(b) Generalized DRF



(c) Generalized mood value (g-mood)

(d) Moodified DRF (gm-drf)

Figure 5.5: Allocations with $w = (1,0,\cdots,0)$. (x_1^*, x_2^*) is the solution of the allocation problem. n is the nadir point, U is the utopia point.

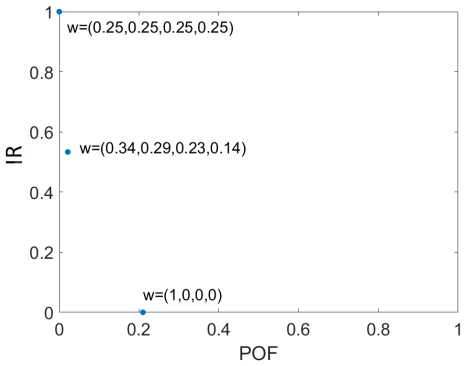
5.3.2 Game theoretic interpretation

Let us compare the four allocations rules defined in the previous section using the corresponding individual satisfaction vectors and the OWA weight vector $(1,0,\cdots,0)$. Fig. 5.5 shows on the tenants' satisfaction plane the region of the admissible solutions and the four allocation rules when we consider an allocation problem with two resources and two users.

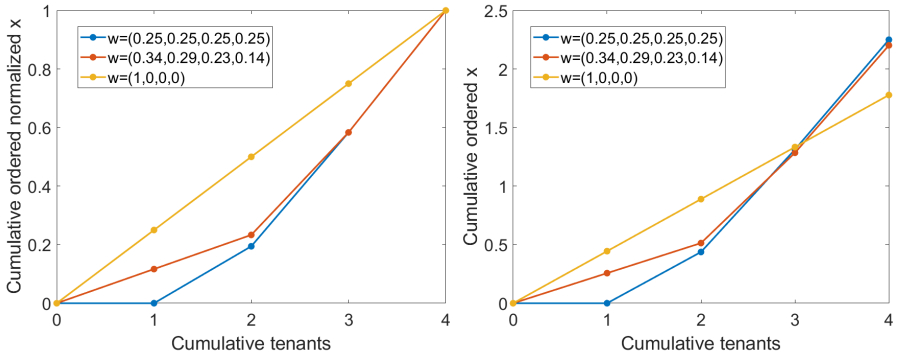
We can notice that the solution is the intersection between a line and the Pareto efficient frontier. The lines are:

- $x_1 = x_2$ for the g-prop allocation,
- $ds_1 x_1 = ds_2 x_2$ for the g-DRF allocation,
- $ps_1 = ps_2$ for the g-mood allocation,
- $ds_1 ps_1 = ds_2 ps_2$ for the gm-drf allocation,

These solutions can be interpreted as solutions of the bargaining game between two users. A bargaining game [35, 83] is a pair (C,n) where C is a bounded closed and convex set and n the utility when the two users are not able to reach an agreement. The egalitarian solution can be interpreted as the Kalai-Smorodinski (KS) solution [35] of the bargaining game, that is the solution on the Pareto frontier obtained joining the nadir and the utopia point. The nadir point n is $(0,0)$ for the first two allocation rules (Fig. 5.5a, 5.5b) while with respect to the two solutions obtained changing the satisfaction measure (Fig. 5.5c and 5.5d) the nadir point gives the minimal right for each user. Each component of the utopia (U) point is obtained maximizing the utility of each user. It follows that for the two cases considering the classical satisfaction the utopia point is respectively $U = (1,1)$, $U = (\frac{1}{ds_1}, \frac{1}{ds_1})$ while for the other two cases it it enough to calculate the maximal right of



(a) POF and IR indices



(b) Lorenz curve with normalized solution (c) Lorenz curve with non-normalized solution

Figure 5.6: Lorentz curves, POF and IR indices.

each user.

5.3.3 Egalitarian and utilitarian fairness trade-off

Let us elaborate on the potential of the unified framework. As anticipated in Section 5.2.1, the two extreme behaviors in terms of fairness are the egalitarian and the utilitarian ones. The utilitarian approach aims at maximizing the total utility of the members of a society without paying attention to social inequality; it is in fact also sometimes referred as *system efficiency* [84]. The egalitarian approach aims at maximizing the individual utility while promoting equitable distributions of utility; for this reason it is commonly used for fair optimization [84]. In most cases, the objective of reducing inequalities comes at a cost that can be measured by the *Price of Fairness (POF)* that is defined as follows.

Definition 5.3.1 The *Price of Fairness* (POF) is:

$$POF = \frac{f(x_f^*) - f(x_{min}^*)}{f(x_f^*)} \tag{5.6}$$

where $f(x) = \sum_{i=1}^{n} x_i$ is the utilitarian criterion, x_f^* is the solution obtained maximizing f and the x_{min}^* is the egalitarian optimum.

■ **Example 5.2** Let us consider a resource allocation problem with $D = \begin{bmatrix} 12 & 1 & 5 \\ 10 & 2 & 15 \\ 5 & 3 & 10 \\ 10 & 1 & 15 \end{bmatrix}$ and

$R = [20,4,20]$. In this case the utilitarian optimum is $x^*_f = (0.94,0,0.88,0.44)$ whereas the egalitarian optimum is $x^*_{min} = (0.44,0.44,0.44,0.44)$. Hence we obtain $POF = 0.21$. This value measures the normalized gap to optimal efficiency induced by the fairness requirement. ∎

In the above example the POF is moderate, which shows that perfect equity can be reached at reasonable cost regarding efficiency. This is not always the case and, in many situations, it can be interesting to determine solutions achieving a better compromise between pure utilitarianism and pure egalitarianism. This is precisely the interest of resorting to an OWA optimization that enables to generate various compromise solutions depending on the OWA weights. Let us come back to Example 5.2.

∎ **Example 5.3** A third solution of the problem presented in Example 5.2 obtained using OWA with the weighting vector $w = (0.34,0.29,0.23,0.14)$ is $x^*_w = (0.92,0.26,0.77,0,26)$. We can notice that:

- $x^*_{min} \geq x^*_w \geq x^*_f$,
- $\sum_{i=1}^4 x^*_{f_i} \geq \sum_{i=1}^4 x^*_{w_i} \geq \sum_{i=1}^4 x^*_{min_i}$,

∎

We give now a finer description of how inequalities and POF may vary when playing with OWA weights. For this purpose, we introduce the two following measures:

- $POF(x^*_w) = \frac{f(x^*_f)-f(x^*_w)}{f(x^*_f)}$ where $f(x) = \sum_{i=1}^n x_i$, x^*_f is the solution obtained in the utilitarian case and x^*_w is the solution maximizing an OWA with weight w.
- $IR(x^*_w) = 1 - \frac{OWA(x^*_w)}{(1/n)f(x^*_w)}$, where $f(x) = \sum_{i=1}^n x_i$ and x^*_w is the solution maximizing an OWA with weight w [85].

The first measure generalizes the one described in [84], that measures the loss of total utility faced by users in order to guarantee the fairness associated to weights vector w. The second index measures the inequality rate between the utility of the tenants. Both the indices have values in the closed interval $[0,1]$. So, according to the POF measure, the utilitarian solution gets value 0 and the price increases when we consider other solutions closer to the egalitarian one. Differently, the IR index has value 0 for the egalitarian solution and its value increases for the other solutions.

Due to the opposite behavior of the indices, a good trade-off between egalitarian and utilitarian criteria can be found in those solutions providing allocations vectors with indices POF and IR close to 0. Looking at the example depicted in Fig 5.6a we can see that the egalitarian solution has good properties in terms of equity but the POF has a value of around 0.2. If we are not willing to pay that price of fairness we can select the intermediate solution with a negligible price of fairness but we loose something in terms of fairness.

Another way to compare the various possible solutions is based on *Lorenz curves* [78]. A Lorenz curve is obtained plotting the cumulative x when we order the users from the less satisfied one to the most satisfied one. We plot in Fig. 5.6b and 5.6c the Lorenz curves for the resource allocation problem of Example 5.2 when we consider the normalized and non-normalized vector x, selecting three solutions of a resource allocation problem obtained using an egalitarian approach ($w = (1,0,0,0)$), an utilitarian approach ($w = (0.25,0.25,0.25,0.25)$)[5] and an intermediate one ($w = (0.34,0.29,0.23,0.14)$). In Fig. 5.6b the straight line represents the perfect equality in the distribution of the satisfaction between tenants and the most distant the curves are, the greater the inequality is.

[5]The weight $w = (0.25,0.25,0.25,0.25)$ is the weight $w = (1,1,1,1)$ normalized.

API Name	Memory (GB)	vCPUs	Gbps	Instance Type
m4.10xlarge	160.00	40.00	10.00	General purpose
m4.16xlarge	256.00	64.00	25.00	General purpose
c5.9xlarge	72.00	36.00	10.00	Compute optimized
c5.18xlarge	144.00	72.00	25.00	Compute optimized
c4.8xlarge	60.00	36.00	10.00	Compute optimized
r4.8xlarge	244.00	32.00	10.00	Memory optimized
r4.16xlarge	488.00	64.00	25.00	Memory optimized
x1.16xlarge	976.00	64.00	10.00	Memory optimized
x1.32xlarge	1952.00	128.00	25.00	Memory optimized
x1e.16xlarge	1952.00	64.00	10.00	Memory optimized
x1e.32xlarge	3904.00	128.00	25.00	Memory optimized
p3.8xlarge	244.00	32.00	10.00	Accelerated comput.
p3.16xlarge	488.00	64.00	25.00	Accelerated comput.
p2.8xlarge	488.00	32.00	10.00	Accelerated comput.
p2.16xlarge	732.00	64.00	25.00	Accelerated comput.
g3.8xlarge	244.00	32.00	10.00	Accelerated comput.
g3.16xlarge	488.00	64.00	25.00	Accelerated comput.
f1.16xlarge	976.00	64.00	25.00	Accelerated comput.
h1.8xlarge	128.00	32.00	10.00	Storage optimized
h1.16xlarge	256.00	64.00	25.00	Storage optimized
d2.8xlarge	244.00	36.00	10.00	Storage optimized
i3.8xlarge	244.00	32.00	10.00	Storage optimized
i3.16xlarge	488.00	64.00	25.00	Storage optimized

Table 5.2: Amazon EC2 instances

It is clear that the egalitarian solution, that aims to equalize the satisfaction of the users, provides a straight line, while the utilitarian solution provides a more unfair allocation. Contrarily, checking figure 5.6c we can notice that the sum of the users satisfaction are maximized with the utilitarian solution ($\sum_{i=1}^{4} x_i = 2.2500$) and it has the lower value for the egalitarian solution ($\sum_{i=1}^{4} x_i = 1.78$). Looking both the criteria (max-min and max-sum), the third considered solution shows an intermediate behavior representing the trade-off between utilitarian and egalitarian solutions.

Finally, it is worth to mention further properties that can be considered from a fairness point of view and can be used by the decision-maker to select the weights to use. For example one can be interested to (i) *strategy-proof* allocation where users should not be able to benefit by lying about their resource demands or to (i) *envy-freeness* allocation where a user should not prefer the allocation of another user. The DRF allocation, for example, satisfies these properties [41]. On the other side one can be interested into allocations equalizing the users satisfaction rate. In this case the DRF allocation is no more suitable and the g-prop and g-mood with weight $w = (1, 0, \cdots, 0)$ can be preferable.

Each allocation obtained with the MURANES framework, varying the weights vector, does not satisfy all the fairness properties we can consider at once. This gives more value to a general framework that can be better adapted to a specific context. The only properties satisfied by all MURANES allocation rules are the *Pareto efficiency* that state that it is not possible to increase the allocation of a user without decreasing the allocation of at least another user, and the Pigou-Dalton transfer already described in Section 5.2.1 [80].

5.4 Numerical evaluation

We test both the single-resource and the presented multi-resource allocation rules, in a realistic scenario. We simulate 100 resource allocation problems with 3 resources

(Memory, vCPU and link capacity) and 10 slices. We randomly generate the slice demands from the 23 templates described in Table 5.2, a subset of Amazon EC2 instances [11] we could extract (by simply copying the rows having a complete information about the three considered resources). In practice, in 5G slicing we can expect quite similar resource quantities and relations, with the link bit-rate at a lower scale as of preliminary specifications of some slices (e.g., the eMBB one) and related scenarios found in [75]. Different scales do not matter, the important aspect being the relation between resources.

In the first scenario we analyze, only 1 resource at time is congested. We randomly generate the amounts in this way:

- for the congested resource, the available amount has a value bigger than the minimum demand and lower than the sum of the demands;
- for the non-congested resource, it is between the sum of the demands and two times the sum of the demands.

In the second scenario, all the resources are congested but not always at the same level of congestion. The ratio of available resource ρ considered is the fraction of the global demand (sum of all demands) that can be allocated; e.g., if the level is 0.9, 90% of the sum of the demands is satisfied, thus we are in a low congestion situation. In the simulations, we consider the following four cases of congestion level combinations:

- 0.1, 0.1, 0.1: 3 resources have the same high congestion;
- 0.9, 0.9, 0.9: 3 resources have the same low congestion;
- 0.1, 0.9, 0.9: 1 resource has high and 2 have low congestion;
- 0.1, 0.5, 0.9: the 1st resource has a high congestion, the 2nd one a medium level and the 3rd one a low level.

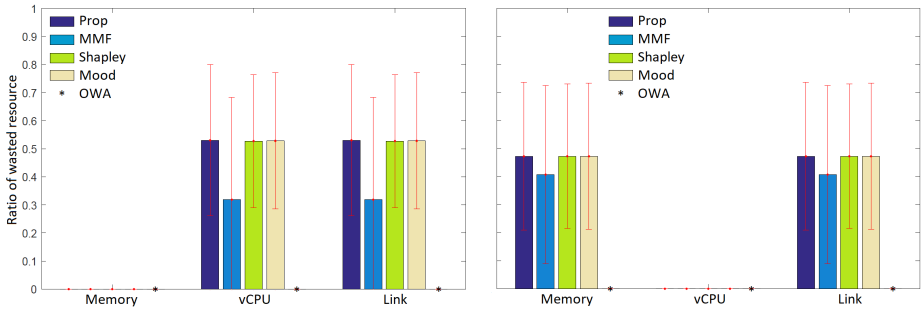
The first two cases show a homogeneous congestion distribution, while the latter two have a heterogeneous distribution that likely corresponds to a more realistic setting.

We test the presented single-resource allocations (weighted proportional with $p_i = d_i$, MMF, Shapley value, Mood value)[6], and the proposed MURANES rules with OWA weights $w = (1, 0, \cdots, 0)$ because we are interested in evaluating the performance of the already known solution (i.e. DRF) compared to the new proposed one, that generalizes single resource allocation (g-prop, g-mood), or that are not known (gm-drf).

5.4.1 Results in terms of wasted and idle resource

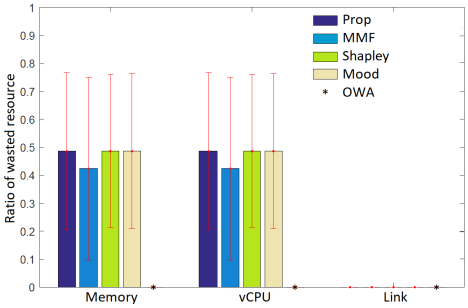
Fig. 5.7 shows the average ratio of wasted resource in the case in which only one resource is congested. Fig. 5.8 shows the same, but when all the resources are congested. We can notice, as we already discussed, that single-resource allocations produce resource wasting, i.e., even if a resource is allocated, it may not be fully needed due to the assumed relation between resources. For single-resource allocations, the trend in terms of wasted resource depends on the congestion level: if the resource is congested it is fully allocated, and consequently the wasted resource is zero; in case of equal congestion level between the resources (Fig. 5.8a, 5.8c), there is a similar ratio of wasted resource between the three resources; in the case in which the level of congestion is heterogeneous, the ratio of waste resource is zero for the most congested resource, and it increases decreasing the congestion level. Multi-resource rules, respecting inter-resource dependency, do not produce wasted resource, in each congestion level configuration. This means that there are no resources allocated and unused by the users because multi-resource rules allocate for each user the same percentage of demand for each resource.

[6] except the Nucleolus, whose computation has a high time complexity



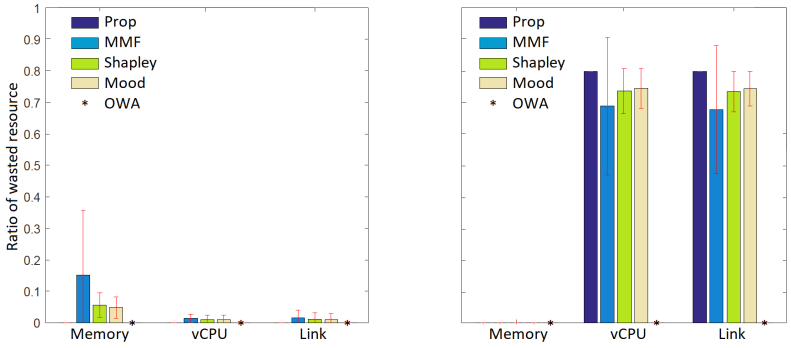
(a) Memory congested

(b) vCPU congested



(c) Link congested

Figure 5.7: Wasted resource ratios (1 congested resource). Multi-resource rules are referred as 'OWA'.



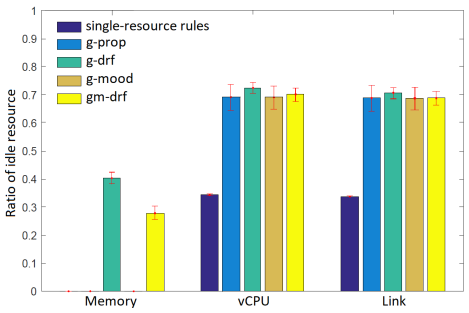
(a) Congestion level (0.1, 0.1, 0.1)

(b) Congestion level (0.1, 0.9, 0.9)

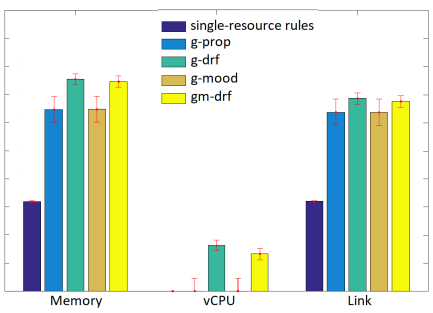
(c) Congestion level (0.9, 0.9, 0.9)

(d) Congestion level (0.1, 0.5, 0.9)

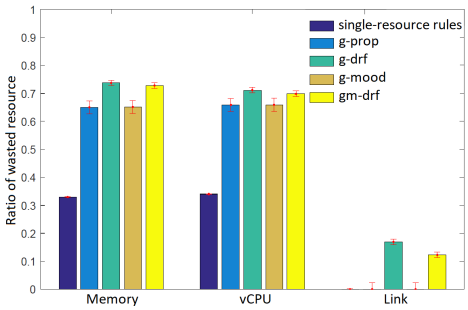
Figure 5.8: Wasted resource ratios (3 congested resources). Congestion level: (Memory, vCPU, Link).



(a) Memory congested

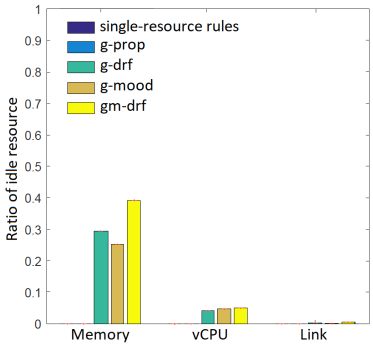


(b) vCPU congested

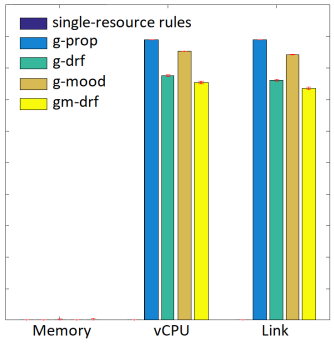


(c) Link congested

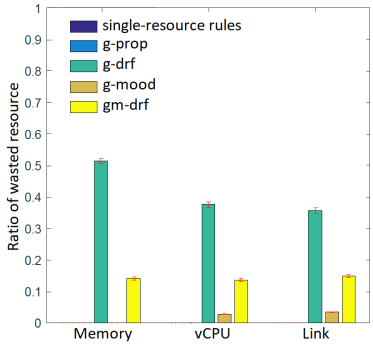
Figure 5.9: Idle resource ratios (1 congested resource).



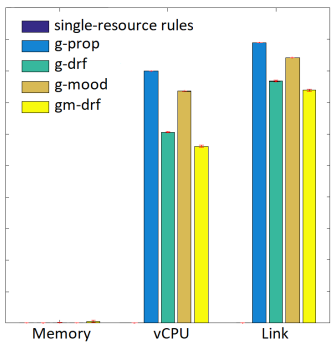
(a) Congestion level (0.1, 0.1, 0.1)



(b) Congestion level (0.1, 0.9, 0.9)

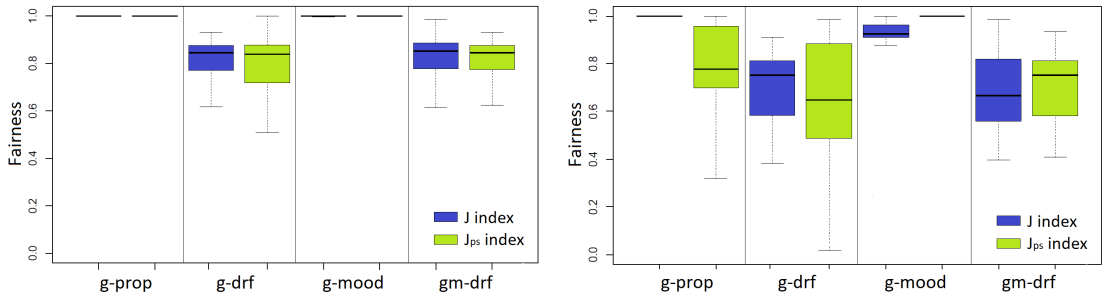


(c) Congestion level (0.9, 0.9, 0.9)



(d) Congestion level (0.1, 0.5, 0.9)

Figure 5.10: Idle resource ratios (3 congested resources). Congestion level: (Memory, vCPU, Link)



(a) 1 resource congested (b) 3 congested resources

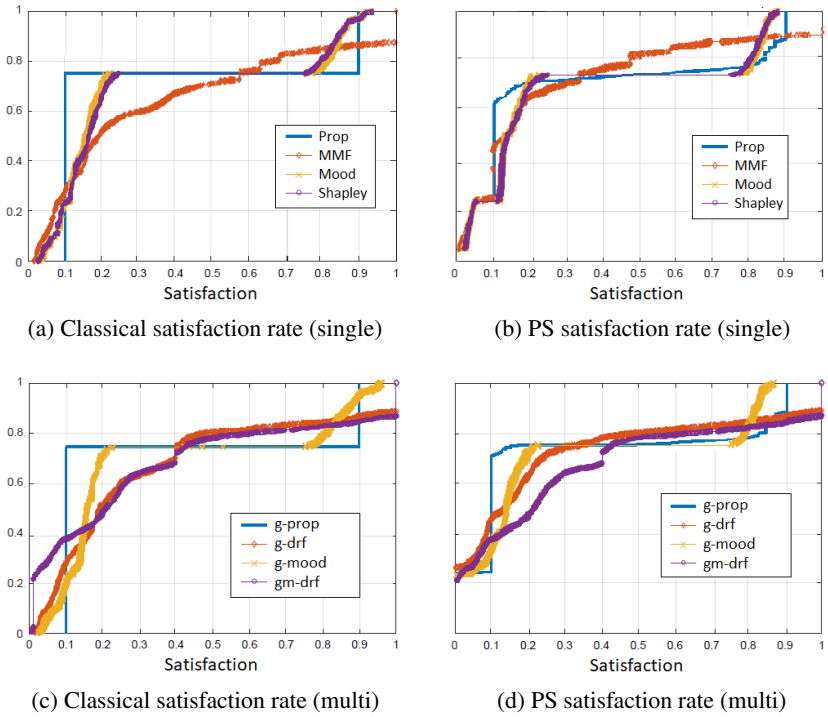
Figure 5.11: Fairness index with different allocation rules.

In a dual way, Fig. 5.9 and Fig. 5.10 show respectively the average ratio of idle resource in the cases in which only one resource is congested, and when all the resources are congested. We can notice that single resource allocation rules produce idle resource only if the resource is non-congested; for this resource, tenants receive exactly what they ask and consequently, being the resource non-congested, an idle resource is produced. For multi-resource allocations, there is a similarity between the two allocations that consider the satisfaction rate (g-prop and g-mood), and between the two allocations that weight the satisfaction rate with the dominant share (g-drf, gm-drf). The first couple of allocation rules produce less idle resource when (i) only one resource is congested or (ii) the congestion level is homogeneous. The second one, adapting the satisfaction to the the resources available in the network in which the slice is situated, produces less idle resource when the congestion level is heterogeneous.

5.4.2 Results in terms of fairness

In order to analyze the fairness of the allocation rules, we analyze the Jain's index of fairness [38] and its modification considering the PS rate instead of the classical Demand Fraction Satisfaction (DFS) rate (see Chapter 3). Fig. 5.11 shows the boxplot results of the fairness index for the dominant resource, and for the two congestion cases. We can notice that the two solutions with better performances in terms of fairness are g-prop and g-mood, i.e., the ones considering as OWA input the DFS and PS rates. This follows from the fact that the two allocations equalize the tenant satisfaction and consequently maximize the respective index of fairness. Considering the dominant resource for each tenant, the satisfaction is no more the same for each tenant thus the fairness decreases, but on average not excessively.

Fig. 5.12 and 5.13 show, for the two satisfaction rate definitions (classical and PS) and for both single- and multi-resource allocation rules, the cumulative distribution function (CDF) of the minimum satisfaction rate, i.e., among the three resource-specific satisfaction rates, the least one. In this way we can focus on the minimum satisfaction rate as a desirable fitness metric to increase. Fig. 5.12 refers to the 3-congested resources case, while Fig. 5.13 to only the heterogeneous cases, i.e., $(0.1, 0.9, 0.9)$ and $(0.1, 0.5, 0.9)$. Again we can notice a similarity between g-prop and g-mood (with OWA input equal to x and ps) from the one hand, and g-drf and gm-drf (with $ds \cdot x$ and $ds \cdot ps$) from the other hand. We see that the minimum satisfaction is clearly linked to the congestion level. In Fig. 5.12 we have 3 cases over 4 with $\rho = 0.1$ for at least one resource; it follows that the least satisfaction is the one related to the most congested resource, getting a value (with the classical DFS rate) exactly



(a) Classical satisfaction rate (single)

(b) PS satisfaction rate (single)

(c) Classical satisfaction rate (multi)

(d) PS satisfaction rate (multi)

Figure 5.12: Minimum satisfaction rates CDF (3 congested resources).

equal to 0.1 for the proportional and the generalized weighted proportional rules. About 50% of the tenants suffer from a very low satisfaction (between 0 and 0.15).

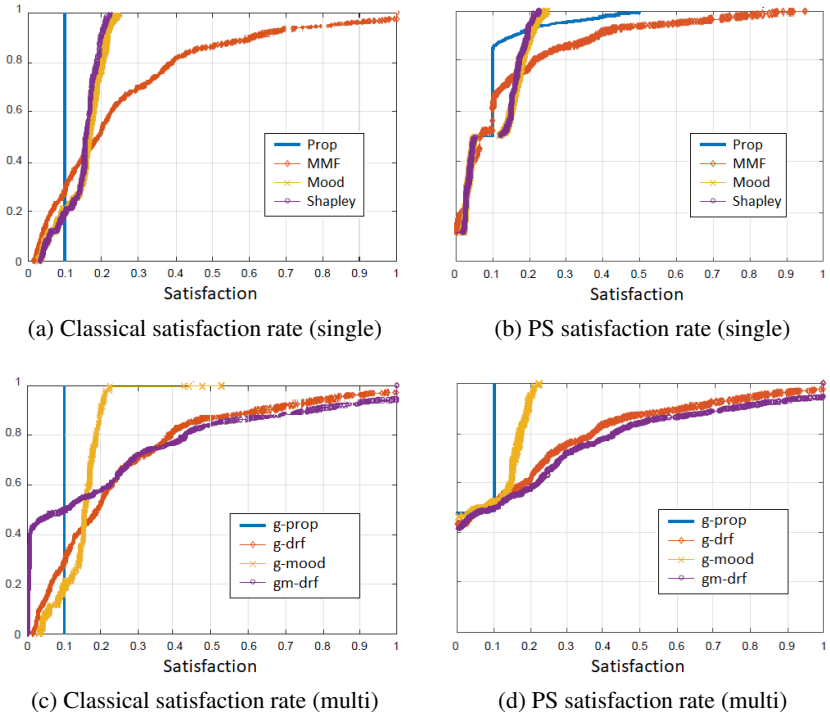
Therefore, we compare the global (i.e., with both heterogeneous and homogeneous congestion cases – Fig. 5.12) results to the one with only heterogenous congestion cases (Fig.5.13). In the former the satisfaction rate CDFs for single and multi-resource allocations are similar: MMF, gm-drf and g-drf assign the highest satisfaction rate to about 10% of tenants, and are hence preferable. This follows from the the fact that g-drf and gm-drf can be considered as generalizations of the MMF allocation. With the heterogeneous cases apart (Fig.5.13), instead, gm-drf is superior to all the other allocation rules (single- and multi-resource ones), except for MMF with the classical satisfaction rate (Fig.5.13a) which however is known to offer low fairness.

These results show that in realistic settings with heterogeneous resource congestion, the MURANES rules we propose, and in particular the m-drf, g-prop and g-mood rules, clearly outperform the application of single-resource allocation rules.

5.5 Resource allocation under Service Level Agreements

We conclude the analysis of the multi-resource allocation for network slicing providing two multi-resource scheduling algorithms, able to slice the resources between tenants and to fulfill their Service Level Agreements (SLAs).

A 5G network needs to fulfill SLAs that are contracts between the providers and the customers that specify the technical conditions of a service provisioning, i.e, connection performance, availability, liability etc., and the price of the services [86], by means of measurable parameters or metrics [87]. In 5G case, the contract between the slice provider and the tenant can specify (i) the minimum guaranteed and nominal capacities for each



(a) Classical satisfaction rate (single)

(b) PS satisfaction rate (single)

(c) Classical satisfaction rate (multi)

(d) PS satisfaction rate (multi)

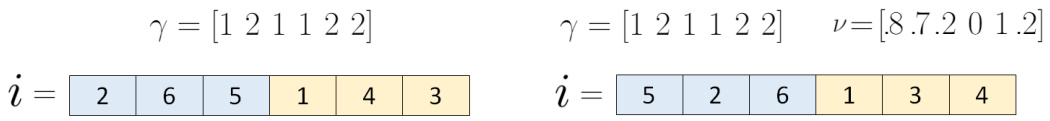
Figure 5.13: Minimum satisfaction rates CDF (3 resources congested - heterogeneous congestion levels)

given resource, (ii) the amount of time the service is guaranteed, (iii) penalties in case the service requirements are not met, (iv) latency or jitter, (v) the service assistance, etc. Between this specification the one strictly linked to the resource allocation is (i) while the ones related to a scheduling process are (ii)-(iv). In particular, in this section we consider three metrics: the first one is the guarantee of the *minimum service*, i.e., the minimum amount of resource that has to be guaranteed to the tenants; the second one is the *nominal capacity*, i.e., the amount of resource required in normal conditions, while the third one is the *service availability*, i.e., the measure, in percentage or units of time, of the successful service access to the tenant. We firstly propose an algorithm considering only the minimum service and the nominal capacity requirements, which we then refine to consider also service availability and to provide an allocation fairly distributed on time.

In the following, we (a) model the problem (Section 5.5.1), (b) establish a users delaying policy (Section 5.5.2), (c) define how to allocate the resources, under the constraint of guaranteeing a minimum share of resource (Section 5.5.3), (d) propose two scheduling algorithms (Sections 5.5.4, 5.5.5) and (e) we test the proposed algorithm (Section 5.5.6).

5.5.1 Problem statement

Given a time frame t, the resource allocation problem is a tuple $(D_t, D_t^m, \gamma_t, \nu_t, R_t)$ where D_t is the demand matrix, D_t^m is the matrix containing the minimum amounts of resource to allocate to each tenant, R_t is the available resource, γ_t is a vector containing the priority index of each tenant, ν_t is a vector containing the availability rate of each tenant, i.e., it contains the percentage of time the tenant was served, with at least the minimum resource. The priority index γ_t is linked to the latency of the service: if the service requires a low latency, its priority is high and the value of γ_t is low; if not, the priority is lower and the



(a) Considering the priority index

(b) Considering the priority index and the availability rate

Figure 5.14: Order of users delaying

value of γ_t is higher. For instance URLLC services are characterized by higher priority indexes compared to eMBB and mMTC services.

Key assumptions are as follows: (i) the demand processing time is discrete, i.e., the matrices D_t and D_t^m collect the information about the users demand in the time frame $t-1$ and they are treated in the same moment when the time frame t starts; (ii) the priority index γ_t depends only on the service latency, while in general it can be linked also to the tenant importance, measurable by how much a tenant is willing to pay for a service. Note that in the following when we avoid the subscript t in the notation we are considering a generic instant of time t.

5.5.2 User delaying policy

When the slice provider is not able to satisfy the minimum allocation of each tenant ($\sum_{i=1}^{n} d_{ij}^m > r_j$, for at least one resource j), it is necessary to introduce a process to eliminate tenants and put them in hold for the next time slot. The order on which tenants have to be held needs to take into account the tenant priority index. Different user delaying policies are possible. Here we propose one that takes into account only the priority index, and one that considers both the user priority index and the current availability rate of each tenant.

The two policies are depicted in Figure 5.14. In the first one only the priority index is used as decision variable. The order of users to remove is established ordering the users from the lower to the higher priority ones, and if tenants belong to the same class of service (i.e. they have the same index) the choice is done randomly. In Fig. 5.14a the vector is giving the position index of the users and it is clear that firstly tenants with priority 2 are eliminated, and then are the ones with priority 1. The second policy is based on the idea that we should firstly consider the priority index and then look at the value of ν. Higher values of ν correspond to higher percentages of time in which tenants are served in the past. It follows that, inside the same class of service we should eliminate users from the one with highest value of availability rate to the one with a lower value in order to enforce fairness within the same class of service. In Fig. 5.14b, for example, the first tenant of the list is the 5th one because it has priority 2 and it was served 100% of the times, while the last is the 4th that has high priority and it was never served until the considered instant of time.

5.5.3 Multi-resource allocation with minimum demand

To provide an allocation that guarantees a quantity of resource not inferior to the minimum demand we need to modify the capacity constraint of the optimization problem 5.1. In this case we can impose the condition that the percentage of resource to allocate to each tenant is bigger than the minimal ones calculated as $x_i^m = max_j(x_{ij}^m) = max_j\left(\frac{d_{ij}^m}{d_{ij}}\right)$.

Algorithm 2 Allocation considering minimum capacity requirements (MIN-CAP)

Input: R, D, D^m, N, M, γ
Output: A

$o \leftarrow$ ordered vector of users using γ
$N^* \leftarrow N$
$P \leftarrow \emptyset$
$count \leftarrow 1$
while it $\exists$ at least one $j \in M$ s.t. $\sum_{i \in N^*} d_{ij}^m > r_j$ **do**
  $i^* \leftarrow o(count)$
  $N^* \leftarrow N^*_{-i^*}$
  $P \leftarrow P_{+i^*}$
  for k=card(P):1 **do**
    if $\sum_{i \in N^*} d_{ij} + d_{kj} \leq r_j, \forall j \in M$ and $k \in P$ **then**
      $N^* \leftarrow N^*_{+k}$
      $P \leftarrow P_{-k}$
    end if
  end for
  $count \leftarrow count + 1$
end while
if it $\exists$ at least one j s.t. $\sum_{i \in N^*} d_{ij} > r_j$ **then**
  $a_i \leftarrow$ solution of (5.7) $\forall i \in N^*$
else
  $a_i \leftarrow d_i \; \forall i \in N^*$
end if
$a_i \leftarrow zeros(m) \; \forall i \notin N^*$

It follows that the problem to solve is:

$$\begin{aligned} &\text{maximize} && OWA(v) \\ &\text{subject to} && x \in \mathscr{F}, \\ & && x_i^m \leq x_i \leq 1, \forall i \in N \end{aligned} \tag{5.7}$$

where v is one of the OWA input described before (i.e., x, $ds \cdot x$, ps, $ds \cdot ps$).

It is possible that the optimization problem has no solution when there are no enough resources to satisfy the minimal demands of the tenants. For this reason we introduced the user delaying policy. In next sections we combine the proposed resource allocation and the delaying policy to get the two scheduling algorithms able to satisfy SLA constraints.

5.5.4 Baseline algorithm: minimum capacity (MIN-CAP)

We propose a baseline algorithm called 'MIN-CAP', that uses the first re-order of the users, i.e., the one considering only the priority index.

The allocation resulting from (5.7) is calculated after having checked that the minimum demands for each tenant can be satisfied. In the case this is not possible, the tenants are, one at time, delayed using the proposed order. Each time a user is delayed the algorithm checks if there is one or more than one user already delayed that can be re-introduced because its own minimal demand can be satisfied. Obviously the order used for the re-introduction check follows the reverse order of the tenants delaying.

Algorithm 3 Refined algorithm (REF-MIN-CAP)

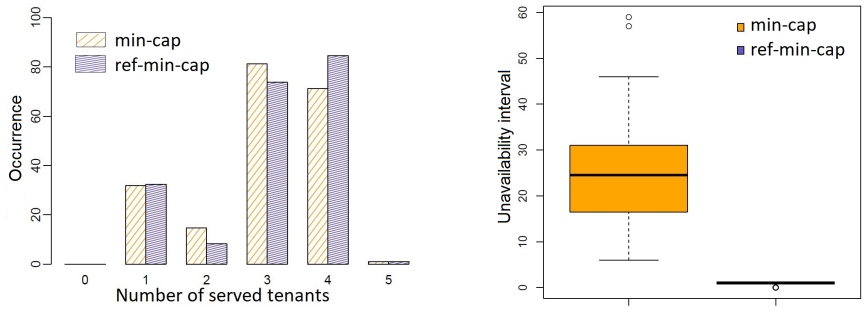
for t = 0:T **do**
  Input: $R_t, D_t, D_t^m, N, M, \gamma_t, \nu_t$
  Output: A_t
  We avoid from here the subscript t
  $o \leftarrow$ ordered vector of users using γ and ν
  $N^* \leftarrow N$
  $P \leftarrow \emptyset$
  $count \leftarrow 1$
  while it $\exists$ at least one $j \in M$ s.t. $\sum_{i \in N^*} d_{ij}^m > r_j$ **do**
    $i^* \leftarrow o(count)$
    $N^* \leftarrow N^*_{-i^*}$
    $P \leftarrow P_{+i^*}$
    for k=card(P):1 **do**
      if $\sum_{i \in N^*} d_{ij} + d_{kj} \leq r_j, \forall j \in M$ and $k \in P$ **then**
        $N^* \leftarrow N^*_{+k}$
        $P \leftarrow P_{-k}$
      end if
    end for
    $count \leftarrow count + 1$
  end while
  update of ν
  if it $\exists$ at least one j s.t. $\sum_{i \in N^*} d_{ij} > r_j$ **then**
    $a_i \leftarrow$ solution of (5.7) $\forall i \in N^*$
  else
    $a_i \leftarrow d_i \ \forall i \in N^*$
  end if
  $a_i \leftarrow zeros(m) \ \forall i \notin N^*$
  $t = t + 1$
end for

The pseudo-code shows the algorithm used at time slot t. The notation is lightened avoiding the subscript t.

5.5.5 Refined algorithm: considering service availability guarantees (REF-MIN-CAP)

The MIN-CAP algorithm does not take into account SLA requirements on the service availability. For example, if a tenant is left in a standby state at scheduling time slot t in order to guarantee the minimum level of service to the other tenants, it shall likely be served in the time slot $t+1$, or not too late. Thus, we want an algorithm that is time-fair, i.e., when the number of time slots T is big enough, the waiting time for each tenant is similar and kept small.

With the refined REF-MIN-CAP algorithm, in order to provide time-fair allocations we take into account the availability rate ν and we use the same algorithm, changing only the user delaying policy. In particular we use the second policy considering both ν and γ.



(a) Average number of served clients

(b) unavailability gap (best served - worst served)

Figure 5.15: Number of served client and unavailability gap

5.5.6 Numerical evaluation

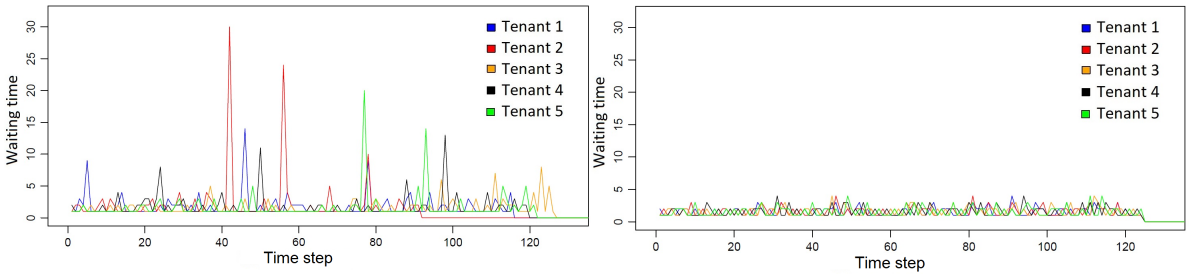
We provide two cases for the numerical analysis. In the first one we compare the two algorithms in the case in which the priority index of each user is the same. This means that for the first algorithm the users delaying policy is random, while for the second one it depends only on the availability rate. We consider 200 time slots and a slicing problem with 3 resources meant to represent live memory, vCPU, link capacity, and 5 slices in the scheduling queue; the resource amounts are respectively fixed to 2000 GB, 150 vCPU and 50 Gbps.

We randomly generate the slice demands using a subset of Amazon EC2 instances [11] (Table 5.2) so that the congestion levels (fraction of the global demand not allocated due to resource scarcity) is heterogeneous. The minimum demand associated to each tenant is the minimum template available for each 'Instance Type': for example if the tenant demand instance type is 'compute optimized', then its minimum demand is 72 GB, 36 vCPUs, 10 Gbps (c5.9xlarge). We repeat the simulation 100 times.

We are interested in evaluating the performance of the two proposed algorithms. Figure 5.15 shows the average number of clients that are served at each time slot and the boxplots of the gap between the number of times the best served and the worst served tenant are served. From Figure 5.15a we can notice that there are no big differences in the number of served client between the two algorithms. The REF-MIN-CAP one is slightly better because it increases the number of times 4 tenants are served. The major differences between the two algorithms are shown in Figure 5.15b. It is clear that for REF-MIN-CAP after 200 time slots the number of time tenants are not served is the same (the gap is between 0 and 2 time slots) while with MIN-CAP the best served client is served a higher number of times, with a median value around 25.

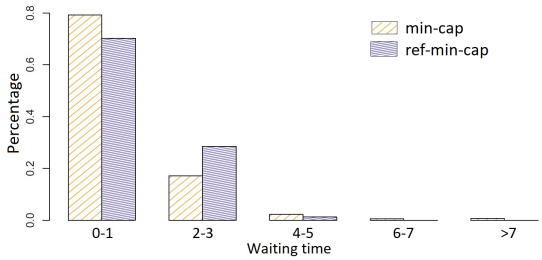
Figure 5.16 shows the results of the waiting time analysis. Figure 5.16a and 5.16b show the waiting time of one simulation repetition on the 200 time slots. We can notice that MIN-CAP does not prevent the waiting time from growing excessively (in our case it can reach 30 time slots). This is due to the absence of the availability index that considers past service times. This index, that is present in REF-MIN-CAP, avoids an excessive growth of the waiting time and in particular, in our case, the waiting time does not exceed 5 time slots (Figure 5.16b). Figure 5.16c confirms that the waiting time of REF-MIN-CAP is bounded at 5, while for MIN-CAP, even if with a low probability, it can take higher values.

In the second numerical case we want to compare the two algorithms when users have different priorities linked to the latency required by the service. Following what is



(a) Waiting time of one repetition using MIN-CAP

(b) Waiting time of one repetition using REF-MIN-CAP



(c) Waiting time histogram

Figure 5.16: Waiting time analysis.

Service type	Instance type	γ
URLLC	Accelerated computing	1
eMBB	Compute optimized Memory optimized	2
mMTC	Storage optimized	3
Best effort	General purpose	4

Table 5.3: Adopted mapping of Amazon templates to 5G slices.

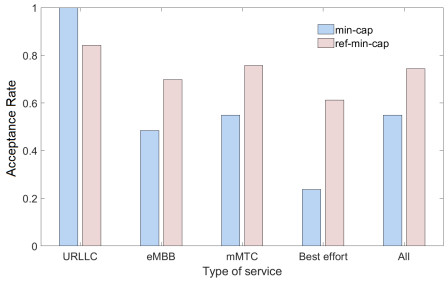


Figure 5.17: Service availability for different slices.

recommended in [88], the importance of latency requirement is high for URLLC services (implying not only very low propagation delay but also very low coding and processing time), medium for eMBB services and low for mMTC services. Moreover, mMTC service are expected to call for in-network storage and reformatting of exchanged IoT or machine generated data. Finally, eMBB services are expected to call for an amount of computing resources proportional to the bit-rate, which is meant to be an important one, in the order of the Gbps. Given this qualitative requirements, at first instance, we consider four levels of priority: three characterizing the three class of services proposed in the 5G and one characterizing the best effort class. Given the lack of slice templates in current 5G specifications, we propose to derive them and differentiate them using Amazon template instance type in Table 5.2. Table 5.3 shows, according to the service requirement assumptions above, we associate the accelerated computing template to URLLC, the storage optimized one to mMTC, the compute and memory optimized one to eMBB and the general purpose one to the best effort class; the value of γ is an arbitrary one, it just indicates the priority order.

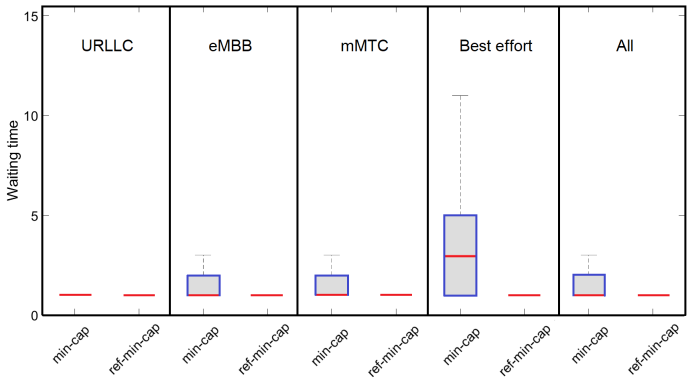


Figure 5.18: Boxplot of the waiting time slots.

Hence we randomly generate the slice demands using the differentiated subset of Amazon EC2 instances [11] (Table 5.2), including the instance type for class differentiation as per Table 5.3, with an heterogeneous level of congestion, and the minimum demand considered for each tenant set as the minimum template available for each 'Instance Type'. We repeat the simulation 100 times.

We plot in Fig. 5.17 the availability performance, i.e., the percentage of time a tenant is served when it submits a request. We do not differentiate among the case where the demand is a new demand, and the one where a demand comes from a tenant that is waiting to be served for already some time-slots. We can clearly see that the best-served tenants are the ones requiring an URLLC service; however, the REF-MIN-CAP algorithm shows more balance in the availability performance also for low priority classes. Non differentiating the service types (columns All), the REF-MIN-CAP brings a better global availability with respect to MIN-CAP.

We then consider in Fig. 5.18 the waiting time of the tenants, i.e., the time passing from the submission of the demand and the time in which the service is provided. We do not plot the outlier values of the boxplot, but we summarize the information about them in Table 5.4. One can clearly observe that the second algorithm has better performance because, with a probability of at least 75%, the tenants are served when they submit the demand, independently of the type of service they require. On the other hand, the first algorithm differentiates the tenants, serving with a probability of at least 75% the URLLC tenants in 1 time-slot, and the eMBB and the mMTC tenants in 1 or 2 time-slot and the best effort in maximum 5 time-slots. Nonetheless, we need to consider that there is a not negligible probability that the service time gets high. Analyzing Table 5.4 we notice that for URLLC services, with the REF-MIN-CAP, it is possible for tenants to wait 2 time slots before being served, while with the first algorithm the probability that tenants are delayed is negligible. For all the other types of service, tenants can wait a long time before being served, but introducing the considerations about the availability rate in the delay policy, we can reduce the waiting time. In particular both the lower bound and the upper bound of the time-slot range decrease using REF-MIN-CAP. This shows how the second algorithm tries to enhance the global availability for the tenants, while providing a scheduling algorithm that is "time-fair".

We can conclude that REF-MIN-CAP differs from MIN-CAP, as it takes into account the history of service of the tenants, by

- avoiding an excessive increase of the waiting time;

Service Type	MIN-CAP		REF-MIN-CAP	
	Probability	Range	Probability	Range
URLLC	10^-4	2	0.18	2
eMBB	0.1	$[4,56]$	0.2	$[2,45]$
mMTC	0.09	$[4,15]$	0.19	$[2,9]$
Best effort	0.06	$[12,46]$	0.2	$[2,16]$

Table 5.4: Boxplots outliers

- improving the overall availability of the system.

When taking into account services belonging to different classes, i.e., with different priorities, to consider the availability rate slightly penalizes tenants with high priority, while it can improve the satisfaction of the other tenants, decreasing the waiting time. This behavior can certainly be marginally modified toward more specific requirements adequately tuning algorithm parameters.

5.6 Summary

In this chapter we explored in depth the problem of resource allocation in network slicing where multiple resources have to be allocated to verticals and shared concurrently. The main contribution is the formalization of the problem, under the important assumptions that not the entire amount of requested resources can be assigned to tenants, and that guaranteeing a relationship between allocated slice resources is important for an efficient operation of related services.

We propose a multi-resource allocation framework, called MURANES, based on the Ordered Weighted Average (OWA) operator to generalize the most known single-resource and multi-resource allocation rules and to define new ones. It is worth to notice that the approach provides meaningful solutions also in the case in which the users demand for a subset of resources is zero. In fact, being the solution the portion of demand to allocate, if the demand is zero for one resource, the resource is not allocated by the provider.

We provide a complete analysis of the proposed framework and we show how it lets to the decision-making the freedom to select the most appropriate allocation, based on the fairness goal it is meant to follow. Through extensive simulations we characterize the behavior of the allocation rules in terms of fairness and in terms of wasted resource. As opposed to single-resource allocation rules, multi-resource allocation rules (i) have the key advantage of not allocating unneeded surplus of resources, (ii) can allow for idle capacity to support traffic peaks, and (iii) are superior in terms of satisfaction rate in case of heterogeneous congestion (i.e., not all resources are equally congested) – which happens for the generalized DRF and moodified DRF. Among multi-resource allocation rules, we could highlight that the fairest ones are the proposed OWA generalization of the weighted proportional allocation and of the mood value.

Extensions of the MURANES framework are possible to deal with the case in which the relationship between resources is not linear and considering Service Level Agreement (SLA) constraints. In particular, we present two scheduling algorithms fulfilling SLA requirements: a baseline algorithm considering the capacity constraints and the users priority and a second one aiming to provide time-fair allocations. The proposed algorithms, and in particular the second one, represent starting points to customize network slicing allocation performance toward more specific SLA requirements. For example, it may be interesting to introduce the notion of demand expiration time, i.e., a deadline within which

the service must be provided.

Up to this chapter we analyzed centralized approaches to allocate resources, i.e., when there is one resource provider providing all the considered resources and taking the decision about the resources re-partition. In the following chapter we deal with decentralized approaches to allocate resources, applied in the specific case of 5G slice resource allocation, but that work, in general, for multi-resource allocation problems where resources do not belong to only one provider.

6. Decentralization of 5G slice orchestration

In the previous chapter we propose a centralized approach to allocate resources in network slices. A centralized approach implies the presence of a multi-domain orchestrator able to manage heterogeneous resources. Nowadays each part of the network is managed separately (Fig. 6.1) and the presence of this kind of orchestrator may be not viable in practice because resource can belong to different resource providers, e.g., the radio resource is managed by radio operator and the cloud resource is managed by a cloud service provider.

In this chapter, we are interested in investigating distributed algorithms able to allocate slices. In particular, we propose three algorithms: two use a cascading approach and one a parallel approach. Our reference scenario is an end-to-end path where the resources to allocate are of three types: radio, link and cloud while being applicable to an arbitrary number of distributed resources. We compare the approaches quantitatively (time complexity, message overhead, latency budget) and qualitatively (advantages, disadvantages).

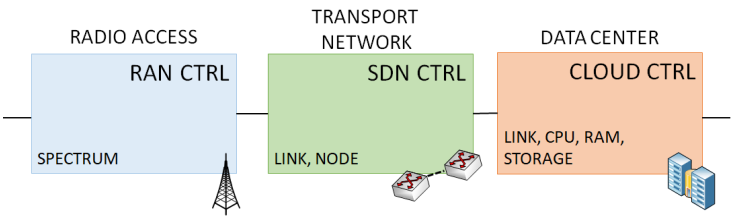


Figure 6.1: End to-end network and orchestrators

The work presented in this chapter is partially presented in [89]

Algorithm 4 Priority-aware allocation rules logic

Input: R, D, N, M, γ
Output: x

for pr = 1:s **do**
  S=set of user with $\gamma = pr$
  Q=set of user with $\gamma > pr$
  if $\sum_{i \in S} d_{ij} \leq r_j, \forall j \in M$ **then**
    x=ones($|S|$)
  else
    x_S= solution of the selected allocation rule
    x_Q=zeros($|Q|$)
    exit for loop
  end if
end for

6.1 Distributing the slice resource allocation

6.1.1 Problem modelling

Let $N = \{1, ..., n\}$ be the set of tenants, $M = \{1, ..., m\}$ be the set of available resources and $P = \{1, ..., p\}$, with $p \leq m$ be the set of resource providers. The allocation problem is represented as a triplet (D, R, γ), where D is a $n \times m$ matrix with d_{ij} equal to the quantity of resource $j \in M$ demanded by tenant $i \in N$, $R = (r_1, ..., r_m)$ is a vector of positive numbers r_j equal to the amount of each available resource $j \in M$, and γ is a n-dimensional vector containing the priority index of the service required by tenants.

In this chapter we consider the priority index γ linked to the latency of the service, as in the previous chapter. Services requiring low latency have high priority and a low value of γ, those tolerant to higher latency have lower priority and the correspondent value of γ is high. E.g., considering the three classes of service formalized for the 5G, following what is recommended in [88], the importance of latency requirement is high for URLLC services, which refers to wireless connection with low latency, medium for eMBB services, which needs high data bandwidth and moderate latency, and low for mMTC services because they focus on massive objects connectivity, with no strict latency requirements [90]. For this reason, at first instance, we consider 3 priority levels characterizing the 3 5G classes of services.

Another important aspect to model in network slice resource allocation is the relation between allocated resources. As already assumed in previous chapters and in some works [65, 72], we model a linear relationship; this means that if a user asks for 10 Gbps, 40 CPU and 160 GB and it receives only 5 Gbps then the cloud resource provider has to allocate 20 CPU and 80 GB because if the allocation is superior, the cloud resource is wasted, while if inferior, the link resource is wasted.

Let the allocation outcome be represented by a matrix A with components $a_{ij} = d_{ij} \cdot x_i$ where $x = (x_1, ..., x_n)$, $0 \leq x_i \leq 1 \ \forall i \in N$, is the vector of the percentage of demand allocated to each tenant. The allocation is not trivial if it exists a resource $j \in M$ such that $\sum_{i=1}^{n} d_{ij} > r_j$ because the resource is not sufficient to fully allocate the demands of the users (i.e. the resource is congested - in the trivial case the resource provider can allocate the demand and $x = (1, \cdots, 1)$). The three algorithms proposed in next subsections take into account that resources can be congested.

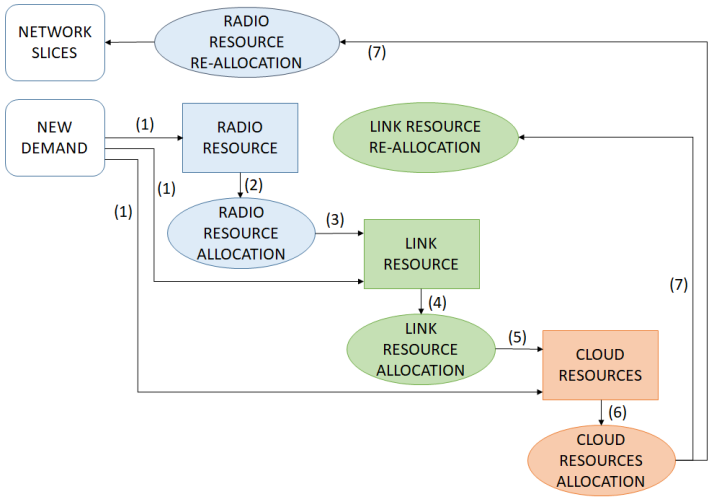


Figure 6.2: CRA algorithm

We define the congestion level (μ) of a resource provider p as the ratio between the sum of the demands for its resource(s) and the available quantity of resource(s), i.e. $\mu_j = \frac{\sum_{i=1}^{n} d_{ij}}{r_j}$, if it provides only one resource $j \in M$. Contrarily, if it provides more than one resource, the congestion level is the maximum between the level of each resource it provides. If $\mu_p > 1$ the resources provided by the provider p are congested.

■ **Example 6.1** Let us consider the problem (D, R, γ) with $R = (100, 30, 600, 80)$ and $D = \begin{bmatrix} 20 & 10 & 160 & 40 \\ 20 & 25 & 488 & 64 \\ 30 & 10 & 160 & 40 \end{bmatrix}$ and $\gamma = (1, 1, 1)$. The resources are resource blocks (RB), link (Gbps), RAM in GB and CPU. The resource providers are 3: radio, network link and cloud providers.

The congestion level is:

- $\mu_1 = \frac{20+20+30}{100} = 0.7$,
- $\mu_2 = \frac{10+25+10}{30} = 1.5$,
- $\mu_3 = max\{\frac{160+488+160}{600}, \frac{40+64+40}{80}\} = 1.8$.

■

Each resource provider has to take into account the priority index so that the up to now considered single and multi-resource allocation rule has to be adapted to the context. In this work we consider a simple algorithm 4 to adapt the allocation rules. We suppose that the priority index takes integer value from 1 to s, where s is the priority index of the lower priority required service, and lower value of γ corresponds to higher service priority.

For the sake of illustration, from now on we consider a reference scenario with 3 resources providers ($P = \{1, 2, 3\}$) providing radio, link and cloud resources. To make the notation clearer from now on we use the subscript r for radio ($p = 1$), l for link ($p = 2$) and c for cloud ($p = 3$).

6.1.2 Cascading Resource Allocation (CRA)

The first algorithm we propose follows a cascading approach, i.e., each resource provider sends to the following one the information about its allocation, and passing through all the providers the allocation is adjusted taking into account the congestion level of each resource. In our scenario, the order we follow is radio-link-cloud, as presented in

Figure 6.2. The step of the algorithm are depicted between the parenthesis and described in the following.

(1) When a new demand arrives, each provider receives the information about the demand for the resource it provides, i.e., a column or a sub-matrix of the demand matrix, depending on the number of resources it manages.
(2) The radio resource provider calculates the single-resource allocation using the allocation rule that it prefers.
(3) The radio provider sends the vector $x_r = (x_{r_1}, ..., x_{r_n})$ containing the information about the demand fraction allocated to each user to the link provider.
(4) The link resource provider checks if it can allocate the same fraction of the radio resource, i.e., it checks if $\sum_{i=1}^{n} d_{ij} x_{r_i} \leq r_j$ with j equal to the link resource. If this is possible it allocates the resources using the x_r (i.e., $x_r = x_l$) otherwise it calculates a new allocation such that $x_{l_i} \leq x_{r_i}, \forall i \in N$.
(5) The link resource provider sends the vector containing the information about the demand fraction allocated to each user to the cloud resource provider.
(6) The cloud resource provider checks if it can allocate the same percentage of the link resource. If this is possible then $x_l = x_c$, contrarily it calculates a new allocation such that $x_{c_i} \leq x_{l_i}, \forall i \in N$.
(7) The cloud provider sends the vector containing the information about the demand fraction allocated to each user to the link resource provider and to the radio resource provider that reallocate the resources. This step can be avoided if the vector x_r is admissible for each resource.

■ **Example 6.2** Let us consider the same problem (D, R, γ) of Example 6.1. The algorithm's steps are:

(1) The radio resource provider receives the demand vector $(20, 20, 30)$, the link resource provider receives the demand vector $(10, 25, 10)$ and cloud resource provider receives the demand matrix $\begin{bmatrix} 160 & 40 \\ 488 & 64 \\ 160 & 40 \end{bmatrix}$.
(2) The radio resource provider calculates the allocation. In this case there is no congestion so $a_r = (20, 20, 30)$ and $x_r = (1, 1, 1)$.
(3) The link resource provider receives the vector x_r.
(4) The link resource provider calculates the allocation. In this case there is congestion so x_r is not an admissible solution. The provider uses an allocation rule; if it is for example the MMF one, the allocation is $a_l = (10, 10, 10)$ and $x_l = (1, 0.4, 1)$.
(5) The cloud resource provider receives the vector x_l.
(6) The cloud resource provider checks if x_l is admissible:
   - $160 \cdot 1 + 488 \cdot 0.4 + 160 \cdot 1 \overset{?}{<} 600 \rightarrow$ yes
   - $40 \cdot 1 + 64 \cdot 0.4 + 40 \cdot 1 \overset{?}{<} 80 \rightarrow$ no

   Due to the fact that the proposed x_l is not admissible; the cloud provider calculates a new allocation, taking into account that for each user i the upper bound for x_{c_i} is x_{l_i}. For example using the DRF rule we get $x_c = (0.68, 0.4, 0.68)$ and $a_c = \begin{bmatrix} 108.8 & 27.2 \\ 195.2 & 25.6 \\ 108.8 & 27.2 \end{bmatrix}$.
(7) The cloud resource provider sends the vector $x_c = (0.68, 0.4, 0.68)$ to the link and radio resource providers that re-allocate the resources obtaining $a_r = (13.6, 8, 20.4)$ and $a_l = (6.8, 10, 6.8)$.

■

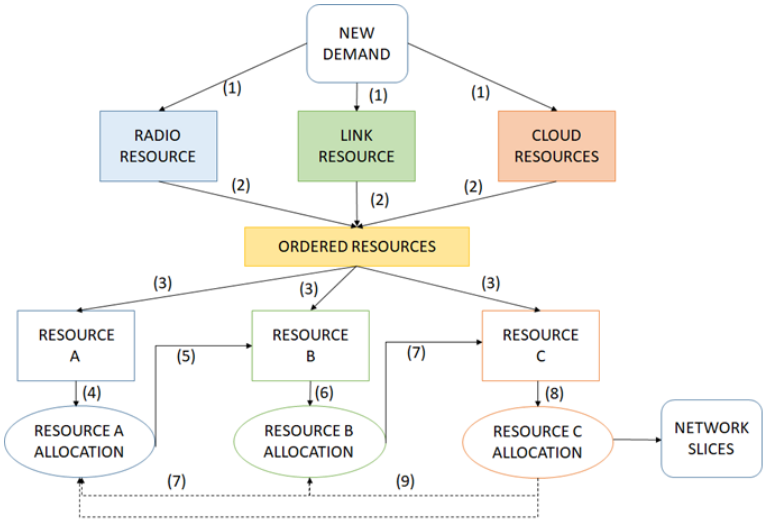


Figure 6.3: OCRA algorithm. The steps not always necessary are drawn in dashed line.

6.1.3 Ordered Cascading Resource Allocation (OCRA)

In the presence of a multi-domain orchestrator that is able to schedule how resource allocation takes, one can partially avoid resource re-allocation. Before each decision is taken, the orchestrator asks or receives the congestion level of each resource (radio, link and cloud) and re-order the resources. This can not guarantee to bypass the re-allocation for all resources, but it can strongly reduce its impact on the solution (see Example 6.3). The algorithm is similar to the CRA one but with two more steps, step (2) and (3) below (see Figure 6.3):

(1) Step (1) of CRA.
(2) Each resource provider calculates the congestion level and sends it to the multi-domain orchestrator.
(3) The multi-domain orchestrator orders the resources from the most to the least congested ones, and it sends the order to the resource providers. In the following steps, the resources are named A, B, C according to the order defined by the multi-domain orchestrator.
(4) Step (2) of CRA replacing radio resource with resource A.
(5) Step (3) of CRA replacing radio resource with resource A and link resource with resource B.
(6) Step (4) of CRA replacing radio resource with resource A and link resource with resource B.
(7) Step (5) of CRA replacing link resource with resource B and cloud resource with resource C. If x_A is not admissible for resource B, x_B is sent to resource A, that provides to re-allocate the resource.
(8) Step (6) of CRA replacing link resource with resource B and cloud resource with resource C.
(9) If x_B is not admissible for resource B, x_C is sent to providers for A and B to re-allocate the resources.

■ **Example 6.3** Let us consider the same problem (D, R, γ) of Example 6.1. The algorithm's steps are:

(1) Each resource provider receives the demand vector/matrix.

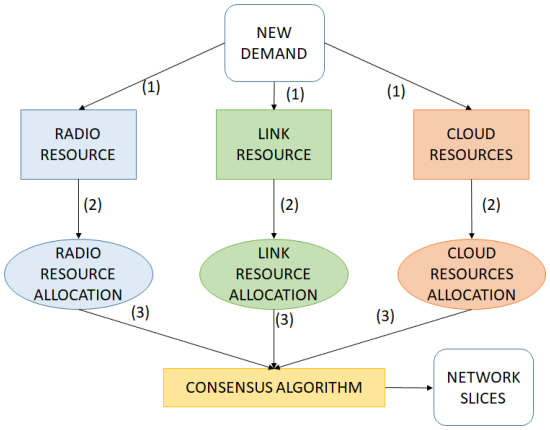


Figure 6.4: PRA algorithm - PRA-1 using the 1-phase consensus algorithm, PRA-2 using the 2-phases consensus algorithm

(2) Each resource provider calculates the congestion level: $\mu_r = 0.7$ $\mu_l = 1.5$, $\mu_c = 1.8$.
(3) The multi-domain orchestrator sends the resources order to the resource providers. The first resource to be allocated is the cloud followed by the link and the radio.
(4) The cloud resource provider calculates the allocation, for example using the DRF: $x_A = (0.67, 0.412, 0.67)$, $a_c = \begin{bmatrix} 107.2 & 26.8 \\ 201.1 & 26.4 \\ 107.2 & 26.8 \end{bmatrix}$
(5) The link resource provider receives the vector x_A.
(6) The link resource provider checks if x_B is admissible: $10 \cdot 0.67 + 25 \cdot 0.412 + 10 \cdot 0.67 \overset{?}{<} 30 \rightarrow$ yes. The allocation is: $x_B = x_A$, $a_l = (6.7, 10.3, 6.7)$.
(7) The radio resource provider receives the vector x_B.
(8) The radio resource provider accepts the proposed x_B because the resource is not congested. The allocation is: $a_r = (13.4, 8.24, 20.1)$.
(9) Step not necessary because no resource re-allocation.

■

It is worth noting that with this algorithm one cannot always avoid re-allocation. In fact if, in Example 6.3 we increase the value of d_{22} from 25 to 30, the order of the resource, based on the congestion level, remains the same ($\mu_l = 1.67$), but if the cloud provider proposes the allocation $x_A = (0.2, 1, 0.2)$, the link provider cannot accept it because $10 \cdot 0.2 + 30 \cdot 1 + 10 \cdot 0.2 > 30$. This shows that the re-allocation is not always avoided with this algorithm, but at least its negative impact is decreased. A numerical analysis of the occurrence of re-allocation is made in section 6.2.3.

An alternative algorithm, that avoid the presence of a multi-domain orchestrator can be obtained assigning the role of orchestrator to the provider usually mostly congested. In this case, when it is the mostly congested the calculus of its the allocation can be done in parallel to the sending of the ordered list of resources.

6.1.4 Parallel Resource Allocation (PRA)

In the previous proposed algorithms the computation of the resource allocation is done following a weakly distributed manner. Indeed, the resource allocation is computed according to a defined sequence among the resource providers, which implies a high dependency and a low collaboration degree between providers. Thus, the computation time

required by these algorithms is related to the resource provider with the highest response time. To limit the impact of such a situation, we design a fully-distributed algorithm which allows to increase the level of parallelism to compute the allocation and to reduce the computation time. Contrary to the two preceding algorithms, the idea of the algorithm is to allow each provider to compute its own allocation, then all the resource providers exchange their allocation and use a distributed consensus approach [91] to obtain the final allocation.

The algorithm depicted in Figure 6.4 is:

(1) When a new demand is formulated, each provider receives the information about the demands for the resource it provides, i.e., a column or a sub-matrix of D, depending on the number of resources it manages.
(2) Each resource provider calculates the allocation.
(3) A consensus algorithm provides the final allocation.

We propose two different consensus algorithms. The first one has the property of being fast, but it does not guarantee to saturate at least one of the congested resources, so it is not Pareto efficient as we prove later (Section 6.2.2). The second one introduce an additional information exchange to the process, but it guarantees to saturate at least one of the congested resources.

The first consensus algorithm is a 1-phase algorithm (PRA-1); each resource provider diffuses to all the other ones the value of x, and the allocations are obtained in the following way: $(min\{x_{r_1}, x_{l_1}, x_{c_1}\}, \cdots, min\{x_{r_n}, x_{l_n}, x_{c_n}\})$. The non-saturation of the resources can happen when there exists at least one user for which the dominant resource, i.e., the resource in percentage most requested by the user, is not the one with higher congestion level (see Example 6.4).

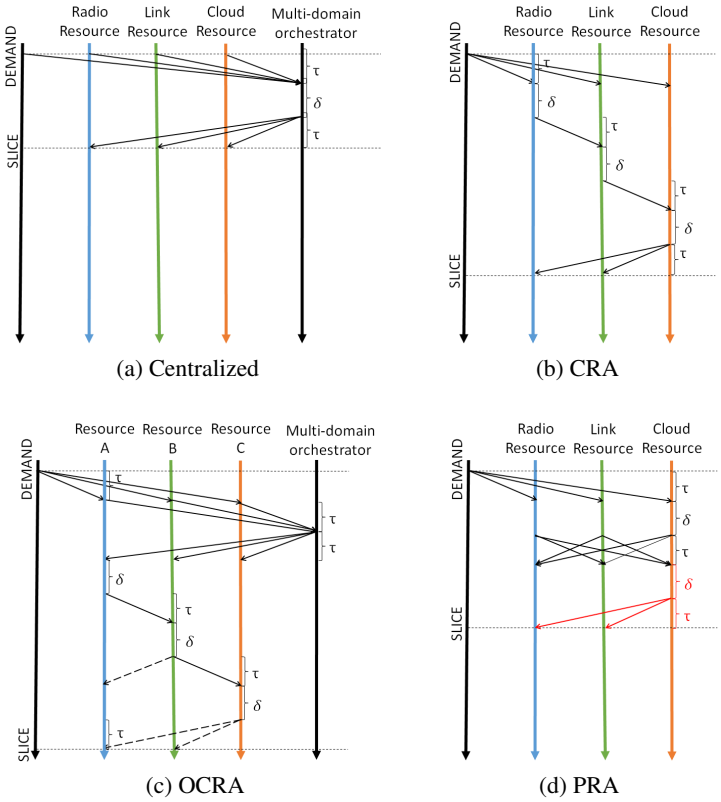
The second algorithm is a 2-phase algorithm (PRA-2); each resource provider diffuses (i) the congestion level and (ii) the resource share of each resource it provides for each user, i.e., $rs_i = \{\frac{d_{ij}}{r_j}\}\ \forall i \in N$ and for each resource j it provides. The provider with the most congested resource can identify itself and calculate the value of x using a multi-resource approach. In fact, the information about the resource share allows the provider to take into account the capacity constraints; moreover the optimization objective is decided by the provider following its fairness goal. PRA-2 guarantees to provide a Pareto optimal allocation as it is proven later.

■ **Example 6.4** Let us consider the problem (D, R, γ) of Example 6.1. The value of x calculated in a parallel way is $x_r = (1,1,1)$ for the radio resource, $x_l = (1, 0.4, 1)$ for the link resource using the MMF allocation rule and $x_c = (0.67, 0.412, 0.67)$ for the cloud resource, using the DRF allocation rule.

Using the 1-phase consensus algorithm (PRA-1) each resource provider allocates the resources using $x = (0.67, 0.4, 0.67)$. The allocations are: $a_r = (13.4, 8, 20.1)$, $a_l = (6.7, 10, 6.7)$, $a_c = \begin{bmatrix} 107.2 & 26.8 \\ 195.2 & 25.6 \\ 107.2 & 26.8 \end{bmatrix}$ and the resource used is $(41.5, 23.4, 409.6, 79.2)$. This shows that the saturation of the resources is not guaranteed when we use the 1-phase algorithm. In fact, for user 2 the dominant resource is the link resource but the resource with higher congestion level is the cloud one.

When we use the 2-phase algorithm (PRA-2), the three resource providers diffuse the following information:

- $rs_r = (0.2, 0.2, 0.3)$, $\mu_r = 0.7$.



(a) Centralized (b) CRA

(c) OCRA (d) PRA

Figure 6.5: Involved signaling for the centralized algorithm and the proposed distributed algorithms, as a function of time, under the hypothesis of equal transfer times (τ) and equal allocation computing times (δ). The dashed arrows indicate not necessary steps, and the red arrows correspond to extra steps of the 2-phase consensus algorithm[a].

[a] With the OCRA algorithm, if the most congested provider assumes the role of multi-domain orchestrator the third τ is in parallel with the first δ.

- $rs_l = (0.33, 0.83, 0.33)$, $\mu_l = 1.5$.
- $rs_c = \begin{bmatrix} 0.27 & 0.5 \\ 0.81 & 0.8 \\ 0.27 & 0.5 \end{bmatrix}$, $\mu_c = 1.8$.

The cloud resource is the most congested one and the cloud provider calculates the value of x. For example, if it choose to use a proportional approach (equalizing the x of each tenant), the solution is $x = (0.556, 0.556, 0.556)$, $a_r = (11.12, 11.12, 16.68)$, $a_l = (5.56, 13.9, 5.56)$, $a_c = \begin{bmatrix} 89 & 22.2 \\ 271.3 & 35.6 \\ 89 & 22.2 \end{bmatrix}$. ∎

6.2 Performance evaluation

In this section we provide a qualitative and quantitative analysis of the proposed algorithms. Section 6.2.1 provides an analysis in terms of delay budget, Section 6.2.2 highlights advantages and disadvantages of each algorithm, and in Section 6.2.3 we numerically compare the algorithms.

Algorithm	Best case	Worst case	Message complexity
Centralized	$2\tau+\delta$	$2\tau+\delta$	$2p+1$
CRA	$(p+1)\tau+\delta$	$(p+1)\tau+p\delta$	$3p-2$
OCRA	$(p+2)\tau+\delta$	$(p+3)\tau+p\delta$	$[4p-1, \frac{(p)(p+7)}{2}-1]$
PRA-1	$2\tau+\delta$	$2\tau+\delta$	p^2
PRA-2	$3\tau+2\delta$	$3\tau+2\delta$	p^2+p-1

Table 6.1: Delay budget and message complexity - General case with p resource providers.

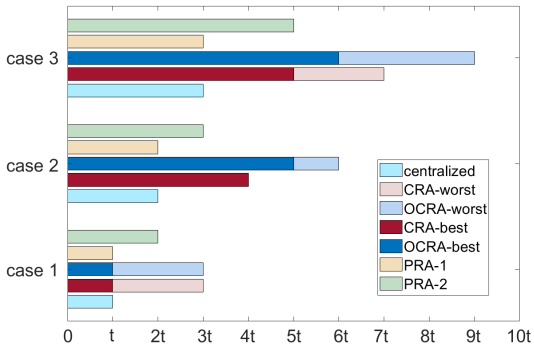


Figure 6.6: Comparison of delay budgets with $p = 3$. Case 1: $\tau \ll \delta, t = \delta$. Case 2: $\delta \ll \tau, t = \tau$. Case 3: $\tau = \delta = t$.

6.2.1 Delay budget

We are here interested in estimating the delay budget of each algorithm, i.e., the global time between the submission of a slice demand and the moment in which the slice is allocated. Delay contributions in slice provisioning are the transmission delay and the propagation delay for each message, and the allocation computation time. The time for checking if x is admissible can be considered negligible. We do also assume in the following that the transmission delay to be negligible, given the likely short message size in stake.

Figure 6.5 shows delay budget diagrams for the three proposed algorithms, and an arbitrary centralized approach where a multi-domain orchestrator receives the tenants demand and computes the allocation as a one-shot operation. Under the simplification that propagation delays are all roughly equal to a value τ and all allocation computing times are equal to δ, we obtain the estimation of the delay budget in Table 6.1 for the general case with p resource providers. We report the value of the delay budget in the best and worst case; these two values do not coincide in case of cascading approaches: the best case is the one in which only one allocation is calculated and is admissible for all the other resource providers, while the worst one is in case an allocation has to be calculated by each resource provider.

Clearly the centralized approach is the one with lower delay budget together with the first distributed approach. The algorithm closer to the centralized approach is PRA-1. Cascading approaches have a higher figure; note that while for the centralized and PRA approaches the value of delay budget does not depend on the number of resource providers p, for cascading approaches it does. Figure 6.6 compares the delay budget of all the approaches with 3 resource providers, in 3 different cases: (case 1) the propagation delay is negligible with respect to the computing time, i.e., $\tau \ll \delta$, (case 2) the reverse case, i.e., $\delta \ll \tau$ and (case 3) the two times are comparable - we plot the case $\tau = \delta$.

Algorithm	Advantages	Disadvantages
Centralized	Low delay budget	Multi-domain orchestrator High confidentiality disclosure
CRA	No multi-domain orcherstrator	Re-allocation
OCRA	Rarely re-allocation	High delay budget Multi-domain orchestrator
PRA-1	No multi-domain orchestrator Low delay budget Independent radio allocation	Pareto optimal solution not guaranteed High message complexity
PRA-2	No multi-domain orchestrator Low delay budget Independent radio allocation	High message complexity Low confidentiality disclosure

Table 6.2: Pros vs cons of studied algorithms.

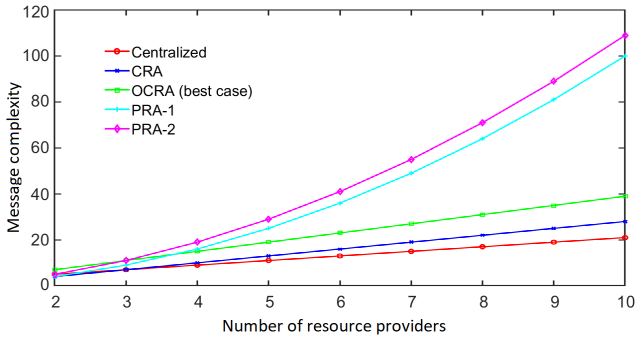


Figure 6.7: Messages complexity as function of the resource provider

6.2.2 Pros and cons

Let us draw advantages and disadvantages of the different algorithms. Table 6.2 summarizes the following observations.

Choosing a centralized approach we have the advantages of a low delay budget in the creation of the slice, due to the fact that the decision is taken atomically by a single entity. Meanwhile the fact of having centralization at a multi-domain orchestrator can be seen as an obvious drawback in terms of reliability from the one hand, and confidentiality from the other hand as each provider has to share possibly sensible information, as for example the quantity of resource available in its domain. The presence of a multi-domain orchestrator is also necessary for the OCRA approach, in order to order the resource providers. In this case it has only a function of dispatcher[1]. All the other approaches have the advantage of non having the necessity of such a centralized orchestrator.

Concerning cascading approaches (CRA, OCRA) they have the disadvantage of re-allocating resources during the slice provisioning; this is expected to be highly reduced with the OCRA approach.

Advantages of parallel approaches are (i) the low delay budget, due to the simultaneously computation of the allocation and diffusion of the information, and (ii) the possibility to independently allocate some resources. For example, this can be useful for the radio resource for which the hypotheses of linearly dependency with the other resources may appear less acceptable with some radio scheduling protocols.

[1]Note that for OCRA it is however possible to avoid the presence of the multi-domain orchestrator using a distributed approach to exchange the information about the resources congestion level, however impacting performance

Allocation rule	No re-allocation	One re-allocation	Two re-allocation
MMF	82.7%	17%	0.3%
Mood value	100%	0%	0%
Proportional	100%	0%	0%

Table 6.3: Occurrence of re-allocations with the OCRA algorithm using common single-resource rules.

Allocation rule	Percentage of non-optimal solutions
MMF	57%
Mood value	72%
Proportional	56%

Table 6.4: Percentage of non-Pareto efficient solutions using the PRA-1 algorithm.

If distributed approaches have good behavior in terms of delay budget compared to the cascading ones, considering the number of messages that have to be exchanged the judgment is reverse. From Table 6.1 and Figure 6.7 we can see that the number of exchanged messages grows quadratically with the number of providers p. In case in which $p = 10$ the number of the exchanged messages is between 21 and 30 for the centralized and cascading approaches, while it is 100 and 109 for the two distributed ones. This is the price to pay when we distribute the calculus of the allocation to avoid a single point of failure.

Among the disadvantages, for the PRA-1 we find also the possibility to get a solution that is not Pareto efficient. In this respect, we can state the following Theorem.

Theorem 6.2.1 CRA, OCRA and PRA-2 algorithms provide Pareto-optimal solutions.

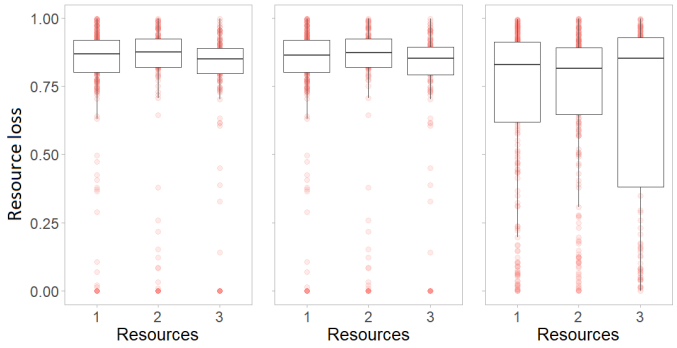
Proof. CRA and OCRA and algorithms provide Pareto-optimal solutions because the allocation coincides with the one proposed by one provider that selects a Pareto efficient allocation rule. The PRA-2 algorithm provides a Pareto efficient solution because the most congested provider calculates a multi-resource allocation; it solves an optimization problem where the objective function depends on its fairness goal and the capacity constraints are written considering the resource share of each user for each resource. The algorithm PRA-1, using the minimum value for each component allows the increasing of the allocation of one tenant without decreasing the one of the other. Let us consider the example 6.4. If we increase the allocation of tenant 2 from 0.4 to 0.412 we obtain the allocation proposed with the OCRA in Example 6.3. Thus, it is possible to increase the allocation of tenant 2 without modifying the one of the others (allocation not Pareto efficient). ■

6.2.3 Numerical analysis

We present a numerical analysis to measure (1) the occurrence of reallocation using the OCRA algorithm, (2) the occurrence of inefficient solutions for the PRA-1 algorithm, and (3) the distance of the proposed decentralized approaches from the centralized one. The analysis for (1) and (2) is done considering services with the same priority.

Occurrence of re-allocation

The aim here is to understand if there is a real gain using an ordered approach, i.e., if the re-allocation of the resources is reduced and consequently the delay budget induced by allocation computation. We generate 300 problems with 3 tenants, 3 resources belonging



(a) MMF (b) Proportional (c) Mood value

Figure 6.8: Percentage of resource loss.

to 3 providers, randomly associating a level of congestion between 0.1 and 2 for each provider. Table 6.3 shows the results of the simulations when all providers use the same allocation rule (Proportional, Mood value, MMF). We can see that there is a real gain in using an OCRA approach because with the proportional allocation and the mood value we have no re-allocations, while with the MMF there are situations in which one re-allocation is needed, but two are needed only for a negligible number of cases.

Percentage of inefficient solutions in PRA-1

We test here the efficiency of the solutions when we use the PRA-1 algorithm. Using the same data generated for the previous simulations, we calculate the percentage of time in which the PRA-1 algorithm does not provide a Pareto-optimal solution (Table 6.4) and we estimate how much is the loss for the tenants in term of resources (Fig. 6.8). Clearly, PRA-1 has high probability to provide allocations that are not Pareto-efficient. When providers use the same allocation rule, more than half of the time the produced allocation is not Pareto efficient. Furthermore the resource loss is high. The median value in percentage, obtained in the boxplot (Fig. 6.8) belongs to the interval of $[0.8, 0.9]$.

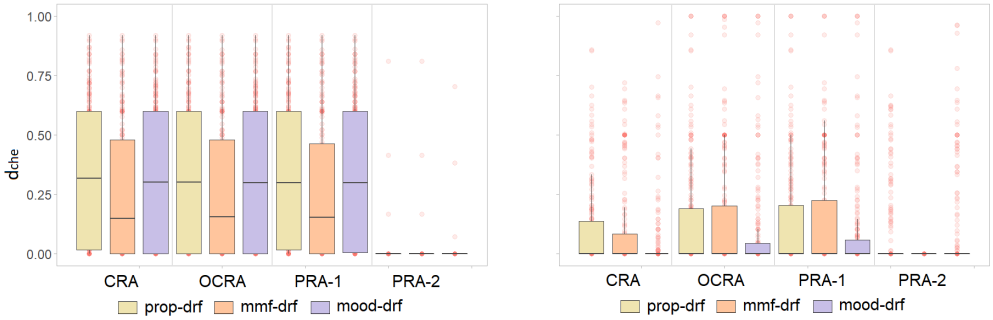
Distance from a centralized approach

We introduce a measure of the distance between a centralized approach and a decentralized one. A simple measure we can consider is the Chebyshev distance (or L_∞ metric) defined as follows.

Definition 6.2.1 The Chebyshev distance between two vectors y_1 and y_2 is $d_{che} = max_i|y_{1_i} - y_{2_i}|$ and it is equal to the limits of the L_p metric.

In our case, considering a solution vector obtained with a centralized approach and one with a decentralized one, the measure indicates the gain (or loss) of the user that obtains the maximum gain (or loss) when a decentralized approach is used. This measure provides an estimation of the satisfaction (unsatisfaction) of the users in adopting a decentralized approach.

We simulate 200 problems with 5 tenants, taking inspiration from Amazon EC2 instances [11]; we select those templates with different 'instance type'('General Purpose', 'Computer Optimized', 'Memory Optimized', 'Accelerated Computing'and 'Storage Optimized') and we consider 3 resources belonging to 2 providers (CPU and memory for the cloud provider link capacity in Gbps for the network link provider), a level of congestion between 0.1 and 1.5 for each provider, and both the case in which the tenants have the



(a) Single-class (b) Multi-class

Figure 6.9: Chebyshev distance.

same priority and belong to the same class (single-class) and the case in which the tenants have different priorities and belong to different classes (multi-class). In this second case we associate to the different Amazon templates a type of service (URLLC with priority 1, eMBB with priority 2, mMTC with priority 3, best effort with priority 4), as done in chapter 5.

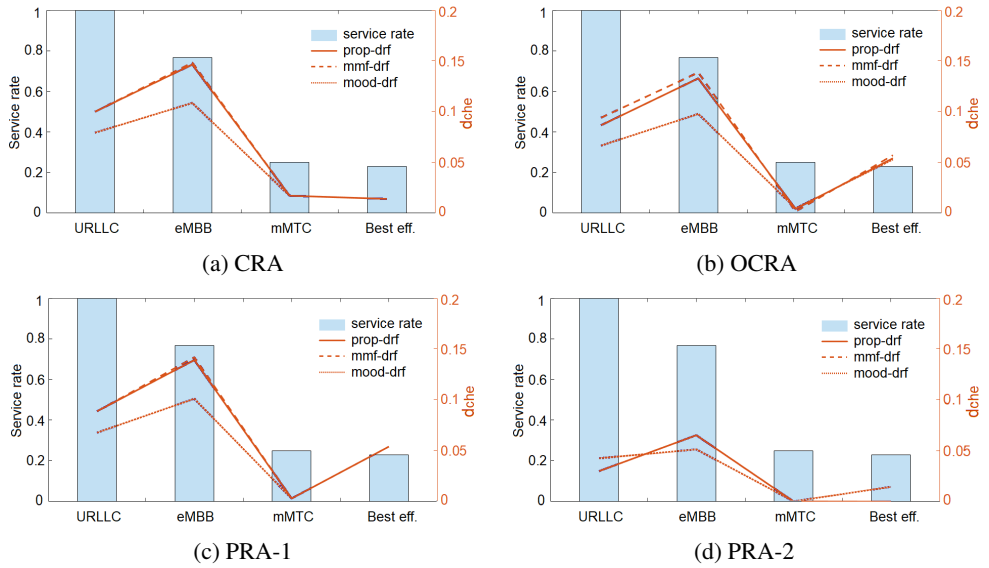
Figure 6.9 shows the boxplot of the distance for each algorithm, using different combinations of allocation rules, when the centralized approach uses the DRF rule. When the priority is the same for each tenant there are users that can gain or loose a lot when the providers adopt as decentralized approach the CRA, OCRA and PRA-1. In the single-class case, the distance is reduced using the PRA-2 approach because the proposed allocation is calculated as a multi-resource allocation taking into account the information provided by each provider. In the multi-class case we notice a performance improvement of the decentralized algorithm. In fact, in this case, the differences emerge only for the group of tenants belonging to the same priority class for which the remaining resource, after that tenants with higher priority are fully served, is not enough. In this case due to the small cardinality of the subset of users belonging to same class, there is a high probability that the decentralized solution is close to the centralized one.

We then consider the distance measure inside each group of services and the service rate (Figure 6.10). The decentralized algorithm always serves the users with higher priority and the service rate decreases with the service priority. On average, the distance increases decreasing the service priority, but a decrease of the distance is possible because (i) services with low priority have high probability not to be served both with the centralized and decentralized approaches (Fig. 6.10) and (ii) as already said, if the cardinality of the last served group is small the decentralized and centralized solutions can be close.

6.3 Dealing with run-time constraints

We propose different decentralized algorithms to slice the network in a given time frame. We want now to give some ideas concerning how to enrich our algorithms designing policies and mechanisms for long term resource allocation and taking into account also Service Level Agreement (SLA) constraints [60]. In particular we discuss how to guarantee the continuity of the service, i.e., when a tenant is served, it has to be served for the required time

Under this setting, at the given time slot t, the resource allocation problem is a tuple



(a) CRA (b) OCRA

(c) PRA-1 (d) PRA-2

Figure 6.10: Chebyshev distance average and service rate.

$(D_t, R_t, \gamma_t, \tau_t)$ where D_t is the demand matrix at time t, γ_t is a vector containing the priority index of each tenant, R_t is the available resource at time t and τ_t is a vector containing the required time of the service asked by tenants. The demands considered at each time t frame include the one requiring the service at time t and the one requiring the service before t but not still served or in service.

The structure of the algorithms remains similar but we need to adapt them so that if we start to serve a tenant at time t, it is served for a required time interval so in $[t, t+\tau]$. The idea is that when a tenant is served its priority is automatically increased setting the value of γ equal to to highest priority value. In this way it is certainly served but the allocated resources can be increased or decreased depending on the resources availability.

The risk of using this algorithm is that tenants asking for low-priority services remain unserved for a long time. To guarantee time-fair allocation and reduce the average waiting time it is possible to consider an expiration time for each required service, after which the priority is increased and the tenant served.

6.4 Summary

We proposed algorithms to decentralize 5G slice provisioning, two using a cascading approach and two a parallel approach. We extensively compared them, showing pros and cons, also with respect to a centralized approach. In particular, decentralized approaches have the advantage to eliminate the presence of a multi-domain orchestrator typical of centralized approaches but they provide allocation less fair compared to centralized approaches. In fact, some user can gain (or loose) in the percentage of allocated resources when decentralized approaches are adopted. Furthermore, parallel approaches have advantages in terms of delay budget but high message complexity compared to cascading approaches and cascading ones have higher delay budget but lower message complexity compared to parallel approaches. Ideas on how to deal with run-time constraints are presented. The work can be extended considering the fault tolerance problematic and considering providers with different priority.

7. Conclusions and perspectives

7.1 Conclusions

The aim of this thesis is to provide a formal analysis of the resource allocation problem in congested networked systems, i.e., in the case in which the resource/es is/are not sufficient to fully satisfy the users demands, focusing on the fairness of the allocations. In fact, when there is/are enough resource/es for each user, they can be fully satisfied while in congested situations it is important to understand how to partition the resource/es in a way that do not advantage or disadvantage users. The work covers both the case in which only one resource is required and the case in which more than one resource is demanded, as predicted by the new 5G environment through the network slicing concept. Furthermore, in the thesis we deal with different scenarios and approaches to solve the allocation problem. In particular, we always consider as baseline scenario the static one, i.e., when the evolution of the demand and of the resources with the time is not considered and the aim is to propose fairness allocation in a certain instant of time. We complete the analysis considering dynamic scenarios, where users can move from one provider to the other or where allocations on a time-frame are considered. In case of multi-resource allocation problems both centralized and decentralized approaches are analyzed. This work provides also a several simulations, covering all the discussed topics, and supporting theoretical analysis.

We firstly analyzed the single-resource case, going to cover the cases of complete and partial information, not formally studied in the literature. If for the incomplete scenario the decision maker is the only actor having information about the available resource and the users demand and if under this condition some fairness property are likable for the allocation (e.g., the envy-freeness and the strategy-proofness typical of the MMF allocation or the satisfaction equality typical of the weighted proportional allocation), in case users are aware of other users' demand and the available resource the classical approach do not work more. We discuss how the measure of the user satisfaction has to be re-defined in complete and partial information settings and how this new satisfaction measure leads

us to define a new allocation rule, we called Mood value and a new measure of fairness. These are appropriate both for the complete and partial information scenarios.

We then moved to analyze multi-resource allocation problems, providing a general framework to select fair allocations. In fact, as consequence of the analysis of single-resource case, we realized that it is possible to select as satisfaction measure simply the ratio of allocated resource (eventually weighted by the dominant share) if users has no information about the network or the new satisfaction measure proposed in Chapter 3 called the Player Satisfaction (eventually weighted by the dominant share) if users are aware of the network information. Furthermore we add a new dimension in the analysis, considering an objective function that let the freedom to the decision maker to select the allocation more appropriate to its fairness goal. In particular, the fairness concept can go from the egalitarian to the utilitarian one. We also show how the proposed framework generalize well-known resource allocations and that a multi-resource approach has better performance compared to the use of single-resource approaches for each resource: it does not allocate unneded surplus of resources and it can allow for idle capacity to support traffic peaks.

The considered application for multi-resource approaches is the network slicing, introduced by the new 5G technology where heterogeneous service has to be provided and so the network has to be partition in slices optimized for the specific required service. The multi-resource approach presented presupposes the presence of a centralized provider (the network slice provider), that can manage each resource. Nowadays each part of the network is managed separately and the presence of a centralized orchestrator may be not viable. We so propose decentralized algorithms analyzing and comparing them together and with centralized approaches to show the advantages and disadvantages of each of them.

The network slicing problem is also studied in chapter 5 from a dynamic point of view, proposing two scheduling algorithms able to take into account some Service Level Agreement requirements, as the allocation of the minimal demand and the time-fairness. Similar algorithms can be potentially used also for decentralized approaches and can be a starting point to customize network slicing allocation performance toward more specific SLA requirements.

Concluding, this work shows how it is difficult to find a consensus about which is the fairest allocation in a resource allocation problem and it aims to provide guidelines for understand which are the best metric to use, depending on the context, to verify the fairness of an allocation and to help the decision-maker to select the allocation that fits with its fairness goal. The analyzed scenarios are not strictly related to networking and computing frameworks and they can be extended to many other fields. For example the complete information settings perfectly model auctions where the good can be divided among the bidders and bids are public and submitted contemporaneously.

7.2 Perspectives

The work done in this thesis can be extended and completed in multiple directions. Firstly, looking at the Figure 1.1 in introduction we can notice that the analysis of multi-resource in partial information sharing allocation is missing. In this case allocation rules, as the DRF [41], are obtained as results of an optimization problem. Differently from the case single-resource in which there are rules with a direct formula to calculate the allocation, the estimation of the error becomes complex. Concerning the last contribution, algorithms

dealing with run-times constraints are only mentioned and they require a more formal and in-depth analysis together with an evaluation via simulations.

Another interesting direction of research could be to investigate the relationship between the mood value and the risk of collusion, i.e. the possibility that using the mood value rule users are encouraged to form a coalition with other users to obtain a greater resource.

Regarding the case of multi-resource allocations, it is interesting to investigate the case study in which some resources are not divisible and the case in which the relationship between resources is not linear. For the first case the OWA operator should still work well and for the second case a hint of how to generalize the allocation rule is given in the appendix but it must be further investigated.

Focusing on the use-case mostly considered in this work, i.e. the network slicing, it does not exist still a consensus on what exactly a network slice is. In particular it is not clear which it will be the granularity associated to the slices. We know that each slice correspond to a specific service but should we consider one slice per family of services (eMBB, mMTC, URLLC), or one slice per set of technical requirements, or one slice per vertical customer, or a combination of the them? Once there will be greater clarity in this regard it might be interesting to redo the simulation with a more realistic scenario.

The work presented can be useful as starting point for in the study of the virtual network function (VNF) orchestration, i.e., VNF placement and routing. In fact we take into consideration only the problem of multi-resource allocation producing a solution giving an amount of each resource to allocate to each tenant, independently of the infrastructure, whereas the actual embedding of each resource into a final resource partitioning, taking into consideration the geographical distribution and interconnection links of computing servers, can be considered as a separate, successive, problem.

Appendices

A. Pricing framework and implementation

To ensure that users formulate true demands, robust pricing frameworks need to be considered. A well-know mechanism used in [23] for the price implementation is the Myerson's mechanism [92] that is a truthful auction, i.e. an auction where every claimant is encouraged to give his true evaluation of the resource[1]. We can think the users demands as the bids of the auctioneers and the resource partition as the result of the auction. Called b_i the bid of user i, we would like that b_i is equal to d_i, i.e. the true demand of user i. The outcome of the mechanism is providing the allocation a and the prices (or payment rule) p. We assume quasi linear utilities expressed by $u_i = v_i\overline{a_i(b)} - p_i(b)$ where $\overline{a_i(b)} = \frac{a_i(b)}{R}$ takes value in $[0,1]$, v_i is the private evaluation per unit of resource and $p_i(b) \in [0, b_i\overline{a_i(b)}]$. Thus an agent's goal is to maximize the difference between his valuation and its payment.

The implementation of the price associated to different allocation rules is possible through the Myerson theorem [92]. To state this theorem, we need firstly some preliminary definitions.

Definition

- The tuple (a,p) is *Dominant-Strategy Incentive-Compatible (DSIC)* if truthful bidding is always a weakly dominant strategy for every bidder and if truthful bidders always obtain nonnegative utility.
- An allocation rule a is monotone if for every agent i and bids b_{-i} of the other agents different from i, the allocation $a_i(z, b_{-i})$ of user i is nondecreasing in his bid z.

The tuple (a,p) is DSIC if choosing the bid equal to the real demand for an user is the strategy that maximizes his utility, no matter what the other users do. Being the utility $u_i = v_i\overline{a_i(b)} - p_i(b)$, if the pricing rule is $p_i = b_i a_i$ then $u_i = 0$ for truth-teller and they have incentive to declare a lower demand to increase the utility. Contrarily, if the price per unit of resource is fixed user has incentive to increase the bids (i.e., the communicated demand

[1] Other types of auctions have the property of being truthful, e.g. the VCG auction [93], [94], [95]

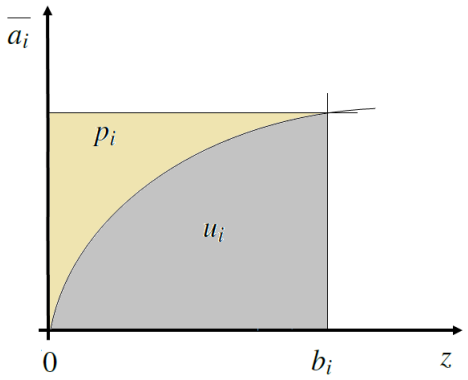


Figure 1: Price and utility interpretation

to receive an higher portion of resource). We state now the theorem, ensuring that user has no incentive to lie about their demand.

Theorem If an allocation rule a is monotone, then there is a unique payment rule p such that the mechanism (a,p) is DSIC. The payment rule is given by the following explicit formula:

$$p(b_i,b_{-i}) = b_i\frac{a_i(b_i,b_{-i})}{R} - \frac{1}{R}\int_0^{b_i} a_i(z_i,b_{-i})dz \tag{1}$$

The pricing formula has an easy interpretation. Plotting the function of $\overline{a_i}$ as function of the declared demand z, the price is the area above the curve when $z = b_i$; the area below is the utility u_i (Figure 1).

Due to the monotonicity of the allocation rules we consider, it is possible to implement the price associated to them. The price formulas for rules 1-4 are described in [23] while we provide the proof of the price formula for the mood value.

1. *Proportional rule*

$$p_i = \frac{b_i^2}{\sum_{j=1}^n b_j} - b_i + \left(\sum_{j\neq i} b_j\right)\log\left(\frac{\sum_{j=1}^n b_j}{\sum_{j\neq i} b_j}\right)$$

2. *MMF*

$$p_i = b_i\frac{\min(b_i,a_i(R))}{R} - \frac{\min^2(b_i,a_i(R))}{2R}$$

3. *Nucleolus*[2]
   It is possible to approximate the integral of the allocation curve as follows:

$$\int_0^{b_i} a_i(z)dz \approx \sum_{k=0}^{2^{n-1}}\left(x_i(k\Delta)\Delta + \frac{\Delta}{2}\left[x_i((k+1)\Delta) - x_i(k\Delta)\right]\right)$$

   where $\Delta = \frac{b_i}{2^{n-1}+1}$. The price follows from (1).

[2]The Shapley value is piece-wise linear as function of b_i, so it is possible to identify the point where the curve change slopes and consequently to calculate the price in closed-form. Transition points of the curve in case of nucleolus cannot be found in closed form so only an approximated estimation is provided as results of numerical methods.

4. *Shapley value*[2]
   The allocation a_i given by Shapley on the interval $[0, b_i]$ is piece-wise linear with respect player i's bid b_i. We know that $a_i(0) = 0$ and the derivative is a stepwise function given by:

$$\frac{\partial a_i(z)}{\partial z} = \begin{cases} \sum_{j=1}^{2^n-1} \hat{\Theta}_j & \text{for} \quad 0 < z < \hat{\Phi}_1 \\ \sum_{j=k+1}^{2^n-1} \hat{\Theta}_j & \text{for} \quad \hat{\Phi}_k < z < \hat{\Phi}_{k+1} \quad k = 1, 2, \dots 2^{n-1} - 1 \\ 0 & \text{for} \quad \hat{\Phi}_{2^{n-1}} < z < b_i \end{cases}$$

   where
   - $\Phi \in \mathbb{R}^{2^n-1}$ is the vector having as entries the image of the function

$$q(S) = \max \left\{ 0, R - \sum_{j \in N \setminus \{S,i\}} b_j \right\} \quad \forall S \in N \setminus \{i\}$$

   - Θ is the corresponding vector having as elements all the Shapley coefficients:

$$\alpha_S = \frac{s!(n-s-1)!}{n!}$$

   - $\hat{\Phi}$ is the vector Φ sorted in increasing order
   - $\hat{\Theta}$ is the vector of coefficients which corresponds to $\hat{\Phi}$

   Thus, the integral of the allocation curve can be calculated as the area under the curve summing up all the areas of triangles and rectangles. And consequently formula (1) is applied to calculate the price.
5. *Mood value*
   The integral of the allocation curve can be calculated as follows:

$$\int_0^{b_i} a_i(z)dz = \begin{cases} \frac{b_i^2}{2}, & b_i \leq min_i \\ \frac{min_i^2}{2} + min_i(b_i - min_i) + \left(R - \sum_{j=1}^N min_j\right)(b_i - min_i) - \\ \left(R - \sum_{j=1}^N min_j\right) \sum_{j \neq i} \left(max_j - min_j\right) ln \left(\frac{\sum_{j=1}^N (max_j - min_j)}{\sum_{j \neq i} (max_j - min_j)} \right), & min_i < b_i < E \\ \frac{min_i^2}{2} + min_i(E - min_i) + \left(E - \sum_j min_j\right)(E - min_i) - \\ \left(E - \sum_j min_j\right) \sum_{j \neq i} \left(max_j - min_j\right) ln \left(\frac{\sum_j (max_j - min_j)}{\sum_{j \neq i} (max_j - min_j)} \right) + \\ \left[min_i + \frac{E - \sum_j min_j}{\sum_j max_j - \sum_j min_j} (E - min_i) \right](b_i - E), & b_i \geq E \end{cases} \tag{2}$$

   and the price follows from (1).

   Proof. Recalling the definition of mood value

$$a_i = min_i + m(max_i - min_i)$$

   where:

$$min_i = \max \left\{ R - \sum_{j \neq i} b_j, 0 \right\} \qquad \text{and} \qquad max_i = \min\{b_i, R\}$$

we can write the allocation rule as

$$a_i(z) = \begin{cases} min_i + \frac{R-\sum_{j=1}^{N} min_j}{\sum_{j\neq i} max_j - \sum_{j=1}^{N} min_j + z}(z - min_i) & if \quad z < R \\ min_i + \frac{R-\sum_{j=1}^{N} min_j}{\sum_{j\neq i} max_j - \sum_{j=1}^{N} min_j + R}(R - min_i) & if \quad z \geq R \end{cases}$$

Since supposing $\mathbf{b}_{-i}$ fixed implies that player i knows the value of his minimum allocation, the function can be consequently modified as follows:

$$a_i(z) = \begin{cases} z & if \quad z < min_i \\ min_i + \frac{R-\sum_{j=1}^{N} min_j}{\sum_{j\neq i} max_j - \sum_{j=1}^{N} min_j + z}(z - min_i) & if \quad min_i < z < R \\ min_i + \frac{R-\sum_{j=1}^{N} min_j}{\sum_{j\neq i} max_j - \sum_{j=1}^{N} min_j + R}(R - min_i) & if \quad z \geq R \end{cases}$$

The integral $\int_0^{b_i} a_i(z, \mathbf{b}_{-i})dz$ inside the pricing function has to be calculated in three different cases:

(a) $b_i \leq min_i$:

$$\int_0^{b_i} a_i(z)dz = \frac{b_i^2}{2}$$

(b) $min_i < b_i < R$:

$$\int_0^{b_i} a_i(z)dz = \int_{min_i}^{b_i} a_i(z)dz + \frac{min_i^2}{2} = \frac{min_i^2}{2} + min_i(b_i - min_i) + \left(R - \sum_{j=1}^{N} min_j\right)(b_i - min_i) - \left(R - \sum_{j=1}^{N} min_j\right) \sum_{j\neq i} \left(max_j - min_j\right) ln \left(\frac{\sum_{j=1}^{N}(max_j - min_j)}{\sum_{j\neq i}(max_j - min_j)}\right)$$

(c) $b_i \geq R$:

$$\int_0^{b_i} a_i(z)dz = \frac{min_i^2}{2} + \int_{min_i}^{R} a_i(z)dz + \int_R^{b_i} a_i(z)dz$$

where the first integral is

$$\int_{min_i}^{R} a_i(z)dz = min_i(R - min_i) + \left(R - \sum_{j=1}^{N} min_j\right)(R - min_i) - \left(R - \sum_{j=1}^{N} min_j\right) \sum_{j\neq i} \left(max_j - min_j\right) ln \left(\frac{\sum_{j=1}^{N}(max_j - min_j)}{\sum_{j\neq i}(max_j - min_j)}\right)$$

and the second is

$$\int_R^{b_i} a_i(z)dz = \left[min_i + \frac{R-\sum_{j=1}^{N} min_j}{\sum_{j=1}^{N} max_j - \sum_j min_j}(R - min_i)\right](b_i - R)$$

■

B. Continuous allocation - supplementary results

We report results on the comparison of allocation rules and fairness indices related to the continuous allocation case introduced in Section 3.4.2. We analyze the behavior of the mood value and the new fairness index compared to the classical allocations and the Jain's index, as function of the level of congestion of the system, when we generate the demands using a uniform distribution.

We provide in the following Fig. 2 and 3 the results of the simulations in terms of fairness. The two figures show the differences obtained by the classical Jain's index and our new players fairness (PF) index.

In Fig. 2 we can notice that the MMF allocation is a fair allocation according to PF index under high congestion. i.e. in presence of greedy claimants. This follows from the closeness between the MMF allocation and the Mood value treating in the same way greedy claimant. Instead, when E increases, the MMF one is not fair anymore because it satisfies more the two users with less claim while it gives the minimal right to the one with bigger claim; in such cases the mood value becomes closer to the Proportional allocation, to the Shapley value and to the Nucleolus. The similarity between the Proportional allocation and the mood value is due to the fact that the correct way to measure the satisfaction of moderate players is through the DFS rate and increasing E the number of moderate players increases. It follows that the allocation equalizing the DFS rates, i.e., the Proportional one, is close to the one equalizing the PS rates of user, i.e., the Mood Value.

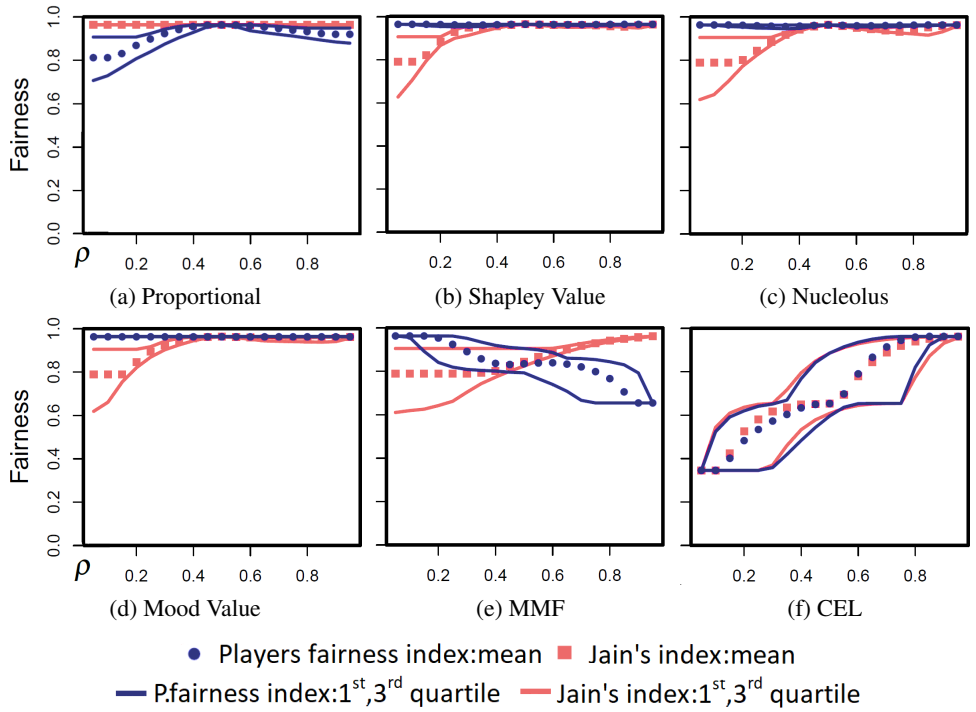


Figure 2: Fairness as a function of ρ (3 users, uniform)

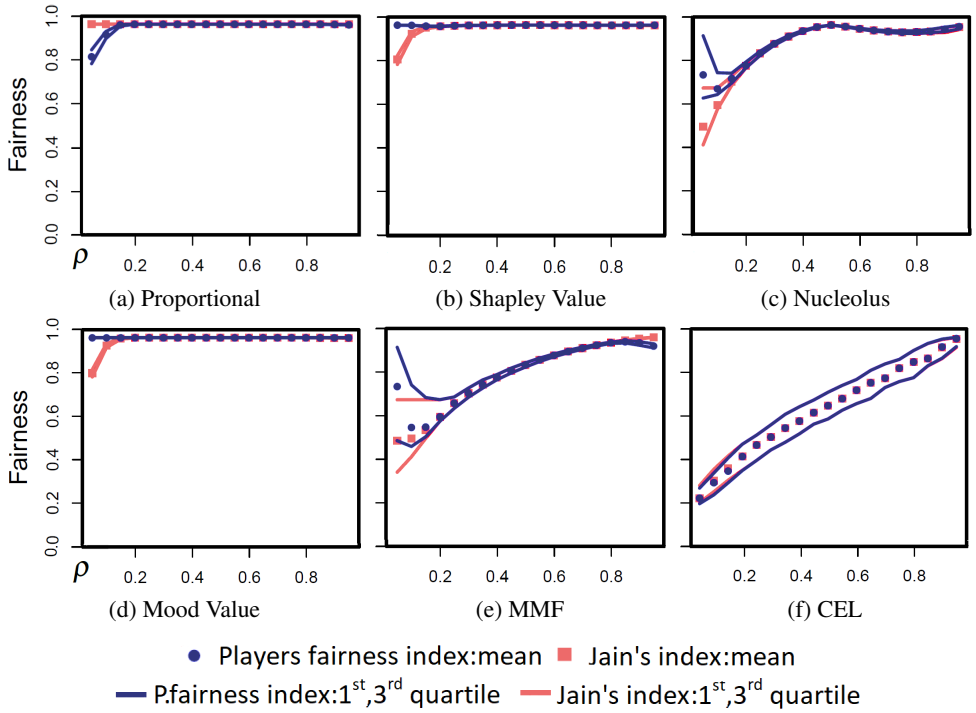


Figure 3: Fairness as a function of ρ (10 users, uniform)

C. Refinement of the MURANES model

If for most of the resource pairs we can realistically model the relationship between resources (as elaborated in Section 5.1.1), for other resources (e.g, the ones depending on particular radio schedulers) such an assumption may be too strong.

In practice, the analytical relationship between the resource can be known a priori, for example as a results of preliminary analysis of the mutual interference or dependency among pairs of resources. If the relationship between the resources is expressed by a strictly increasing monotonic function, the allocation problem can still be solved using an OWA approach, but we need to re-define the resource allocation problem. More precisely, the relationship can no longer be included in the multi-resource allocation settings, but has to be added as a constraint. In particular x is no more a vector but a matrix $n \times m$, whose components x_{ij}, with $0 \leq x_{ij} \leq 1 \ \forall i \in N$, is the percentage of resources j allocated to tenant i. The allocation matrix A corresponding to x is given by

$$\begin{bmatrix} a_{11} & \dots & a_{1m} \\ \dots & \dots & \dots \\ a_{n1} & \dots & a_{nm} \end{bmatrix} = \begin{bmatrix} d_{11} \cdot x_{11} & \dots & d_{1m} \cdot x_{1m} \\ \dots & \dots & \dots \\ d_{n1} \cdot x_{n1} & \dots & d_{nm} \cdot x_{nm} \end{bmatrix}.$$

The constraints to add to classical problem (5.1) are of type $x_{ik} = f(x_{is}), \forall i \in N, \forall k \neq s$.

The following example illustrates the relaxation of the linear dependency hypothesis.

■ **Example** Let us consider two resources, A ($j = 1$) and B ($j = 2$), such that the dependence between A and B is quadratic for each user $i \in N$. Let the matrix demand $D = \begin{bmatrix} 6 & 4 \\ 9 & 3 \end{bmatrix}$ and the resource vector $R = \begin{bmatrix} 10 & 5 \end{bmatrix}$. The problem to solve is:

$$\begin{aligned} &\text{maximize} && OWA(v) \\ &\text{subject to} && 6x_{11} + 9x_{21} \leq 10, \\ &&& 4x_{12} + 3x_{22} \leq 5, \\ &&& x_{i1} = x_{i2}^2, i = 1,2 \\ &&& 0 \leq x_i \leq 1, \forall i \in N \end{aligned} \tag{3}$$

where v is one of the OWA input described in section 5.2.2 for one of the resources. ∎

More generally we can suppose there is no relationship between resources. In this case one way can be to considerate each resource separately but to guarantee a global fairness we need to introduce a multidimensional inequality measure. Our indication in this other possible direction is to resort to the Multidimensional Generalized Gini Index [96] that is a sum over the resources of inequality indices defined as instances of OWA for every resources.

Bibliography

[1] *Oxford English Dictionary*. https://en.oxforddictionaries.com/english.

[2] *Cambridge English Dictionary*. https://dictionary.cambridge.org/dictionary/english/.

[3] Terrence E Daniel. "Pitfalls in the theory of fairness—comment". In: *Journal of Economic Theory* 19.2 (1978), pages 561–564.

[4] Richard L Sawyer, Nancy S Cole, and James WL Cole. "Utilities and the issue of fairness in a decision theoretic model for selection". In: *Journal of Educational Measurement* 13.1 (1976), pages 59–76.

[5] Frank P Kelly, Aman K Maulloo, and David KH Tan. "Rate control for communication networks: shadow prices, proportional fairness and stability". In: *Journal of the Operational Research society* 49.3 (1998), pages 237–252.

[6] Dimitri P Bertsekas, Robert G Gallager, and Pierre Humblet. *Data networks*. Volume 2. Prentice-Hall International New Jersey, 1992.

[7] Wlodzimierz Ogryczak, Hanan Luss, Michał Pióro, Dritan Nace, and Artur Tomaszewski. "Fair optimization and networks: A survey". In: *Journal of Applied Mathematics* 2014 (2014).

[8] Hyojoon Kim and Nick Feamster. "Improving network management with software defined networking". In: *IEEE Communications Magazine* 51.2 (2013), pages 114–119.

[9] Kun Zhu and Ekram Hossain. "Virtualization of 5G cellular networks as a hierarchical combinatorial auction". In: *IEEE Transactions on Mobile Computing* 15.10 (2016), pages 2640–2654.

[10] 3GPP TS 22.261 V15.7.0. *Technical Specification Group Services and System Aspects; Service requirements for 5G system*.

[11] *Amazon EC2 instances comparison*. https://www.ec2instances.info.

[12] William Jasper Spillman and Emil Lang. *The Law of Diminishing Returns: Part One: The Law of the Diminishing Increment*. World Book Company, 1924.

[13] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[14] Harold W Kuhn and Albert W Tucker. "Nonlinear programming". In: *Traces and emergence of nonlinear programming*. Springer, 2014, pages 247–258.

[15] William Karush. "Minima of functions of several variables with inequalities as side constraints". In: *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago* (1939).

[16] Srinivas Shakkottai and Rayadurgam Srikant. "Network optimization and control". In: *Foundations and Trends® in Networking* 2.3 (2008), pages 271–379.

[17] John Rawls. *A theory of justice*. Harvard university press, 2009.

[18] Jeonghoon Mo and Jean Walrand. "Fair end-to-end window-based congestion control". In: *IEEE/ACM Transactions on networking* 5 (2000), pages 556–567.

[19] William Thomson. "Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: a survey". In: *Mathematical social sciences* 45.3 (2003), pages 249–297.

[20] Marek M Kaminski. "'Hydraulic'rationing". In: *Mathematical Social Sciences* 40.2 (2000), pages 131–155.

[21] M Carmen Lucas-Estañ, Javier Gozálvez, and Joaquín Sanchez-Soriano. "Bankruptcy-based radio resource management for multimedia mobile networks". In: *Transactions on emerging telecommunications technologies* 23.2 (2012), pages 186–201.

[22] Sahar Hoteit, Stefano Secci, Rami Langar, and Guy Pujolle. "A nucleolus-based approach for resource allocation in ofdma wireless mesh networks". In: *IEEE Transactions on Mobile Computing* 12.11 (2013), pages 2145–2154.

[23] Sahar Hoteit, Mahmoud El Chamie, Damien Saucez, and Stefano Secci. "On fair network cache allocation to content providers". In: *Computer Networks* 103 (2016), pages 129–142.

[24] Guillermo Owen. *Game Theory 3rd Edition*. Academic Press, 1995.

[25] Barry O'Neill. "A problem of rights arbitration from the Talmud". In: *Mathematical Social Sciences* 2.4 (1982), pages 345–371.

[26] Robert J Aumann and Michael Maschler. "Game theoretic analysis of a bankruptcy problem from the Talmud". In: *Journal of economic theory* 36.2 (1985), pages 195–213.

[27] Lloyd S Shapley. "Cores of convex games". In: *International journal of game theory* 1.1 (1971), pages 11–26.

[28] Lloyd S Shapley. "A value for n-person games". In: *Contributions to the Theory of Games* 2.28 (1953), pages 307–317.

[29] Vincent Feltkamp. "Alternative axiomatic characterizations of the Shapley and Banzhaf values". In: *International Journal of Game Theory* 24.2 (1995), pages 179–186.

[30] René van den Brink. "An axiomatization of the Shapley value using a fairness property". In: *International Journal of Game Theory* 30.3 (2002), pages 309–319.

[31] David Schmeidler. "The nucleolus of a characteristic function game". In: *SIAM Journal on applied mathematics* 17.6 (1969), pages 1163–1170.

[32] SH Tijs and Theo SH Driessen. *The τ-value as a feasible compromise between utopia and disagreement*. Volume 8312. Report, 1983.

[33] Tamás Fleiner and Balázs Sziklai. "The nucleolus of the bankruptcy problem by hydraulic rationing". In: *International Game Theory Review* 14.01 (2012), page 1250007.

[34] John F Nash Jr. “The bargaining problem”. In: *Econometrica: Journal of the Econometric Society* (1950), pages 155–162.

[35] Ehud Kalai and Meir Smorodinsky. “Other solutions to Nash’s bargaining problem”. In: *Econometrica* 43.3 (1975), pages 513–518.

[36] Imma J Curiel, Michael Maschler, and Stef H Tijs. “Bankruptcy games”. In: *Zeitschrift für Operations Research* 31.5 (1987), A143–A159.

[37] Nir Dagan and Oscar Volij. “The bankruptcy problem: a cooperative bargaining approach”. In: *Mathematical Social Sciences* 26.3 (1993), pages 287–297.

[38] Rajendra K Jain, Dah-Ming W Chiu, and William R Hawe. “A quantitative measure of fairness and discrimination”. In: *Eastern Research Laboratory, Digital Equipment Corporation, Hudson, MA* (1984).

[39] Tian Lan, David Kao, Mung Chiang, and Ashutosh Sabharwal. *An axiomatic theory of fairness in network resource allocation*. IEEE, 2010.

[40] Anthony B Atkinson. “On the measurement of inequality”. In: *Journal of economic theory* 2.3 (1970), pages 244–263.

[41] Ali Ghodsi, Matei Zaharia, Benjamin Hindman, Andy Konwinski, Scott Shenker, and Ion Stoica. “Dominant Resource Fairness: Fair Allocation of Multiple Resource Types.” In: *Nsdi*. Volume 11. 2011. 2011, pages 24–24.

[42] Thomas Bonald and James Roberts. “Multi-resource fairness: Objectives, algorithms and performance”. In: *ACM SIGMETRICS Performance Evaluation Review*. Volume 43. 1. ACM. 2015, pages 31–42.

[43] Hervé Moulin. *Fair division and collective welfare*. MIT press, 2004.

[44] Hal Varian. “Efficiency, equity and envy”. In: *Journal of Economic Theory* 9.1 (1974), pages 63–91.

[45] Danny Dolev, Dror G Feitelson, Joseph Y Halpern, Raz Kupferman, and Nathan Linial. “No justified complaints: On fair sharing of multiple resources”. In: *proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ACM. 2012, pages 68–75.

[46] Patrick Poullie, Thomas Bocek, and Burkhard Stiller. “A survey of the state-of-the-art in fair multi-resource allocations for data centers”. In: *IEEE Transactions on Network and Service Management* 15.1 (2017), pages 169–183.

[47] Francesca Fossati, Stefano Moretti, and Stefano Secci. “A mood value for fair resource allocations”. In: *2017 IFIP Networking Conference (IFIP Networking) and Workshops*. 2017, pages 1–9.

[48] Francesca Fossati, Sahar Hoteit, Stefano Moretti, and Stefano Secci. “Fair Resource Allocation in Systems With Complete Information Sharing”. In: *IEEE/ACM Transactions on Networking* 26.6 (2018), pages 2801–2814.

[49] Kun Zhu and Ekram Hossain. “Virtualization of 5G cellular networks as a hierarchical combinatorial auction”. In: *IEEE Transactions on Mobile Computing* 15.10 (2015), pages 2640–2654.

[50] Jianwei Huang, Randall A Berry, and Michael L Honig. “Auction-based spectrum sharing”. In: *Mobile Networks and Applications* 11.3 (2006), pages 405–418.

[51] Xia Zhou and Haitao Zheng. “TRUST: A general framework for truthful double spectrum auctions”. In: *IEEE INFOCOM 2009*. IEEE. 2009, pages 999–1007.

[52] Dermot Gately. “Sharing the gains from regional cooperation: A game theoretic application to planning investment in electric power”. In: *International Economic Review* (1974), pages 195–208.

[53] Stephen C Littlechild and KG Vaidya. “The propensity to disrupt and the disruption nucleolus of a characteristic function game”. In: *International Journal of Game Theory* 5.2-3 (1976), pages 151–161.

[54] Balázs Sziklai. “On the computation of the nucleolus of cooperative transferable utility games, P.h.D. Thesis”. In: *Eötös Loránd University, Budapest* (2015).

[55] Yvo de Jong Bultitude and Terhi Rautiainen. “IST-4-027756 WINNER II D1. 1.2 V1. 2 WINNER II Channel Models”. In: *EBITG, TUI, UOULU, CU/CRC, NOKIA, Technical Report* (2007).

[56] Francesca Fossati, Deep Medhi, Stefano Moretti, and Stefano Secci. “Error Estimate and Fairness in Resource Allocation with Inaccurate Information Sharing”. In: *IEEE Networking Letters* 1.4 (2019), pages 173–177.

[57] Roch A Guerin and Ariel Orda. “QoS routing in networks with inaccurate information: theory and algorithms”. In: *IEEE/ACM transactions on Networking* 7.3 (1999), pages 350–364.

[58] Francesca Fossati, Stefano Moretti, Patrice Perny, and Stefano Secci. “Multi-Resource Allocation for Network Slicing”. In: *submitted*.

[59] Francesca Fossati, Stefano Moretti, and Stefano Secci. “Multi-Resource Allocation for Network Slicing under Service Level Agreements”. In: *IEEE 10th International conference on the Network of the Future (NoF’ 19)*. 2019.

[60] 5G Americas. *Network Slicing for 5G and Beyond*. white paper, 2016.

[61] NGMN. *NGMN 5G white paper*. 2014.

[62] ITU-R. *Framework and overall objectives of the future development of IMT for 2020 and beyond, M.2083-0*. Sept. 2015.

[63] 3GPP TS 22.261 V15.5.0. *Service requirements for next generationnew services and markets*.

[64] Pablo Caballero, Albert Banchs, Gustavo De Veciana, and Xavier Costa-Pérez. “Multi-tenant radio access network slicing: Statistical multiplexing of spatial loads”. In: *IEEE/ACM Transactions on Networking* 25.5 (2017), pages 3044–3058.

[65] Mathieu Leconte, Georgios S Paschos, Panayotis Mertikopoulos, and Ulaş C Kozat. “A resource allocation framework for network slicing”. In: *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE. 2018, pages 2177–2185.

[66] Wanqing Guan, Xiangming Wen, Luhan Wang, Zhaoming Lu, and Yidi Shen. “A service-oriented deployment policy of end-to-end network slicing based on complex network theory”. In: *IEEE Access* 6 (2018), pages 19691–19701.

[67] Gang Wang, Gang Feng, Wei Tan, Shuang Qin, Ruihan Wen, and SanShan Sun. “Resource allocation for network slices in 5G with network resource pricing”. In: *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE. 2017, pages 1–6.

[68] Menglan Jiang, Massimo Condoluci, and Toktam Mahmoodi. “Network slicing in 5G: An auction-based model”. In: *2017 IEEE International Conference on Communications (ICC)*. IEEE. 2017, pages 1–6.

[69] Pablo Caballero, Albert Banchs, Gustavo de Veciana, and Xavier Costa-Pérez. “Network slicing games: Enabling customization in multi-tenant networks”. In: *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE. 2017, pages 1–9.

[70] Yong Xiao, Mohammed Hirzallah, and Marwan Krunz. “Distributed resource allocation for network slicing over licensed and unlicensed bands”. In: *IEEE Journal on Selected Areas in Communications* 36.10 (2018), pages 2260–2274.

[71] Hassan Halabian. “Distributed resource allocation optimization in 5G virtualized networks”. In: *IEEE Journal on Selected Areas in Communications* 37.3 (2019), pages 627–642.

[72] Seungik Lee, Sangheon Pack, Myung-Ki Shin, E Paik, and Rory Browne. “Resource management in service chaining”. In: *IETF Internet-Draft, draft-irtf-nfvrg-resource-management-service-chain-01* (2015).

[73] Yoav Etsion, Dan Tsafrir, and Dror G Feitelson. “Process prioritization using output production: scheduling for multimedia”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 2.4 (2006), pages 318–342.

[74] Intel. *Impact of the Intel Data Plane Development Kit (Intel DPDK) on packet throughput in virtualized network element*. 2013.

[75] NGMN Alliance. “Recommendations for NGMN KPIs and Requirements for 5G”. In: *techreport, June* (2016).

[76] Carlee Joe-Wong, Soumya Sen, Tian Lan, and Mung Chiang. “Multiresource allocation: Fairness–efficiency tradeoffs in a unifying framework”. In: *IEEE/ACM Transactions on Networking* 21.6 (2013), pages 1785–1798.

[77] Ronald R Yager. “On ordered weighted averaging aggregation operators in multicriteria decisionmaking”. In: *IEEE Transactions on systems, Man, and Cybernetics* 18.1 (1988), pages 183–190.

[78] Anthony F Shorrocks. “Ranking income distributions”. In: *Economica* 50.197 (1983), pages 3–17.

[79] John A Weymark. “Generalized Gini inequality indices”. In: *Mathematical Social Sciences* 1.4 (1981), pages 409–430.

[80] Hugh Dalton. “The measurement of the inequality of incomes”. In: *The Economic Journal* 30.119 (1920), pages 348–361.

[81] Włodzimierz Ogryczak and Tomasz Śliwiński. “On solving linear programs with the ordered weighted averaging objective”. In: *European Journal of Operational Research* 148.1 (2003), pages 80–91.

[82] Julien Lesca and Patrice Perny. “LP Solvable Models for Multiagent Fair Allocation Problems.” In: *ECAI*. 2010, pages 393–398.

[83] John F Nash Jr. “The bargaining problem”. In: *Econometrica: Journal of the Econometric Society* (1950), pages 155–162.

[84] Dimitris Bertsimas, Vivek F Farias, and Nikolaos Trichakis. “The price of fairness”. In: *Operations research* 59.1 (2011), pages 17–31.

[85] Corrado Gini. “Measurement of inequality of incomes”. In: *The Economic Journal* 31.121 (1921), pages 124–126.

[86] Dinesh C Verma. “Service level agreements on IP networks”. In: *Proceedings of the IEEE* 92.9 (2004), pages 1382–1388.

[87] Mohammad Asif Habibi, Bin Han, Meysam Nasimi, and Hans D Schotten. “The Structure of Service Level Agreement of Slice-based 5G Network”. In: *arXiv preprint arXiv:1806.10426* (2018).

[88] M Series. “IMT Vision–Framework and overall objectives of the future development of IMT for 2020 and beyond”. In: *Recommendation ITU* (2015), pages 2083–.

[89] Francesca Fossati, Stéphane Rovedakis, Stefano Moretti, and Stefano Secci. “Decentralization of 5G slice resource allocation”. In: *IEEE/IFIP Network Operations and Management Symposium (NOMS 2020)*.

[90] Petar Popovski, Kasper Fløe Trillingsgaard, Osvaldo Simeone, and Giuseppe Durisi. “5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view”. In: *IEEE Access* 6 (2018), pages 55765–55779.

[91] George Coulouris, Jean Dollimore, and Tim Kindberg. “Distributed Systems: Concepts and Designs. 3rd”. In: *Edition, AddisonWesley–Pearson Education* (2001).

[92] Roger B Myerson. “Optimal auction design”. In: *Mathematics of operations research* 6.1 (1981), pages 58–73.

[93] William Vickrey. “Counterspeculation, auctions, and competitive sealed tenders”. In: *The Journal of finance* 16.1 (1961), pages 8–37.

[94] Edward H Clarke. “Multipart pricing of public goods”. In: *Public choice* 11.1 (1971), pages 17–33.

[95] Theodore Groves. “Incentives in teams”. In: *Econometrica* 41.4 (1973), pages 617–631.

[96] Thibault Gajdos and John A Weymark. “Multidimensional generalized Gini indices”. In: *Economic Theory* 26.3 (2005), pages 471–496.