A Distributed Coalition Formation Framework for Fair User Cooperation in Wireless Networks

Walid Saad[1], Zhu Han[2], Mérouane Debbah[3], and Are Hjørungnes[1]

Abstract

Cooperation in wireless networks allows single antenna devices to improve their performance by forming virtual multiple antenna systems. However, performing a distributed and fair cooperation constitutes a major challenge. In this work, we model cooperation in wireless networks through a game theoretical algorithm derived from a novel concept from coalitional game theory. A simple and distributed merge-and-split algorithm is constructed to form coalition groups among single antenna devices and to allow them to maximize their utilities in terms of rate while accounting for the cost of cooperation in terms of power. The proposed algorithm enables the users to self-organize into independent disjoint coalitions and the resulting clustered network structure is characterized through novel stability notions. In addition, we prove the convergence of the algorithm and we investigate how the network structure changes when different fairness criteria are chosen for apportioning the coalition worth among its members. Simulation results show that the proposed algorithm can improve the individual user's payoff up to 40.42% as well as efficiently cope with the mobility of the distributed users.

Keywords: Game theory, coalitional games, cooperation, coalition formation

I. Introduction

The wireless network performance can be improved through cooperation techniques. Cooperation allows wireless network users to benefit from various gains such as an increase in the achieved rate or an improvement in the bit error rate. Designing an efficient cooperation algorithm faces numerous challenges. First and foremost, cooperation entails various costs, such as power, that can limit its benefits or even impair the users' performance. Second, wireless network users tend to be selfish in nature. Therefore, deriving a fair and practical cooperation algorithm where the decision to cooperate does not degrade the performance of any of the cooperating users is a tedious task. Moreover, if a cooperation algorithm depends on a centralized entity in the network such as a base station (BS), an extra amount of communication overhead is required for information exchange among the users. Such a centralized scheme will heavily depend on the availability of resources at the centralized entity. In consequence, there is a strong need to design a cooperation algorithm that can reduce this extra communication overhead by allowing the users to autonomously take the decision for cooperation without relying on a centralized intelligence. In summary, deriving a distributed and fair cooperative strategy is highly challenging but desirable in practice.

An important application for cooperation is the formation of virtual MIMO systems through cooperation among single antenna devices. In this context, a number of single-antenna devices can form virtual multiple antenna transmitters or receivers through cooperation, consequently, benefiting from the advantages of MIMO systems without the extra burden of having multiple antennas physically present on each transmitter or receiver. The information theoretical aspects of virtual MIMO systems were thoroughly exploited in [1], [2] and [3]. On one hand, the authors in [1] showed the gains in terms of outage capacity resulting from the cooperation of two single antenna devices that are transmitting to a far away receiver in a Rayleigh fading channel. On the other hand, the work in [2], [3] considered cooperation among multiple single antenna transmitters as well as receivers in a broadcast channel. Different cooperative scenarios were studied in the presence of a power cost for cooperation and the results showed the benefits of cooperation from a sum-rate perspective. Virtual MIMO gains are not only limited to rate gains. For example, forming virtual MIMO clusters in sensor networks can yield gains in terms of energy conservation [4].

Implementing distributed cooperation algorithms that allow the wireless network to reap these capacity or energy benefits requires an adequate analytical tool. In this regard, game theory provides a highly appealing mathematical tool for designing such distributed algorithms for cooperation or competition scenarios in wireless networks [5]. For instance, using coalitional game theory, the work in [6] and [7] focused on devising a distributed cooperative

algorithm for rate improvement through *receivers* cooperation. The authors showed that for the receiver coalition game in a Gaussian interference channel and synchronous CDMA multiple access channel (MAC), a stable grand coalition of all users can be formed if no cost for cooperation is accounted for. Subsequently, two schemes were provided for dividing the payoffs, in terms of rate, among the users: the Nash Bargaining game solution and the proportional fair payoff division. Furthermore, using game theory, this fair allocation of rate for cooperating users in an interference channel was also studied in [8] for the transmitters. The authors in [8] assumed that the users in a Gaussian multiaccess channel will bargain for favorable rate allocation by threatening to cooperate and form coalitions of devices that will jam the channel. Based on this jamming assumption, the authors showed that a unique rate allocation exists verifying certain well defined fairness axioms from coalitional games.

Moreover, distributed cooperation through game theory is not restricted to the virtual MIMO problem but it is also of interest at higher layers such as the network and transport layers. Cooperation in routing protocols was tackled in [9] and [10] to reduce energy cost. The system derived in these papers encourages cooperation by rewarding service providers according to their contribution. Another aspect of cooperative networks, resource allocation, is discussed in [11]. Finally, cooperation in packet forwarding was studied in [12] and [13] using cooperative game theory, repeated game theory, and machine learning.

In summary, previous work on cooperation focused mainly on the information theoretical analysis of the cooperation gains, and characterized these gains in the presence of no cost, namely in the virtual MIMO problem [1–3], [6–8]. The main objectives of this paper are two fold: (i) investigate the limitation on the cooperation gains in a virtual MIMO system, in the presence of a cost, and most importantly (ii) provide a distributed algorithm that enables the users in a wireless networks to form these virtual MIMO systems and benefit from their widely established gains (in the presence of a cost). In fact, while existing literature answered the question of "why to cooperate?", we aim to answer questions such as when to cooperate and with whom to cooperate?, notably when cooperation incurs a cost as well as a benefit. In this context, the main contributions of this paper are: (1) to design a distributed game theoretical framework that enables single antenna *transmitters* to *autonomously* take decisions to cooperate and form virtual MIMO coalitions while accounting for the inherent benefit-cost trade off involved in this formation; (2) to provide fair rules for performance improvement through cooperation, and; (3) to characterize the coalition structure that will potentially form when these users take the decisions to cooperate. For this purpose, we construct a coalition formation algorithm based on well-defined and distributed merge-and-split rules from cooperative games suitable for tackling this transmitter cooperation problem. The convergence of this merge-and-split transmitter cooperation algorithm is

shown and the stability of the resulting coalition structure is characterized through suitable game theoretical stability notions. Finally, various fairness criteria for allocating the extra benefits among coalition users are discussed and their effect on the network structure is analyzed. Simulation results show that our algorithm can improve the individual user's payoff up to 40.42% as well as efficiently handle environmental changes such as the users' mobility.

The rest of this paper is organized as follows: Section II presents the transmitter cooperation system model. Section III presents the proposed game theoretic algorithm while Section IV discusses fairness criteria for payoff division. Simulation results are analyzed in Section V and conclusions are drawn in Section VI.

II. SYSTEM MODEL

In this section, we present the transmitter cooperation coalitional game model and discuss its properties.

A. Transmitter Cooperation Model

Consider a network having M_t single antenna transmitters (e.g. mobile users) sending data in the uplink to a fixed receiver, e.g., a BS, with M_r receive antennas (multiple access channel). Denote $N = \{1 \ldots M_t\}$ as the set of all M_t users in the network, and let $S \subseteq N$ be a coalition consisting of $|S|$ users ($|\cdot|$ represents the cardinality of a set). We consider a TDMA transmission in the network thus, in a non-cooperative manner, the M_t users require a time scale of M_t slots to transmit as every user occupies one time slot. When cooperating, the single antenna transmitters form different disjoint coalitions (each coalition can be seen as a single user MIMO device) and they will subsequently transmit in a TDMA manner, which is one coalition per transmission. During the time scale M_t, each coalition is able to transmit within all the time slots *previously held by its users*. For a cooperating coalition S, we consider a block fading $M_r \times |S|$ channel matrix $\boldsymbol{H}_S$ with a path loss model between the users in S and the BS with each element of the matrix $h_{i,k} = e^{j\phi_{i,k}} \sqrt{\kappa/d_{i,k}^\alpha}$ with α the path loss exponent, κ the path loss constant, $\phi_{i,k}$ the phase of the signal from transmitter i to the BS receiver k, and $d_{i,k}$ the distance between transmitter i and the base BS's receiver k. An illustration of the model is shown in Fig. 1 for $M_t = 6$.

As we are considering a TDMA system, we define a fixed transmit power constraint *per time slot*, i.e., a power constraint *per coalition* $\tilde{P}$ as in [2], [3]. This average power constraint is applied to all the transmitters that are part of the coalition active in the slot. In the non-cooperative scenario, this same power constraint per slot is simply the power constraint per individual user active in the slot. In fact, due to ergodicity, for each time slot, the average long term power constraint per individual user and the power constraint per slot (i.e. constraining all transmitters of a coalition active in a slot) are the same [2], [3]. In the considered TDMA system, each coalition transmits in a slot, hence, perceiving no interference from other coalitions during transmission. As a result, in a slot, the sum-rate capacity of

the virtual MIMO formed by a coalition S, under a power constraint P_S with Gaussian signaling is given by [14]

$$C_S = \max_{\mathbf{Q}_S} I(\boldsymbol{x}_S; \boldsymbol{y}_S) = \max_{\mathbf{Q}_S} \log \det(\boldsymbol{I}_{M_r} + \boldsymbol{H}_S \cdot \boldsymbol{Q}_S \cdot \boldsymbol{H}_S^{\dagger}) \text{ s.t. } \text{tr}[\boldsymbol{Q}_S] \leq P_S, \tag{1}$$

where $\boldsymbol{x}_S$ and $\boldsymbol{y}_S$ are, respectively, the transmitted and received signal vectors of coalition S of size $|S| \times 1$ and $M_r \times 1$, $\mathbf{Q}_S = \mathrm{E}\left[\boldsymbol{x}_S \cdot \boldsymbol{x}_S^{\dagger}\right]$ is the covariance of $\boldsymbol{x}_S$ and $\boldsymbol{H}_S$ is the channel matrix with $\boldsymbol{H}_S^{\dagger}$ its conjugate transpose.

The considered channel matrix $\boldsymbol{H}_S$ is assumed perfectly known at the transmitter and receiver, thus, the maximizing input signal covariance is given by $\boldsymbol{Q}_S = \boldsymbol{V}_S \boldsymbol{D}_S \boldsymbol{V}_S^{\dagger}$ ([14], [15]) where $\boldsymbol{V}_S$ is the unitary matrix given by the singular value decomposition of $\boldsymbol{H}_S = \boldsymbol{U}_S \boldsymbol{\Sigma}_S \boldsymbol{V}_S^{\dagger}$ and $\boldsymbol{D}_S$ is an $|S| \times |S|$ diagonal matrix given by $\boldsymbol{D}_S = \mathrm{diag}(D_1, \ldots, D_K, 0, \ldots, 0)$ where $K \leq \min(M_r, |S|)$ is the number of positive singular values of the channel $\boldsymbol{H}_S$ (eigenmodes) and each D_i given by $D_i = (\mu - \lambda_i^{-1})^+$, where μ is determined by water-filling to satisfy the coalition power constraint $\text{tr}[\boldsymbol{Q}_S] = \text{tr}[\boldsymbol{D}_S] = \sum_i D_i = P_S$, and λ_i is the ith eigenvalue of $\boldsymbol{H}_S^{\dagger} \boldsymbol{H}_S$. Using [14], the resulting capacity, in a slot, for a coalition S is

$$C_S = \sum_{i=1}^{K} (\log(\mu \lambda_i))^+. \tag{2}$$

For forming the considered virtual MIMO coalitions and benefiting from the capacity gains, the users need to exchange their data information and their channel (user-BS) information. For this purpose, we will consider a cost for information exchange in terms of transmit power. This transmit power cost mainly models the data exchange penalty. As we consider block fading channels with a long coherence time, the additional power penalty for exchanging the user-BS channel information can be deemed as negligible relatively to the data exchange cost, since the considered channel varies slowly (for example, exchange of the channel information can be done only periodically). Consequently, the cost for information exchange is taken as the sum of the powers required by each user in a coalition S to broadcast to its corresponding farthest user inside S. Due to the broadcast nature of the wireless channel, once a coalition member broadcasts its information to the farthest user, all the other members can also receive this information simultaneously. The power needed for broadcast between a user $i \in S$ and its corresponding *farthest* user $\hat{i} \in S$ is

$$\bar{P}_{i,\hat{i}} = \frac{\nu_0 \cdot \sigma^2}{g_{i,\hat{i}}^2}, \tag{3}$$

where ν_0 is a target average SNR for information exchange, σ^2 is the noise variance and $g_{i,\hat{i}} = \sqrt{\kappa/d_{i,\hat{i}}^{\alpha}}$ is the path loss between users i and $\hat{i}$ with $d_{i,\hat{i}}$ the distance between users i and $\hat{i}$. In consequence, the total power cost for a coalition S having $|S|$ users is given by $\hat{P}_S$ as follows

$$\hat{P}_S = \sum_{i=1}^{|S|} \bar{P}_{i,\hat{i}}. \tag{4}$$

It is interesting to note that the defined cost depends on the location of the users and the size of the coalition; hence, a higher power cost is incurred whenever the distance between the users increases or the coalition size increases. Thus, the actual power constraint P_S per coalition S *with cost* is given by

$$P_S = (\tilde{P} - \hat{P}_S)^+, \tag{5}$$

where $\tilde{P}$ is the average power constraint per coalition (per slot), $\hat{P}_S$ the cooperation power cost given in (4) and $a^+ \triangleq \max(a, 0)$. In order to achieve the capacity in (2), within a slot, each user of a coalition S adjusts its power value based on the water-filling solution, taking into account the available power constraint P_S. Note that, since the power constraint $\tilde{P}$ applied over a coalition is the same as the maximum power constraint per individual user in the coalition, the water-filling solution always yields a power value per user that does not violate the user's available power after deducing the cost for cooperation in (3) from its individual long term power constraint.

The considered power cost does not take into account the interference for exchange of information between users and can be considered as a lower bound of the penalty incurred by cooperation. In addition to this power cost, a fraction of time may be required for the data exchange between the users prior to cooperation. However, due to the fact that the proposed power cost given in (4) depends on distance and coalition size, the formed coalitions will typically consist of small clusters of nearby close users (as will be verified through simulations), and thus the users can exchange information at high rates rendering the time for data exchange negligible relatively to the transmission time slot (typically, the distance between the users of a coalition and the BS is larger than the distance between the coalition users themselves). Furthermore, in practice, performing a cooperation for virtual MIMO formation can require a synchronization at the carrier frequency between the nodes, yielding some costs for practical implementation. In this work, we will not account for these carrier synchronization costs similar to existing virtual MIMO literature [1], [3], [4], [6], [7]. Finally, the coalition formation results derived in this paper could also be applied for other cost functions without loss of generality. For example, the cost of power can be replaced by a cost of bandwidth where one could quantify the use of an additional band for signaling and information exchange, orthogonal to the band of transmission.

Based on the defined capacity benefit and power cost, over the TDMA time scale of M_t, for every coalition $S \subseteq N$, we define the utility function (or *characteristic function* in coalitional game theory terms [16]) as

$$v(S) = \begin{cases} |S| \cdot C_S, & \text{if } P_S > 0; \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where P_S is given by (5), C_S is given by (2) and $|S|$ is the number of users in S. This utility represents the total capacity achieved by coalition S during the time scale M_t while accounting for the cost through the power constraint. A coalition of $|S|$ users will transmit with capacity C_S during $|S|$ time slots; thus achieving a total capacity of $v(S)$ during the time scale M_t (e.g. in Fig. 1 during $M_t = 6$ coalition 2 consisting of 2 users achieves C_2 in slot 4 and C_2 in slot 6; thus a total of $2 \cdot C_2$ during $M_t = 6$ slots). The second case in (6) implies that if the power for information exchange is larger than (or equal to) the available power constraint the coalition cannot be formed due to a utility of 0. The payoff of each user in a coalition S is computed by a fair division of the utility $v(S)$, through various criteria explained in Section IV. This individual user payoff denoted ϕ_i^v represents the total rate achieved by user i during the transmission time scale M_t. Thus, we have a coalitional game (N,v) with a transferable utility (i.e. the coalition value can be arbitrarily divided among its users) and we seek a coalition structure that allows the users to maximize their utilities with cost.

B. Transmitter Cooperation Game Properties

For modeling cooperation in wireless networks, existing work mainly sought to prove that the grand coalition of all users can form. However, in the proposed transmitter cooperation (N,v) coalitional game we will prove that, due to the cooperation costs, the grand coalition will seldom form through the following definitions and theorems.

Definition 1: A coalitional game (N, v) with transferable utility is said to be superadditive if for any two disjoint coalitions $S_1, S_2 \subset N$, $v(S_1 \cup S_2) \geq v(S_1) + v(S_2)$.

Theorem 1: The proposed transmitter (N,v) coalitional game with cost is non-superadditive.

Proof: Consider two disjoint coalitions $S_1 \subset N$ and $S_2 \subset N$ in the network with their corresponding utilities $v(S_1)$ and $v(S_2)$ when they do not cooperate with each other. Assume that the users of $S_1 \cup S_2$ are located far enough to yield a power cost per (4) $\hat{P}_{S_1 \cup S_1} \geq \tilde{P}$. In this case, by (5) $P_{S_1 \cup S_2} = 0$ yielding $v(S_1 \cup S_2) = 0 < v(S_1) + v(S_2)$ (6); hence the game is not superadditive. ∎

Definition 2: A payoff vector $\boldsymbol{\phi}^v = (\phi_1^v, \ldots, \phi_{M_t}^v)$ for dividing the value v of a coalition is said to be *group rational* or efficient if $\sum_{i=1}^{M_t} \phi_i^v = v(N)$. A payoff vector $\boldsymbol{\phi}^v$ is said to be *individually rational* if the player can obtain the benefit no less than acting alone, i.e. $\phi_i^v \geq v(\{i\}), \forall\, i$. An *imputation* is a payoff vector satisfying the above two conditions.

Definition 3: An imputation $\boldsymbol{\phi}^v$ is said to be unstable through a coalition S if $v(S) > \sum_{i \in S} \phi_i^v$, i.e., the players

have incentive to form coalition S and reject the proposed ϕ^v. The set $\mathcal{C}$ of stable imputations is called the *core*, i.e.,

$$\mathcal{C} = \left\{ \phi^v : \sum_{i \in N} \phi_i^v = v(N) \text{ and } \sum_{i \in S} \phi_i^v \geq v(S) \ \forall \ S \subseteq N \right\}. \tag{7}$$

A non-empty core means that the players have an incentive to form the grand coalition.

Remark 1: In general, the core of the (N,v) transmitter cooperation game with cost is empty.

In the proposed model, the costs for cooperation for a coalition S increase as the number of users in a coalition increase as well as when the distance between the users increase. In particular, consider the grand coalition $\{N\}$ of all M_t users in the network. This coalition consists of a large number of users who are randomly located at different distances. Hence, the grand coalition N will often have a value of $v(N) = 0$ due to the cooperation costs and several coalitions $S \subset N$ have an incentive to deviate from this grand coalition and form independent disjoint coalitions. As a result, no imputation can be found to lie in the core, and the core of the the transmitter cooperation (N,v) game is generally empty. As a result of the non-superadditivity of the game as well as the emptiness of the core, the grand coalition of all transmitters will not form. Instead, independent disjoint coalitions will form. Hence, in the next section, we will devise a novel algorithm for coalition formation that can characterize these disjoint coalitions.

III. COALITION FORMATION ALGORITHM

In this section, we construct a novel coalition formation algorithm and we prove its key properties.

A. Coalition Formation Concepts

Coalition formation has been a topic of high interest in game theory [17–21]. In [19], [20] and [21], an interesting approach for dynamic coalition formation is derived. The mathematical tools presented in [20] and [21] allow to build algorithms to dynamically form coalitions among players through two simple merge-and-split rules, which can be applied in a distributed manner and, thus, deemed suitable for wireless network games. Introducing this framework and applying it into wireless networks requires several concepts to be defined as follows.

Definition 4: A *collection* of coalitions in the grand coalition N, denoted S, is defined as the set $S = \{S_1, \ldots, S_l\}$ of mutually disjoint coalitions S_i of N. In other words, a collection is any arbitrary group of disjoint coalitions S_i of N not necessarily spanning all players of N. If the collection spans all the players of N; that is $\bigcup_{j=1}^{l} S_j = N$, the collection is referred to as a *partition* of N.

Definition 5: A preference operator or *comparison relation* $\rhd$ is defined for comparing two collections $R = \{R_1, \ldots, R_l\}$ and $S = \{S_1, \ldots, S_p\}$ that are partitions of the same subset $A \subseteq N$ (i.e. same players in R and S). Thus, $R \rhd S$ implies that the way R partitions A is preferred to the way S partitions A.

Various well known orders can be used as comparison relations [20], [21]. These orders are split into two categories: coalition value orders and individual value orders. Coalition value orders compare two collections (or partitions) using the value of the coalitions inside these collections such as in the utilitarian order where $R \rhd S$ implies $\sum_{i=1}^{l} v(R_i) > \sum_{i=1}^{p} v(S_i)$. Individual value orders perform the comparison using the individual payoffs such as the Pareto order. For these orders, two collections R and S are seen as sets of individual payoffs of the same length L (number of players) resulting from a group rational division of the utilities of each coalition $R_i \in R$ and $S_i \in S$. For a collection $R = \{R_1, \ldots, R_l\}$, the payoff of a player j in a coalition $R_i \in R$ is denoted by $\phi_j^v(R)$; and $\sum_{i=1}^{l} v(R_i) = \sum_{j=1}^{L} \phi_j^v(R)$. The Pareto order is defined as

$$R \rhd S \iff \{\phi_j^v(R) \geq \phi_j^v(S) \ \forall \ j \in R, S\} \text{ with } at\ least\ one\ strict\ inequality\ (>) \text{ for a player } k. \tag{8}$$

The Pareto order implies that a collection R is preferred over S, if at least one player is able to improve its payoff when the coalition structure changes from S to R without decreasing other players' payoffs.

B. Merge-and-Split Coalition Formation Algorithm

Using the coalition formation concepts prescribed in the previous section, a coalition formation algorithm for self organization in wireless networks can be generated. This algorithm will be based on simple rules of merge-and-split that allow to modify a partition T of N as follows [20]:

- **Merge Rule:** Merge any set of coalitions $\{S_1, \ldots, S_l\}$ where $\{\bigcup_{j=1}^{l} S_j\} \rhd \{S_1, \ldots, S_l\}$, therefore, $\{S_1, \ldots, S_l\} \rightarrow \{\bigcup_{j=1}^{l} S_j\}$. (each S_i denotes a coalition).

- **Split Rule:** Split any coalition $\bigcup_{j=1}^{l} S_j$ where $\{S_1, \ldots, S_l\} \rhd \{\bigcup_{j=1}^{l} S_j\}$, thus, $\{\bigcup_{j=1}^{l} S_j\} \rightarrow \{S_1, \ldots, S_l\}$. (each S_i denotes a coalition).

In brief, multiple coalitions will merge (split) if merging (splitting) yields a preferred collection based on a chosen $\rhd$. In [20] and [21] it is shown that any arbitrary iteration of merge-and-split operations *terminates*, therefore, it will be suitable to devise a coalition formation algorithm by means of merge-and-split. In the transmitter cooperation game, the Pareto order given by (8) is highly appealing as a comparison relation $\rhd$ for the merge-and-split rules. With the Pareto order, coalitions will merge only if at least one user can enhance its individual payoff through this merge without decreasing the other users' payoffs. Similarly, a coalition will split only if at least one user in that coalition is able to strictly improve its individual payoff through the split without hurting other users. A decision to merge or split is, thus, tied to the fact that all users must benefit from merge or split, thus, any merged (or split) form is reached only if it allows all involved users to maintain their payoffs with at least one user improving. In summary, the

proposed algorithm is a coalition formation algorithm with partially reversible agreements [18], where the users sign a binding agreement to form a coalition through the merge operation (if *all* users are able to improve their individual payoffs from the previous state) and they can only split this coalition if splitting does not decrease the payoff of any coalition member (partial reversibility). Having partial reversibility through the split operation reduces the complexity of the coalition formation process relatively to a fully reversible process [18] but can impact the coalition stability as further discussed in Section III-C.

For the proposed virtual MIMO formation game, the self-organizing algorithm consists of two phases: adaptive coalition formation and transmission. In the adaptive coalition formation phase, an iteration of sequential merge-and-split rules is performed until the iteration terminates yielding a final network partition composed of independent disjoint coalitions. In the transmission phase, the formed coalitions transmit in their corresponding slots in a TDMA manner. The transmission phase may occur several times prior to the repetition of the coalition formation phase, notably in low mobility environment where changes in the coalition structure due to mobility are seldom.

Although any arbitrary merge can be performed, we propose a distributed cost-based merge process allowing the coalitions (users) to perform a local search for partners. Consequently, the decision to merge with neighboring coalitions is taken based on the Pareto order proceeding from the partner that provides the lowest cost. In order for coalition S_1 to merge with another coalition S_2, the utility of the formed coalition through merge must be positive; that is $v(S_1 \cup S_2) > 0$ otherwise no benefits exist for the merge. Thus, based on the defined power cost (4) and utility (6); coalitions can only merge when the cost for cooperation is less than $\tilde{P}$. Otherwise, when the cost is greater than or equal $\tilde{P}$, through (6) the utility of the merged coalition will be 0 and there is no mutual benefit. Thus, using (4) the merge is possible (non zero utility) for S_1 with S_2 if $\hat{P}_{S_1 \cup S_2} < \tilde{P}$, that is $\sum_{i=1}^{|S_1 \cup S_2|} \bar{P}_{i,\hat{i}} < \tilde{P}$ which, by (3), yields

$$\sum_{i=1}^{|S_1 \cup S_2|} \frac{1}{d_{i,\hat{i}}^{\alpha}} < \frac{\tilde{P}}{\nu_0 \cdot \sigma^2 \cdot \kappa}. \tag{9}$$

Thus, a coalition will only attempt to merge with other coalitions where (9) can be verified.

Each stage of the proposed algorithm starts from an initial network partition $T = \{T_1, \ldots, T_l\}$ of N. In this partition, any random coalition (user) can start with the merge process. For implementation purposes, assume that the coalition $T_i \in T$ which has the highest utility in the initial partition T starts the merge by attempting to cooperate with the coalition yielding the lowest cost. On one hand, if merging occurs, a new coalition $\tilde{T}_i$ is formed and, in its turn, coalition $\tilde{T}_i$ will attempt to merge with the lowest cost partner. On the other hand, if T_i was unable to merge with the smallest cost coalition, it tries the next lowest cost partner, proceeding sequentially through the coalitions

verifying (9). The search ends by a final merged coalition T_i^{final} composed of T_i and one or several of coalitions in its vicinity (or just T_i, if no merge occured). The algorithm is repeated for the remaining $T_i \in T$ until all the coalitions have made their local merge decisions, resulting in a final partition W. The coalitions in the resulting partition W are next subject to split operations, if any is possible. An iteration consisting of multiple successive merge-and-split operations is repeated until it terminates. Table I shows a summary of one stage of the proposed algorithm.

C. Stability Notions

The result of the proposed algorithm in Table I is a network partition composed of disjoint independent coalitions. The stability of this resulting network structure can be investigated with respect to a novel concept of defection function $\mathbb{D}$ [19], [20].

Definition 6: A *defection* function $\mathbb{D}$ is a function which associates with each partition T of N a family (group) of collections in N. A partition $T = \{T_1, \ldots, T_l\}$ of N is $\mathbb{D}$-*stable* if no group of players is interested in leaving T when the players who wish to leave can only form the collections allowed by $\mathbb{D}$.

Two important defection functions can be pinpointed [19], [20]. First, the $\mathbb{D}_{hp}(T)$ function (denoted $\mathbb{D}_{hp}$) which associates with each partition T of N the family of all partitions of N that can form by merging or splitting coalitions in T. This function allows any group of players to leave the partition T of N through merge-and-split operations to create another *partition* in N. Second, the $\mathbb{D}_c(T)$ function (denoted $\mathbb{D}_c$) which associates with each partition T of N the family of all collections in N. This function allows any group of players to leave the partition T of N through *any* operation and create an arbitrary *collection* in N. Two forms of stability stem from these definitions: $\mathbb{D}_{hp}$ stability and a stronger $\mathbb{D}_c$ stability. A partition T is $\mathbb{D}_{hp}$-stable, if no players in T are interested in leaving T through merge-and-split to form other partitions in N; while a partition T is $\mathbb{D}_c$-stable, if no players in T are interested in leaving T to form other collections in N.

The $\mathbb{D}$-stability of a partition depends on various properties of its coalitions. A partition $T = \{T_1, \ldots, T_l\}$ is $\mathbb{D}_{hp}$-stable, if the following two conditions are met ([19], [20]): (i)- For each $i \in \{1, \ldots, p\}$ and for each partition $\{R_1, \ldots, R_p\}$ of T_i we have $\{R_1, \ldots, R_p\} \not\rhd T_i$, and (ii)- For each $S \subseteq \{1, \ldots, l\}$ we have $\bigcup_{i \in S} T_i \not\rhd \{T_i | i \in S\}$ ($\not\rhd$ is the opposite rule of the preference operator $\rhd$). By definition of $\mathbb{D}_{hp}$ stability, we have

Theorem 2: Every partition resulting from our proposed merge-and-split algorithm is $\mathbb{D}_{hp}$-stable.

Proof: As every iteration of merge-and-split terminates, a resulting partition from such iterations cannot be subject to any further merge or split. Therefore, the players in a partition T resulting from sequential merge-and-split operations such as in the algorithm of Table I cannot leave this partition through merge or split. Assume $T = \{T_1, \ldots, T_l\}$ is

the partition resulting from the proposed merge-and-split algorithm. If for any $i \in \{1, \ldots, l\}$ and for any partition $\{S_1, \ldots, S_p\}$ of T_i we assume that $\{S_1, \ldots, S_p\} \rhd T_i$ then the partition T can still be modified through the application of the split rule on T_i contradicting with the fact that T resulted from a termination of the merge-and-split iteration; therefore $\{S_1, \ldots, S_p\} \not\rhd T_i$ (first $\mathbb{D}_{hp}$ stability condition verified). A similar reasoning is applicable in order to prove that T verifies the second condition; since otherwise a merge rule would still be applicable. ∎

One drawback of $\mathbb{D}_{hp}$-stable partitions is that coalitions within such partitions may be prone to deviations due to the partial reversibility of the merge-and-split algorithm. For instance, in a $\mathbb{D}_{hp}$-stable partition, once a group of users form a coalition S by Pareto order merge, some subset of S may be able to deviate from this coalition but is not allowed to do so unless the deviation does not decrease the payoff of the remaining users in S (split only by Pareto order). The rationale behind this is that, once an agreement is signed to form a coalition by Pareto order, the users can only deviate if they do not hurt the other coalition members. Coalitions exhibiting internal deviations are referred to as coalitions "prone to deviations". For the proposed algorithm, the number of such coalitions will generally be very small due to the usage of Pareto order for merge and the presence of cooperation costs which limit the possible deviations that are not captured by the split operation (generally, nearby users merge into a coalition and splitting occurs when they distance themselves due to mobility). Moreover, by imposing stringent fairness criteria for the payoff division as will be seen in Section V, the number of such coalitions can be further reduced.

Due to the possibility of having coalitions prone to deviations in a $\mathbb{D}_{hp}$-stable partition, a stronger $\mathbb{D}_c$-stable partition can be sought by the proposed algorithm. For instance, the work in [20] showed that, if it exists, a $\mathbb{D}_c$-stable partition has the following properties:

1) If it exists, a $\mathbb{D}_c$-stable partition is the *unique* outcome of any *arbitrary* iteration of merge-and-split and is a unique $\mathbb{D}_{hp}$-stable partition.

2) A $\mathbb{D}_c$-stable partition T is a unique $\rhd$-maximal partition, that is for all partitions $T' \neq T$ of N, $T \rhd T'$. In the case where $\rhd$ represents the Pareto order, this implies that the $\mathbb{D}_c$-stable partition T is the partition that presents a *Pareto optimal* payoff distribution for all the players.

3) A $\mathbb{D}_c$-stable partition does not contain any coalitions prone to deviations.

Clearly, a $\mathbb{D}_c$-stable partition is an optimal partition that the wireless network can seek as it provides a payoff distribution that is Pareto optimal with for all users with respect to any other network partition. In addition, this partition is a unique outcome of any arbitrary iteration of merge-and-split rules. However, the existence of a $\mathbb{D}_c$-stable partition is not always guaranteed [20]. The $\mathbb{D}_c$-stable partition $T = \{T_1, \ldots, T_l\}$ of the whole space N exists if a

partition of N verifies two necessary and sufficient conditions [20]:

A) For each $i \in \{1, \ldots, l\}$ and each pair of disjoint *coalitions* A and B such that $\{A \cup B\} \subseteq T_i$ we have $\{A \cup B\} \rhd \{A, B\}$ (referred to as cond. A) hereafter).

B) For the partition $T = \{T_1, \ldots, T_l\}$ a coalition $G \subset N$ formed of players belonging to different $T_i \in T$ is T-incompatible if for no $i \in \{1, \ldots, l\}$ we have $G \subset T_i$. $\mathbb{D}_c$-stability requires that for all T-incompatible coalitions $\{G\}[T] \rhd \{G\}$ where $\{G\}[T] = \{G \cap T_i \,\forall\, i \in \{1, \ldots, l\}\}$ is the projection of coalition G in partition T (referred to as cond. B) hereafter).

If no partition of N can satisfy these conditions, then no $\mathbb{D}_c$-stable partitions of N exists. Since the $\mathbb{D}_c$-stable partition is a unique outcome of any arbitrary merge-and-split iteration, we have

Lemma 1: For the proposed (N, v) transmitter cooperation coalitional game, the merge-and-split algorithm of Table I converges to the optimal $\mathbb{D}_c$-stable partition, if such a partition exists. Otherwise, the proposed algorithm yields a final network partition that is $\mathbb{D}_{hp}$-stable.

In the transmitter cooperation game, the existence of a $\mathbb{D}_c$-stable partition depends on various factors. For instance, cond. A) states that, for a $\mathbb{D}_c$-stable partition T, every coalition $T_i \in T$ must verify the Pareto order not only at the level of the whole coalition T_i but also at the level of all disjoint coalitions subsets of T_i. Verifying the Pareto order requires that the utility of every union of any two disjoint coalitions subsets of a coalition T_i must yield an extra utility over the disjoint case; that is $v(A \cup B) > v(A) + v(B) \,\forall\, A, B \subset T_i$. In an ideal case with no cost, as the number of transmit antennas is increased for a fixed power constraint, the overall system's diversity increases as the data passes through different channel values allowing, with adequate coding, the symbols to be recovered without error at a higher rate [15]. In such a case, since $A \cup B$ has a larger number of antennas than A and B, $\forall\, A, B \subset T_i$ and for each T_i we have

$$C_{A \cup B} > \max\left(C_A, C_B\right) \Leftrightarrow |A \cup B| \cdot C_{A \cup B} > |A| \cdot \max\left(C_A, C_B\right) + |B| \cdot \max\left(C_A, C_B\right), \tag{10}$$

$$|A \cup B| \cdot C_{A \cup B} > |A| \cdot C_A + |B| \cdot C_B \Leftrightarrow v(A \cup B) > v(A) + v(B), \tag{11}$$

which is sufficient to verify cond. A) for $\mathbb{D}_c$-stability when adequate payoff divisions are done. However, due to the cost $C_{A \cup B}, C_A$ and C_B can have different power constraints and (10)-(11) may not be guaranteed $\forall A, B \subset T_i$. Guaranteeing this condition is directly dependent on the cooperation cost within the coalitions in T and, thus, on the users' location. In practical networks, verifying cond. A) for $\mathbb{D}_c$-stability depends on the location of the users which are random.

Cond. B) for the existence of a $\mathbb{D}_c$-stable partition T is that players formed from different $T_i \in T$ have no incentive to form a coalition G outside of T. In the transmitter cooperation game, cond. B is also dependent on the location of the coalitions $T_i \in T$; specifically on the distance between the users in different coalitions $T_i \in T$. Thus, cond. B) is verified whenever two users belonging to different coalitions in a partition T are separated by large distances. A sufficient condition for verifying this second requirement can be derived.

Theorem 3: For a network partition $T = \{T_1, \ldots, T_l\}$ resulting from the proposed algorithm; if the distance $d_{i,j}$ between *any* two users $i \in T_i$ and $j \in T_j$ with $T_i \neq T_j$ verifies $d_{i,j} > \left(\frac{\kappa \cdot \tilde{P}}{2 \cdot \nu_0 \cdot \sigma^2} \right)^{\frac{1}{\alpha}} = \hat{d}_0$ then the second condition for $\mathbb{D}_c$ stability, cond. B), is verified.

Proof: Since a $\mathbb{D}_c$-stable partition is a unique outcome of any merge-and-split iteration, we will consider the partition $T = \{T_1, \ldots, T_l\}$ resulting from any merge-and-split iteration in order to show when cond. B) can be satisfied. A T-incompatible coalition is a coalition formed from users belonging to different $T_i \in T$. Consider the T-incompatible coalition $\{i, j\}$ that can potentially form between two users $i \in T_i$ and $j \in T_j$ with $T_i \neq T_j$. The total power cost for $\{i, j\}$ is given by (3) as $\hat{P}_{\{i,j\}} = \bar{P}_{i,j} + \bar{P}_{j,i} = 2 \cdot \bar{P}_{i,j}$. In the case where the total power cost is larger than the constraint, we have $\hat{P}_{\{i,j\}} \geq \tilde{P}$ and thus $\bar{P}_{i,j} \geq \frac{\tilde{P}}{2}$ which yields the required condition on distance $d_{i,j} \geq \left(\frac{\kappa \cdot \tilde{P}}{2 \cdot \nu_0 \cdot \sigma^2} \right)^{\frac{1}{\alpha}} = \hat{d}_0$. We will have by (6) $v(\{i, j\}) = 0$, and, thus, $\phi_i^v(\{i, j\}) = \phi_j^v(\{i, j\}) = 0$. Or we have that, $\{i, j\}[T] = \{\{i, j\} \cap T_k \ \forall \ k \in \{1, \ldots, l\}\} = \{\{i\}, \{j\}\}$, and, thus, $\phi_i^v(\{i, j\}[T]) = v(\{i\}) > \phi_i^v(\{i, j\}) = 0$ and $\phi_j^v(\{i, j\}[T]) = v(\{j\}) > \phi_j^v(\{i, j\}) = 0$. Consequently, $\{i, j\}[T] \rhd \{i, j\}$ and cond. B) is verified for any T-incompatible coalition formed of 2 users. Moreover, when *any* two users $i \in T_i$ and $j \in T_j$ with $T_i \neq T_j$ are separated by $\hat{d}_0$, T-incompatible coalitions G with $|G| > 2$ will have a cost $\hat{P}_G > \hat{P}_{\{i,j\}} \geq \tilde{P}$ and thus $v(G) = 0$; yielding $G[T] \rhd G$ for all T-incompatible coalitions G. Hence, when any two users in the network partition T resulting from merge-and-split are separated by a distance larger than $\hat{d}_0$, then cond. B) for $\mathbb{D}_c$ stability existence is verified. ∎

In summary, the existence of the $\mathbb{D}_c$-stable partition is closely tied to the users' location; which is a random parameter in practical networks. A partition resulting from our algorithm will be either $\mathbb{D}_c$ or $\mathbb{D}_{hp}$-stable as per Lemma 1.

D. Distributed Implementation of Merge-and-Split

The proposed algorithm in Table I can be implemented in a distributed way. As the user can detect the strength of other users' uplink signals (through techniques similar to those used in the ad hoc routing discovery [22]), nearby coalitions can be discovered and the local merge algorithm performed. Each coalition surveys neighboring coalitions satisfying (9) and attempts to merge based on the Pareto order. The users in a coalition need only to know the maximum distances with respect to the users in neighboring coalitions. Moreover, each formed coalition internally

decides to split if its members find a split form by Pareto order. By using a control channel, the distributed users can exchange some channel information and then and then cooperate using our model (exchange data information if needed, form coalition then transmit). Signaling for this handshaking can be minimal.

The complexity of the algorithm in Table I lies in the complexity of the merge-and-split operations. For a given network structure, one run of the cost-based *merge* process detailed in Section III-B implies that each coalition will try to merge with other coalitions where (9) is verified. The most complex case for the merge occurs when the network partition consists of M_t non-cooperative users that are located closely to verify (9) but not close enough to merge. In such a scenario, every user attempts to merge with all the others; but the merge is unsuccessful due to high cost. The first user requires $M_t - 1$ attempts for merge, the second requires $M_t - 2$ attempts and so on. The total number of merge attempts will be $\sum_{i=1}^{M_t-1} i = \frac{M_t(M_t-1)}{2}$. In practice, the merge process requires a significantly *lower* number of attempts. For instance, after the first run of the algorithm, the initial M_t non-cooperative users will self-organize into larger coalitions. Subsequent runs of the algorithms will deal with a network composed of a number of coalitions that is much smaller than M_t; reducing the number of merge attempts per coalition. This complexity is further *reduced* by the fact that, due to the cost, a coalition does not need to attempt to merge with far away or large coalitions which violate (9). Finally, once a group of coalitions merges into a larger coalition, the number of merging possibilities for the remaining users will decrease. In summary, although in worst case scenarios the merge process requires around $\frac{M_t \cdot (M_t-1)}{2}$ attempts, in practice, the process is far less complex.

At first glance, the *split* rule can be seen as a complex operation. For instance, splitting can involve finding all the possible partitions of the set formed by the users in a coalition In set theory, the number of all possible partitions of a set, i.e., a coalition in our case, is given by a value known as the Bell number which grows exponentially with the number of elements in the set, i.e., the number of users in the coalition [23]. In practice, since the split operation is restricted to each formed coalition in the network, it will operate on relatively small sets. As per (4), the size of each coalition is limited by the increasing cost as well as by the users' locations, thus, the coalitions formed in the proposed game are relatively small. Consequently, the split complexity will be limited to finding all possible partitions for small sets which can be affordable in terms of complexity. This complexity is further reduced by the fact that a coalition is not required to search for all the split forms. Typically, a coalition does not need to go through all the possible partitions As soon as a coalition finds a split form verifying the Pareto order, the users in this coalition will split, and the search for further split forms is not required. Briefly, the practical aspects dictated by the wireless network such as the increasing cost, users' locations and sequential search will significantly diminish the split complexity. The

reduction in the merge-and-split complexity will be further corroborated through simulations in Section V.

IV. FAIRNESS CRITERIA FOR PAYOFF DIVISION WITHIN COALITION

In this section, various fairness rules for dividing the utility $v(S)$ among the members of coalition S are inspected. A merge or a split operation by Pareto order directly depends on the fairness criterion selected for payoff division.

A. Egalitarian Fair (EF)

The most simple division method is to divide the *extra* utility *equally* among users. In other words, the utility of user i among the coalition S is

$$\phi_i = \frac{1}{|S|}\left(v(S) - \sum_{j \in S} v(\{j\})\right) + v(\{i\}),\tag{12}$$

where $v(\{i\})$ and $v(\{j\})$ are the non-cooperative payoffs of users i and j. EF does not imply dividing the whole utility equally but rather the *extra* benefits equally while conserving individual rationality.

B. Proportional Fairness (PF)

The EF is a very simple and strict fairness criterion. However, in practice, the user experiencing a good channel might not be willing to cooperate with a user under bad channel conditions, if the extra is divided equally. To account for the channel differences, we use a criterion named proportional fairness (PF) [24], in which the extra benefit is divided in weights according to the users' non-cooperative utilities. Thus,

$$\phi_i = w_i \left(v(S) - \sum_{j \in S} v(\{j\})\right) + v(\{i\}),\tag{13}$$

where $\sum_{i \in S} w_i = 1$ and within the coalition $\frac{w_i}{w_j} = \frac{v(\{i\})}{v(\{j\})}$. Thus, within the coalition for PF, the users with good channel conditions deserve more extra benefits than the users with bad channel conditions.

C. Shapley Value Fairness (SV)

Additional fairness criteria can be used while dividing the worth of a coalition among its members. For instance, another measure of fairness is given using the Shapley value (SV) [16].

Definition 7: A *Shapley value* ϕ^v is a function that assigns to each possible characteristic function v a vector of real numbers, i.e., $\phi^v = (\phi_1^v, \phi_2^v, \ldots, \phi_{M_t}^v)$, where ϕ_i^v represents the worth or value of user i in the game (N, v). There are four Shapley axioms that ϕ^v must satisfy:

1) *Efficiency Axiom*: $\sum_{i \in N} \phi_i^v = v(N)$.

2) *Symmetry Axiom*: If user i and user j are s. t. $v(S \cup \{i\}) = v(S \cup \{j\})$ for every coalition S not containing user i and user j, then $\phi_i^v = \phi_j^v$.

3) *Dummy Axiom*: If user i is s. t. $v(S) = v(S \cup \{i\})$ for every coalition S not containing i, then $\phi_i^v = 0$.

4) *Additivity Axiom*: If u and v are characteristic functions, then $\phi^{u+v} = \phi^{v+u} = \phi^u + \phi_i^v$.

It is shown [16], [25] that there exists a unique function ϕ satisfying the Shapley axioms given by

$$\phi_i^v = \sum_{S \subseteq N-\{i\}} \frac{(|S|)!(|N| - |S|)!}{(|N| + 1)!}[v(S \cup \{i\}) - v(S)]. \tag{14}$$

The SV provides a fair division which takes into account the random-ordered joining of the users in the coalition. Under the assumption of randomly-ordered joining, the Shapley function of each user is its expected marginal contribution when it joins the coalition [16].

In our transmitter cooperation game, we are interested in dividing payoffs using SV among the users in any formed coalition G by merge or split. Thus, the payoff division by SV occurs by applying (14) on each restricted game (G, v) in the structure. In [26], they proved that, in a game with coalitional structure, the SV of the whole game (N, v) is found by using the SV on the game restricted to each coalition G in the structure as in our case. For a non-superadditive game, the SV might not be individually rational, however, the proposed algorithm handles it with an appropriate merge-and-split decision by Pareto order.

D. Maximin Fairness Using the Nucleolus (NU)

A stricter fairness rule for payoff division is given using the nucleolus (NU). We introduce the concepts of *excess*, *kernel*, and *nucleolus* [16], [25], [27]. For a given characteristic function v, an allocation $\boldsymbol{x}$ is found such that, for each coalition S and its associated dissatisfaction, an optimal allocation is calculated to minimize the maximum dissatisfaction. The dissatisfaction is quantified as follows.

Definition 8: The measure of dissatisfaction of an allocation ϕ^v for a coalition S is defined as the *excess* $e(\phi^v, S) = v(S) - \sum_{j \in S} \phi_j^v$. A *kernel* of v is the set of all allocations ϕ^v such that

$$\max_{S \subseteq N-\{j\}, i \in S} e(\phi^v, S) = \max_{G \subseteq N-\{i\}, j \in G} e(\phi^v, G). \tag{15}$$

The kernel states that if players i and j are in the same coalition, then the highest excess that i can make in a coalition without j is equal to the highest excess that j can make in a coalition without i.

Definition 9: Define $\boldsymbol{O}(\phi^v)$ as the vector of all excesses in a game (N, v) arranged in non-increasing order (except the excess of the grand coalition $\{N\}$). A vector $\boldsymbol{y} = (y_1, \ldots, y_k)$ is said to be lexographically less than a vector $\boldsymbol{z} = (z_1, \ldots, z_k)$ (denoted by $\mathbf{y} \prec_{\text{lex}} \mathbf{z}$) if $\exists l \in \{1, \ldots, k\}$ where $y_1 = z_1, y_2 = z_2, \ldots, y_{l-1} = z_{l-1}, y_l < z_l$. A group rational allocation $\boldsymbol{\delta}^v$ is a *nucleolus* (or prenucleolus) if for every other ϕ^v, $\boldsymbol{O}(\boldsymbol{\delta}^v) \prec_{\text{lex}} \boldsymbol{O}(\phi^v)$. Hence, the nucleolus is the group rational allocation $\boldsymbol{\delta}^v$ which *minimizes the excesses* in a non-increasing order.

The NU of a coalitional game exists and is unique. The NU is group rational, lies in the kernel of the game and satisfies the symmetry and dummy axioms. Moreover, if the core is not empty, the NU is in the core. In other words, the NU is the best allocation under a min-max criterion.

Our main interest is to use the NU to find the allocation $\phi^v = (\phi_1^v, \ldots, \phi_{|G|}^v)$ for a coalition $G \subset N$ that will potentially form by merge (or split); that is the NU of the restricted game (G, v). Unlike the SV, the NU of a game (N, v) with a coalition structure T is not the same as the NU of the restricted games (G, v) with $G \in T$ [26]. When the NU is considered over the restricted game (G, v), it will not minimize the excesses pertaining to coalitions formed by a combination of players belonging to G and players outside G. A similar reasoning also applies to the kernel. However, finding the NU of the whole game (N, v) requires a centralized intelligence which can also find the excesses among the disjoint coalitions that are formed; which contradicts with the goal of deriving a distributed clustering algorithm. Thus, for distributed cooperation, it suffices to use the NU of the restricted game (G, v) as it allows a payoff allocation which minimizes excesses inside G while having all the fairness properties of the NU restricted over (G, v). If a game is non-superadditive the NU may not be individually rational; however, this will simply be handled by adequate merge or split decisions.

V. SIMULATION RESULTS AND ANALYSIS

For simulations, a BS is placed at the origin with $M_r = 3$ equally spaced antennas. Without loss of generality, at the receiver, we consider antennas that are separated enough without phases [1]. Users are randomly located within a square of 4km$\times$ 4km around the BS. The propagation loss $\alpha = 3$ and the path loss constant $\kappa = 1$. The power constraint per slot is $\tilde{P} = 0.01$ W, the cost SNR target for information exchange is $\nu_0 = 10$ dB and noise level is -90 dBm. 10000 independent runs with randomly located users are conducted for different network sizes.

Fig. 2 shows the resulting network's structure in terms of the average number of coalitions for different fairness and networks sizes. Cooperation organizes the network in clustered coalitions with the average number of users per coalition seen through Fig. 2 by dividing the network size with the number of coalitions. The number of coalitions and the average number of users per coalition increase with the network size due to the availability of more partners for forming coalitions. Moreover, as per Fig. 2, in general, the network tends to self-organize into a large number of small coalitions rather than a small number of large coalitions. The PF division yields the largest average coalition size for

---

[1]This choice provides a lower bound on the performance gain of our algorithm (i.e. the gains mainly stemming from the transmitters cooperation which is the main objective of the paper); considering different phases certainly yields an additional multiplexing gain and it does not affect the analysis or the results hereafter.

all networks since it provides cooperation incentives to users with better non-cooperative channels by assigning them a larger weight. In contrast, the SV and NU division yield smaller coalitions due to additional fairness constraints imposed on the division (e.g. Shapley axioms and excess minimization for nucleolus). The results in this figure also provide us with an insight on the complexity of the merge-and-split operations as discussed in Section III-D. For example, for the merge operation, we note that one run of the proposed algorithm transforms a network of $M_t = 100$ non-cooperative users into a network consisting of at most an average of 38.63 coalitions (NU case). Thus, the maximum number of merge attempts for future runs of the algorithm is reduced by a factor of almost 2.5. In addition, because the network self-organizes into a large number of small coalitions, the complexity of the split operation is generally affordable as it will be restricted to small coalitions. For instance, for a network as large as $M_t = 100$ users, the maximum average coalition size (over which a split may be applicable) is only 2.6 users for PF.

In Fig. 3, we show the average total individual user payoff (rate) improvement achieved during the whole transmission time scale as a function of the network size. We compare the performance of the proposed algorithm to that of the non-cooperative case as well as the optimal partition found by a centralized entity through exhaustive search. For the cooperative case, the average user's payoff increases with the number of users since the possibility of finding cooperating partners increases. In contrast, the non-cooperative approach presents an almost constant performance with different network sizes. Cooperation presents a significant advantage over the non-cooperative case in terms of average individual utility for all fairness types, and this advantage increases with the network size. The PF division presents the best performance, as it allows an improvement of up to 40.42% over the non-cooperative case at $M_t = 100$. This result also highlights the trade off between fairness and cooperation gains. For instance, while the PF presents an advantage in terms of payoff gain, since it allows larger coalitions to form (Fig. 2), the SV and NU present lower gains but more fairness in allocating payoffs (Section IV). Furthermore, compared to the optimal solution, clearly the proposed merge and split algorithm achieve a highly comparable performance with a performance loss not exceeding 1% at $M_t = 20$ users. This clearly shows that, by using the proposed distributed merge-and-split algorithm, the network can achieve a performance that is very close to optimal. Finally, note that, for networks larger than 20 users, finding the optimal partition through exhaustive search is mathematically and computationally untractable.

Depending on the chosen fairness criteria for payoff division, the resulting network topology changes as the merge-and-split through Pareto order becomes different. For showing the fairness effect on the network structure, we show in Fig.4, for a random network of $M_t = 6$ users, the users' positions and the *final structure* for each fairness type. Moreover, the payoffs of relevant coalitions in this network are shown in Table II. The merge process starts with User

3 (best non-cooperative utility) which attempts to merge with User 6 (closest user). Forming coalition $\{3, 6\}$ allows both users to improve their payoff, for all fairness. The PF gives a different division than other fairness types as it assigns a higher weight to User 3 which has the best non-cooperative utility. Subsequently, for all fairness types except the SV, coalition $\{3, 6\}$ merges with User 1 to yield a 3-users coalition. For the SV, since User 6 cannot improve its payoff through this merge, coalition $\{3, 6, 1\}$ does not form. In this case, coalition $\{3, 6\}$ merges successfully with User 2 (the next lowest cost) as shown in Table II. Moreover, for the SV, coalition $\{3, 6, 2\}$ tries to merge with User 1; but the Pareto order cannot be verified and $\{3, 6, 2\}$ cannot merge any further. For the other fairness types, the newly formed coalition $\{3, 6, 1\}$ tries to merge with User 4. This merge is possible for EF and PF since all 4 users can improve their payoff. However, for the NU coalition $\{3, 6, 1, 4\}$ does not form since the payoff of User 6 would decrease by the merge. For EF, coalition $\{3, 6, 1, 4\}$ cannot merge with User 2 since the payoffs of Users 3, 6, 1 and 4 will degrade. As a result, for EF, this coalition cannot merge any further and the final network structure is $T = \{\{3, 6, 1, 4\}, \{2\}, \{5\}\}$. In contrast, for PF, coalition $\{3, 6, 1, 4\}$ is able to merge with User 2 but it can no longer merge with the remaining User 5. However, the grand coalition cannot form since the cost for cooperation would be 0.0939 W which is larger than the total power constraint $\tilde{P} = 0.01$ W yielding $v(\{3, 6, 1, 4, 2, 5\}) = 0$. For the SV, User 1 will further merge with User 4 but coalition $\{1, 4\}$ cannot merge any further with User 5 since $v(\{1, 4, 5\}) = 0$. For the NU, User 4 merges with User 2 forming coalition $\{4, 2\}$ which can no longer merge with User 5 as $v(\{4, 2, 5\}) = 0$. Finally, for all 2-users coalitions the EF, SV and NU division coincide as they obey the same equations in this case [16], [25].

Furthermore, a network with $M_t = 10$ users is generated where the users are located in a way that a $\mathbb{D}_c$-stable partition exists. Fig. 5 shows that the proposed algorithm converges to the final $\mathbb{D}_c$-stable network partition $T = \{T_1, \ldots, T_6\}$ (valid for all fairness criteria). Cond. A) for $\mathbb{D}_c$-stability is easily verified for coalitions consisting of at most 2 users since such coalitions do not form unless the Pareto order is internally verified (definition of the merge rule). For the 3-users coalition $T_2 = \{7, 9, 10\}$ Table III shows the payoffs of the different sub-coalitions for the various fairness types. Table III shows that the Pareto order is internally verified for T_2, that is $\forall A, B \subset T_2; \{A \cup B\} \rhd \{A, B\}$ for all fairness cases. In addition, by inspecting Fig. 5 it is clear that any two users belonging to T-incompatible coalitions are separated by a distance larger than the maximum distance, which is $\hat{d}_0 = 0.793$ km, computed using Theorem 4 for the simulation parameters. Thus, Theorem 3 is satisfied and cond. B) is verified. For example, for the T-incompatible coalition $G = \{4, 7\}$ equation (6) yields $v(G) = 0$ and thus $\phi_4^v(G) = \phi_7^v(G) = 0$ due to the distance between Users 4 and 7. The projection of G in T is $G[T] = \{\{4, 7\} \cap T_1, \{4, 7\} \cap T_2, \{4, 7\} \cap T_3, \{4, 7\} \cap T_4, \{4, 7\} \cap$

$T_5, \{4, 7\} \cap T_6\} = \{\{4\}, \{7\}\}$. In $G[T]$, the payoffs of users 4 and 7 are respectively $\phi_4^v(G[T]) = v(\{4\}) = 7.6069$ and $\phi_7^v(G[T]) = v(\{7\}) = 3.1077$, by Pareto order $\phi_4^v(G[T]) > \phi_4^v(G)$ and $\phi_7^v(G[T]) > \phi_7^v(G)$, thus, $G[T] \rhd G$.

In Fig. 6, we plot the percentage of coalitions prone to deviations (averaged over random starting positions of the users) in the final network structure for different fairness criteria and different network sizes. First and foremost, this figure shows that the percentage of coalitions prone to deviation is generally small and does not exceed 10% for a relatively large network with $M_t = 100$ users, for the PF fairness criteria. This corroborates the fact that, by using Pareto order merge and split, the number of coalitions prone to deviations is generally small. Moreover, in this figure, we can see that for stringent fairness criteria such as the SV and the NU, the final network partitions contain no coalitions prone to deviation even if the partition is $\mathbb{D}_{hp}$-stable. This is mainly due to two reasons. On one hand, the SV and the NU have strict fairness requirements on the coalitions, hence yielding more cohesive and stable coalitions. On the other hand, for such fairness criteria, the size of the coalitions is generally small as previously shown. In contrast, the PF presents the largest percentage due to the large size of the coalitions that form for PF division as well as the fact that the PF can give incentives to deviate for coalition members that have a good non-cooperative utility. Similarly, the EF presents a percentage of coalitions prone to deviation, due to the coalition sizes that are generally smaller than PF, and to the fact that extra utility benefits are equally distributed among the users, giving less incentive for users to deviate than in the PF. In a nutshell, Fig. 6 summarizes the possible loss of stability that can occur when the final network partitions are $\mathbb{D}_{hp}$-stable, while highlighting an interesting fairness-stability trade off.

In Fig. 7, we show how the merge-and-split algorithm handles mobility. The network setup is the same as in Fig. 5 with User 7 moving along the y-axis upwards for 1.4 km. The figure depicts the results for PF, as the other division types yield similar curves and are omitted for space limitation. When User 7 moves upwards, it becomes closer to the BS while the cost in coalition $\{7, 9, 10\}$ increases. As a result, its utility increases at first while the utilities of Users 9 and 10 decrease, since the PF division gives a higher weight to the user with the best non-cooperative utility, i.e., User 7. When User 7 covers 0.3 km, the cost in coalition $\{7, 9, 10\}$ increases significantly, and the utilities of all three users drop. Afterwards, at 0.4 km the splitting step occurs as User 7 splits from coalition $\{7, 9, 10\}$ and all three users improve their utilities by Pareto order. User 7 continues to improve its utility as it gets closer to the BS. Once User 7 moves about 1.3 km, it will be beneficial to Users 7, 4 and 8 to form a 3-user coalition. The merge algorithm allows User 7 to join the coalition of Users 4 and 8. Therefore, the payoffs of Users 4, 7 and 8 start improving significantly. These results show how merge-and-split algorithm operates in a wireless network.

The algorithm's performance is further investigated in a mobile network of $M_t = 50$ users (random walk mobility)

for a period of 5 minutes. For $M_t = 50$, each TDMA transmission requires $50 \cdot \theta$ seconds with θ the slot duration, we let $\theta = 10$ ms. The results in terms of frequency of merge-and-split operations per minute are shown in Fig. 8 for various speeds. As the speed increases, for all fairness types, the number of merge-and-split operations increases due to the changes in the network structure incurred by mobility. Fairness types that yield large coalitions, incur a higher frequency of merge-and-split since such coalitions require additional merge operations and are more prone to splitting due to mobility. In this regard, the PF and EF record the highest frequencies notably at high speeds. EF has the highest frequency since it yields coalitions with size comparable to PF, but these coalitions are more prone to split at high mobility since EF divides the extra benefit equally without accounting for the users' non-cooperative utilities like PF. Fig. 9 shows how the structure of a mobile network of $M_t = 50$ users evolves with time for PF (other fairness are omitted for space limitation), while the velocity of the users is constant equal to 120 km/h for a period of 5 minutes. As the users move, the structure of the network changes, with new coalitions forming and others splitting. The network starts with a non-cooperative structure made up of 50 independent users. In the first step, the network self-organizes in 20 coalitions with an average of 2.5 users per coalition. With time the structure changes as new coalitions form or others split. After the 5 minutes have elapsed the final coalition structure is made up of 23 coalitions that is an average of 2.17 users per coalition. Finally, the behavior and performance of the network for different cost SNRs ν_0 is assessed. Fig. 10 shows the average total user payoff achieved for different cost SNRs for $M_t = 100$ users. This figure shows that cooperation maintains utility gains at almost all costs; however, as the cost increases these gains decrease converging towards the non-cooperative gains at high cost since cooperation becomes difficult due to the cost. In fact, at $\nu_0 = 40$ dB all fairness types yield an average number of coalitions of around 96, i.e., almost every user acts non-cooperatively, hence the performance is close to that of the non-cooperative case.

VI. CONCLUSIONS

In this paper, we constructed a novel game theoretical framework suitable for modeling cooperation among wireless network nodes. The framework is applied to the transmitter cooperation game with cost. Unlike existing literature which sought algorithms to form the grand coalition of transmitters, we inspected the possibility of forming disjoint independent coalitions using a novel algorithm from coalitional game theory. We proposed a simple and distributed merge-and-split algorithm for forming coalitions and benefiting from spatial gains. The proposed algorithm enables single antenna transmitters to cooperate and self-organize. Moreover, the algorithm can be implemented in a distributed way since the decision to merge or split is individually taken by each user based on the Pareto order, implying that at least one of the users is able to improve his payoff without hurting the others. The stability of the network partitions is studied through the novel concept of defection function $\mathbb{D}$. The proposed algorithm converges to a network partition

that is $\mathbb{D}_{hp}$-stable, i.e., no user has an incentive to leave this partition through merge or split. Depending on the location of the users, the proposed algorithm can also converge to a Pareto optimal (in terms of payoff distribution) $\mathbb{D}_c$-stable partition which is a unique outcome of any merge-and-split iteration. The derived algorithm efficiently adapts to mobility as the coalitions form or split depending on the users' time varying positions. Simulation results show how the proposed algorithm allows the network to self-organize and improve the user payoff by 40.42%.

ACKNOWLEDGEMENT

REFERENCES

[1] K. Yazdi, H. E. Gamal, and P. Schitner, "On the design of cooperative transmission schemes," in *Proc. Allerton Conf. on Communication, Control, and Computing*, Illinois, IL, USA, Oct. 2003.

[2] C. Ng and A. Goldsmith, "Transmitter cooperation in ad-hoc wireless networks: Does dirty-payer coding beat relaying?" *IEEE Int. Symposium for Information Theory*, Jun. 2004.

[3] M. Jindal, U. Mitra, and A. Goldsmith, "Capacity of ad-hoc networks with node cooperation," in *Proc. Int. Symp. on Information Theory*, Chicago, IL, USA, Jun. 2004, p. 271.

[4] A. Coso, S. Savazzi, U. Spagnolini, and C. Ibars, "Virtual MIMO channels in cooperative multi-hop wireless sensor networks," in *Proc. Conf. on Information Sciences and Systems*, New Jersey, NY, USA, Mar. 2006, pp. 75–80.

[5] A. MacKenzie, L. DaSilva, and W. Tranter, *Game theory for wireless engineers*. Morgan and Claypool Publishers, Mar. 2006.

[6] S. Mathur, L. Sankaranarayanan, and N. Mandayam, "Coalitional games in receiver cooperation for spectrum sharing," in *Proc. Conf. on Information Sciences and Systems*, New Jersey, NY, USA, Mar. 2006, pp. 949–954.

[7] ——, "Coalitional games in cooperative radio networks," in *Proc. Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA, Jan. 2006, pp. 1927–1931.

[8] R. La and V. Anantharam, "A game-theoretic look at the Gaussian multiaccess channel," in *Proc. of the DIMACS Workshop on Network Information Theory*, New Jersey, NY, USA, Mar. 2003.

[9] J. Cai and U. Pooch, "Allocate fair payoff for cooperation in wireless ad hoc networks using shapley value," in *Proc. Int. Parallel and Distributed Processing Symposium*, Santa Fe, NM, USA, Apr. 2004, pp. 219–227.

[10] E. Berez and R. Adve, "Cooperation and routing in multi-hop networks," in *Proc. Int. Conf. on Communications*, Glasgow, Scotland, Jun. 2007, pp. 4767–4772.

[11] Z. Han and K. J. Liu, *Resource allocation for wireless networks: basics, techniques, and applications*. Cambridge University Press, 2008.

[12] Z. Han and V. Poor, "Coalition games with cooperative transmission: a curse for the curse of boundary nodes in selfish packet-forwarding wireless networks," in *Proc. Int. Symp. on Modeling and Opt. in Mobile Ad Hoc and Wireless Networks*, Limasol, Cyprus, Apr. 2007.

[13] Z. Han, C. Pandana, and K. J. R. Liu, "A self-learning repeated game framework for optimizing packet forwarding networks," in *Proc. IEEE Wireless Communications and Networking Conf.*, New Orleans, LA, USA, Mar. 2005.

[14] I. E. Telatar, "Capacity of multi-antenna gaussian channels," *Euro. Trans. Telecommun.*, vol. 10, pp. 585–595, Dec. 1999.

[15] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications*. Cambridge, United Kingdom: Cambridge University Press, Jan. 2007.

[16] G. Owen, *Game theory, 3rd edition*. London, UK: Academic Press, Oct. 1995.

[17] T. Sandholm, K. Larson, M. Anderson, O. Shehory, and F. Tohme, "Coalition structure generation with worst case guarantees," *Artifical Intelligence*, vol. 10, pp. 209–238, Jul. 1999.

[18] D. Ray, *A Game-Theoretic Perspective on Coalition Formation*. New York, USA: Oxford University Press, Jan. 2007.

[19] K. Apt and T. Radzik, "Stable partitions in coalitional games," *arXiv:cs/0605132v1 [cs.GT]*, May 2006.

[20] K. Apt and A. Witzel, "A generic approach to coalition formation," in *Proc. of the Int. Workshop on Computational Social Choice (COMSOC)*, Amsterdam, the Netherlands, Dec. 2006.

[21] ——, "A generic approach to coalition formation," *arXiv:0709.0435v1 [cs.GT]*, Sep. 2007.

[22] C. S. R. Murthy and B. Manoj, *Ad Hoc Wireless Networks: Architectures and Protocols*. New Jersey, NY, USA: Prentice Hall, Jun. 2004.

[23] L. Lovasz, *Combinatorial Problems and Exercises 2nd edition*. Amsterdam, the Netherlands: North-Holland, Aug. 1993.

[24] F. Kelly, "Charging and rate control for elastic traffic," *Euro. Trans. Telecommun.*, vol. 8, pp. 33–37, Jan. 1997.

[25] R. B. Myerson, *Game Theory, Analysis of Conflict*. Cambridge, MA, USA: Harvard University Press, Sep. 1991.

[26] R. Aumann and J. Drèze, "Cooperative games with coalition structures," *Int. Journal of Game Theory*, vol. 3, pp. 317 – 237, Dec. 1974.

[27] T. Ferguson, *Game Theory*. Online course, UCLA, Department of Mathematics.

[28] W. Saad, Z. Han, M. Debbah, and A. Hjørungnes, "A distributed merge and split algorithm for fair cooperation in wireless networks," in *Proc. Int. Conf. on Communications, Workshop on Cooperative Communications and Networking*, Beijing, China, May 2008.
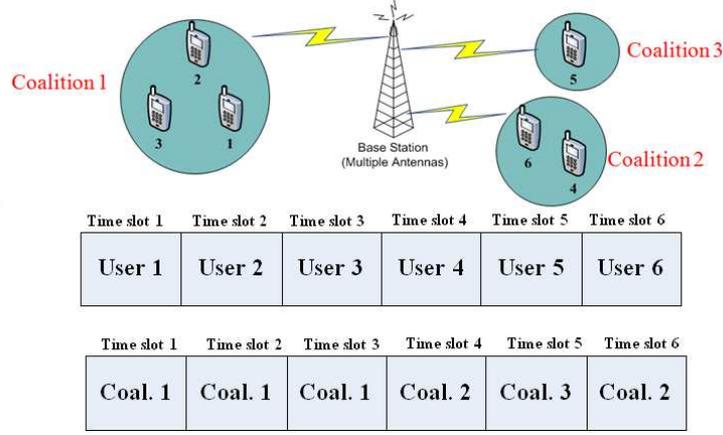
Fig. 1. User cooperation example coalitions and TDMA transmission.

TABLE I
ONE STAGE OF THE PROPOSED MERGE-AND-SPLIT ALGORITHM.

Initial State

The network is partitioned by $T = \{T_1, \ldots, T_k\}$ (At the beginning of all time $T = N = \{1, \ldots, M_t\}$ with non-cooperative users).

Proposed Coalition Formation Algorithm

Phase I - Adaptive Coalition Formation:

In this phase, coalition formation using merge-and-split occurs.

repeat

a) Coalitions begin the local search merge operation in Section III-B:
$W = \mathbf{Merge}(T)$. In the merge process, the coalitions in T seek neighboring partners for cooperation through Pareto order.

b) Coalitions in W decide to split based on the Pareto order.
$T = \mathbf{Split}(W)$. In the split phase, each coalition may divide into split coalitions based on the Pareto order.

until merge-and-split iteration terminates.

Phase II - Virtual MIMO Transmission:

The coalitions transmit during the time scale M_t with 1 coalition per slot with each coalition occupying all the time slots previously held by its members.

The proposed algorithm is repeated periodically, enabling the users to autonomously self-organize and adapt the topology to environmental changes such as mobility.
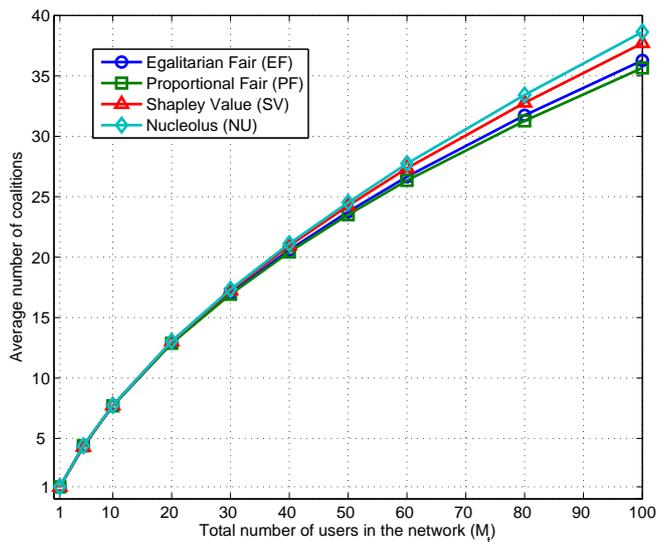

Fig. 2. Resulting network structure shown through the average number of coalitions formed for different network sizes.
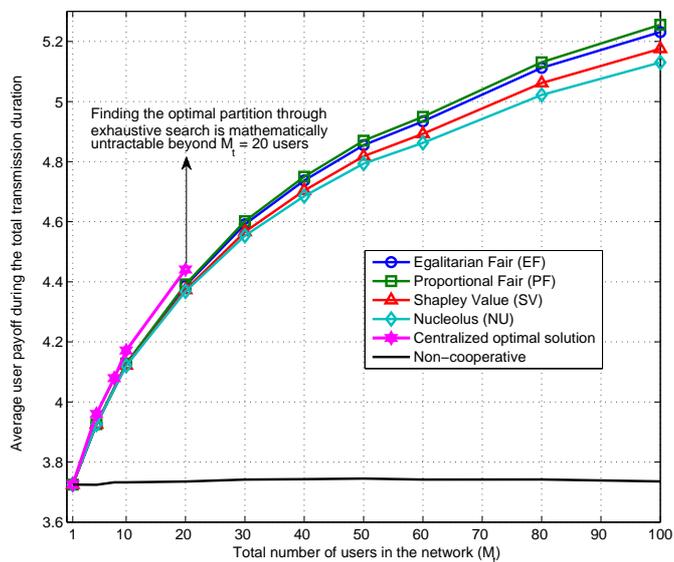

Fig. 3. Cooperation gains in terms of the average individual user payoff achieved by the proposed scheme during the whole transmission duration compared to the non-cooperative case and the centralized optimal solution for different network sizes and different fairness criteria.
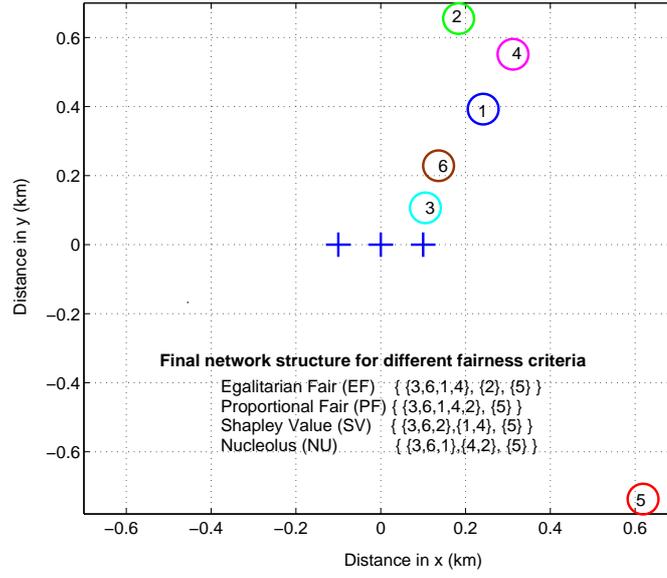
Fig. 4. User positions and network structure for different fairness criteria.

TABLE II

DEPENDENCE OF THE FINAL NETWORK STRUCTURE ON THE DIFFERENT FAIRNESS CRITERIA FOR PAYOFF DIVISION AMONG THE COALITION USERS FOR NETWORK OF FIG.4.

| Non-Cooperative Payoffs									
$v(\{1\}) = 8.2827, v(\{2\}) = 6.5697, v(\{3\}) = 13.5481, v(\{4\}) = 6.9005, v(\{5\}) = 5.3854, v(\{6\}) = 10.6611$									
Individual Payoffs for Relevant Coalitions									
	Egalitarian Fair					Proportional Fair				
	User 1	User 2	User 3	User 4	User 6	User 1	User 2	User 3	User 4	User 6
$\{3,6\}$	-	-	19.1089	-	16.2219	-	-	19.7720	-	15.5587
$\{3,6,1\}$	15.4172	-	20.6826	-	17.7956	13.7388	-	22.4726	-	17.6839
$\{3,6,2\}$	-	12.7609	19.7393	-	16.8523	-	10.5342	21.7237	-	17.0946
$\{3,6,1,4\}$	15.6885	-	20.9539	14.3063	18.0669	14.5114	-	23.7364	12.0897	18.6783
$\{3,6,1,4,2\}$	15.1963	13.4833	20.4617	13.8141	17.5747	14.5121	11.5108	23.7376	12.0904	18.6793
$\{1,4\}$	9.4233	-	-	8.0411	-	9.5271	-	-	7.9372	-
$\{4,2\}$	-	7.5588	-	7.8896	-	-	7.5345	-	7.9139	-
$\{3,6,2,1\}$	15.1604	13.4475	20.4258	-	17.5388	14.1162	11.1967	23.0900	-	18.1697
	Shapley					Nucleolus				
	User 1	User 2	User 3	User 4	User 6	User 1	User 2	User 3	User 4	User 6
$\{3,6\}$	-	-	19.1089	-	16.2219	-	-	19.1089	-	16.2219
$\{3,6,1\}$	12.8630	-	25.1391	-	15.8931	13.4236	-	24.0180	-	16.4536
$\{3,6,2\}$	-	10.1499	22.9161	-	16.2865	-	10.2958	22.6245	-	16.4323
$\{3,6,1,4\}$	14.1229	-	26.8091	11.9101	16.1738	13.1070	-	28.9640	11.0105	15.9342
$\{3,6,1,4,2\}$	13.9112	10.9714	27.6011	11.8467	16.1999	12.7567	9.0421	32.0028	10.7246	16.0041
$\{1,4\}$	9.4233	-	-	8.0411	-	9.4233	-	-	8.0411	-
$\{4,2\}$	-	7.5588	-	7.8896	-	-	7.5588	-	7.8896	-
$\{3,6,2,1\}$	13.7479	11.0347	25.7791	-	16.0108	12.7514	9.6235	28.8135	-	15.3842
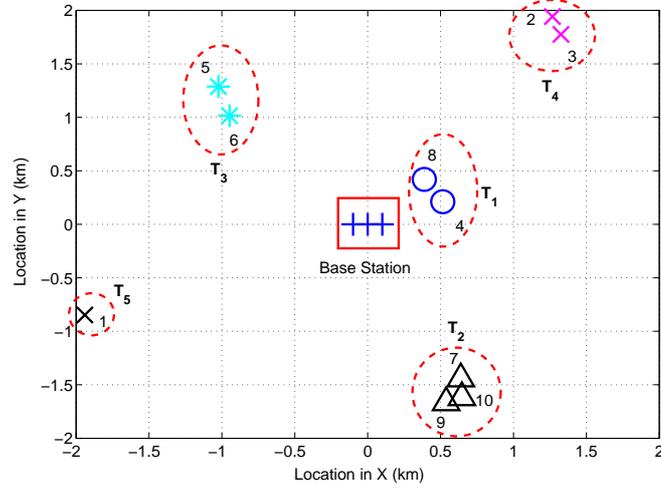
Fig. 5. Convergence of the algorithm to a final $\mathbb{D}_c$-stable partition.

TABLE III

PAYOFFS FOR COALITION $T_2 = \{7, 9, 10\}$ OF FIGURE 5 AND ITS SUB-COALITIONS (DURING $M_t = 10$ SLOTS).

|	Egalitarian Fair			Proportional Fair			Shapley			Nucleolus		
	User 7	User 9	User 10	User 7	User 9	User 10	User 7	User 9	User 10	User 7	User 9	User 10
$\{7\}$	3.1077	-	-	3.1077	-	-	3.1077	-	-	3.1077	-	-
$\{9\}$	-	2.7173	-	-	2.7173	-	-	2.7173	-	-	2.7173	-
$\{10\}$	-	-	2.7345	-	-	2.7345	-	-	2.7345	-	-	2.7345
$\{7, 9\}$	3.9829	3.5925	-	4.0416	3.5338	-	3.9829	3.5925	-	3.9829	3.5925	-
$\{7, 10\}$	4.0075	-	3.6352	4.0648	-	3.5779	4.7863	-	4.4406	4.7863	-	4.4406
$\{9, 10\}$	-	3.5996	3.6177	-	3.6206	3.5967	-	3.5996	3.6177	-	3.5996	3.6177
$T_2 = \{7, 9, 10\}$	4.5175	4.1452	4.1271	4.6431	4.0869	4.0598	4.5210	4.1558	4.1131	4.5244	4.1663	4.0991
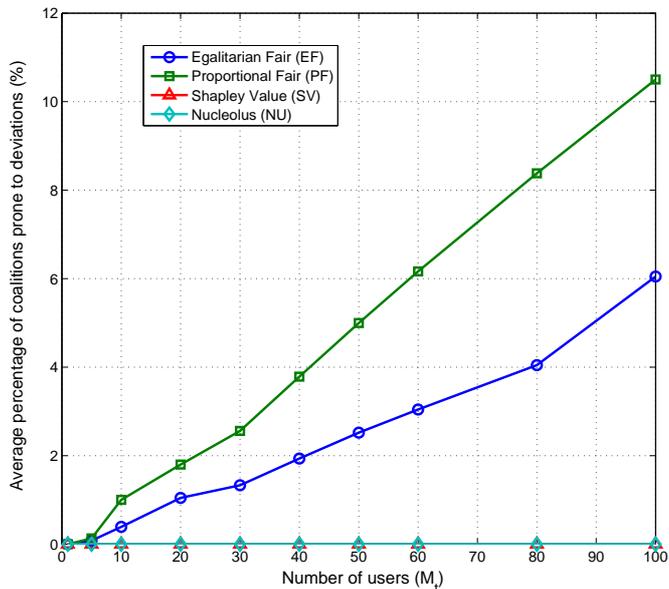
Fig. 6. Average percentage of coalitions prone to deviations in the final network structure vs. number of users M_t for different fairness criteria.
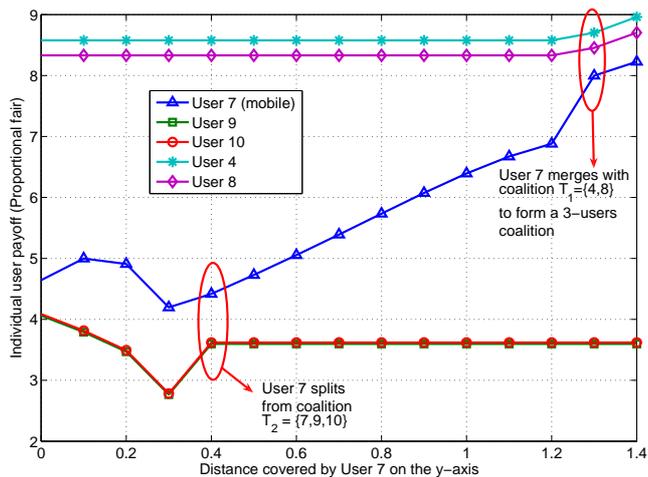

Fig. 7. Adaptive coalition formation: coalition merging/splitting due to mobility of the users.
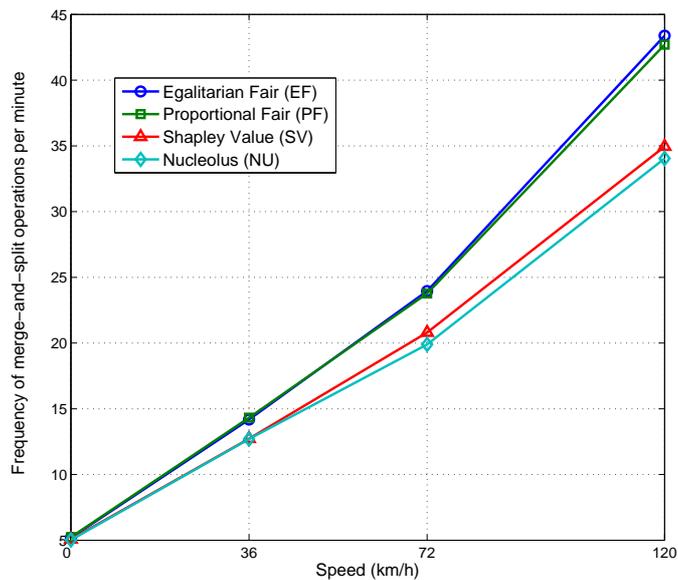
28

Fig. 8. Frequency of merge-and-split operations per minute for different speeds in a mobile network of $M_t = 50$ users.
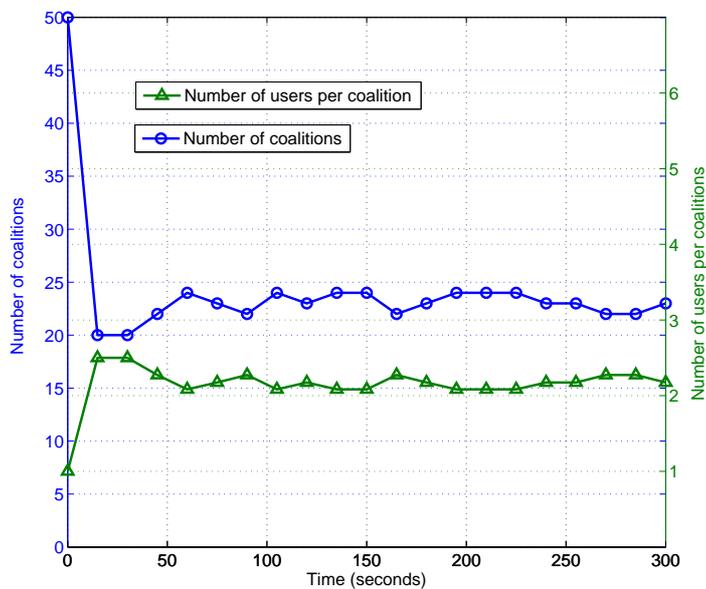

Fig. 9. Network structure changes with time for $M_t = 50$ users, a constant speed of 120 km/h and a proportional fair division.
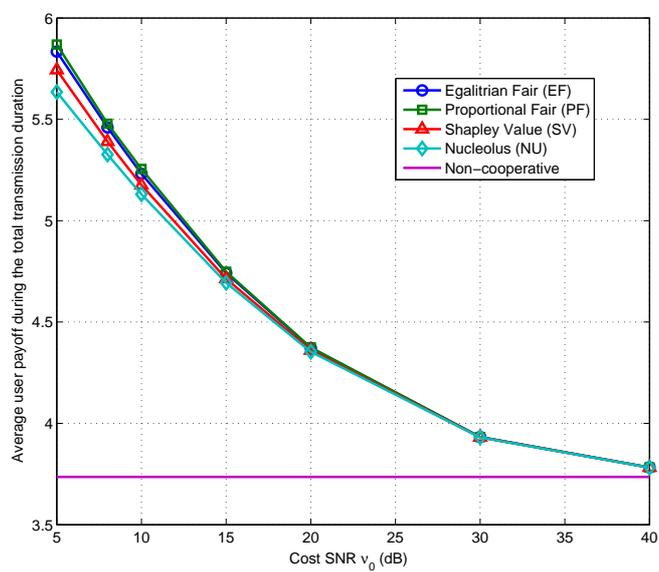
Fig. 10. Average individual user payoff for different cost SNRs ν_0 for a network having $M_t = 100$ users for the cooperative and non-cooperative cases.