STABLE MATCHINGS FOR RESOURCE ALLOCATION IN WIRELESS NETWORKS

Eduard A. Jorswieck

Dresden University of Technology
Communications Laboratory, Communications Theory
E-Mail: Eduard.Jorswieck@tu-dresden.de

ABSTRACT

In this paper, we apply and extend the theory of one-to-one and many-to-one matching markets to the resource allocation in wireless communications. We develop a general framework to find stable matchings of users and resources based on the channel and context aware preference lists of users and resources. The score-based, maximum throughput, and proportional fair scheduler do not lead necessarily to a stable matching. We apply the user and resource proposing deferred acceptance algorithm in order to find stable matchings and to identify their properties. If the preference lists of users and resources are strict and based on the same information, e.g., the channel state, the stable matching is always unique. The optimality properties of stable matchings are characterized. The framework is illustrated by an example ad-hoc communications scenario.

***Index Terms*—** Resource management, Cross-layer design, Scheduling, Stable Matching

1. INTRODUCTION

Matchings occur in our everyday life. When there are non-divisible goods and entities which have different interests in these goods, there is a corresponding matching market. There are one-sided or two-sided markets depending on each preference for potential goods of the other side. Matchings can have different properties (Pareto optimality, maximum rank, stability) which are of different importance in different applications.

One of the most popular matching problems is the *stable marriage problem* if a set of men and a set of women decide on who to get married with; that is the marriage matching market. This is usually a two-sided one-to-one matching problem. Both sides have different preference lists. In the marriage market, these preferences are built on phenotypic properties, e.g., hair color, weight, or face features. Since society as well as the couples themselves have interest in enduring marriages, the notion of stability is important here [1].

Part of this work has been performed in the framework of the European research project SAPHYRE, which is partly funded by the European Union under its FP7 ICT Objective 1.1 - The Network of the Future.

The first who studied stable matchings and showed that there always exists at least one stable matching by a constructive algorithm - the deferred acceptance algorithm - are Gale and Shapley [1].

In the monograph [2], one-to-one and many-to-one stable matchings, their construction, properties and applications to the labor market and college admissions are studied. Because of the many applications of matchings markets and their importance, there is a large body of work on stable matchings in one-to-one and one-to-many matching markets. In [3], the properties of the preference relations of colleges and students are analyzed and equivalence shown between Pareto efficiency, group strategyproofness, consistency, and acyclicality. An axiomatic framework for deferred acceptance is developed in [4]. An asymptotic analysis of incentive compatibility and stability in (asymptotic) large two-sided matching markets is performed in [5].

Due to the limited resources time, spectrum, and space, resource allocation problems are extensively studied in communications [6]. There is a large body of work available and we can only list a few representative examples which are related to the problems addressed here. Typical resource allocation problems include power allocation, bit loading, beamforming optimization, subcarrier selection, user grouping, and user selection. We focus on the problem of assigning resources (e.g. time-frequency chunks) to users. This is usually referred as user scheduling.

The downlink scheduling problem is difficult because it is a combinatorial problem of matching users to subcarriers. Even in its simplest case, i.e., a resource is matched exactly to one user, it cannot be solved easily or in closed form but it is has exponential complexity [7]. One-to-one stable matchings and the deferred acceptance algorithm were recently applied to cognitive radio systems in [8]. In multiple antenna systems multiple users are allocated to a single time-frequency chunk and the resource allocation problem is even more complex [9]. Often, resource allocation is formulated as an optimization problem [10]. If channel state information as well as service requirements (QoS) [11] are considered, a cross-layer approach to resource allocation [12] is necessary to exploit the properties of the physical layer, channel and source (video) coding [13].

978-1-4577-0274-7/11/$26.00 ©2011 IEEE

In this paper, we develop a novel framework for stable resource allocations in wireless networks. First, we present the system model and provide basic definitions on one-to-many matchings and stability. We show that three common channel aware schedulers, the maximum throughput scheduler, the score-based scheduler, and the proportional fair scheduler, do not result necessarily in a stable matching. Next, we describe the user and resource proposing deferred acceptance algorithm and analyze the properties of the algorithm and its outcome. A sufficient condition for uniqueness of the stable matching is derived and weak Pareto optimality on any unstable matching is shown. The performance of the stable matching is studied and it is shown that the resource proposing stable matching gives the maximum sum rate stable matching. Finally, an application is an ad-hoc network without arbitrator and coordination is studied. The transmitter proposing deferred acceptance algorithm leads to a stable matching which is agreed on by all transmitter and receiver nodes.

2. SYSTEM MODEL AND BASIC RESULTS

We consider a general scenario with K users, N resources (undivisible) which can be exlusively allocated to any one user. The set of users is denoted by $\mathcal{K} = \{1, ..., K\}$ and the set of resources is denoted by $\mathcal{N} = \{1, ..., N\}$. User k can have up to q_k resources. The resource allocation problem is to match the users to the resources. This is a one-to-many matching problem. These types of problems have a long history, since marriages (typically with quota $q_k = 1$) and college admissions ($q_k > 1$) are important and popular examples [2]. Regarding the college-student terminology, we identify the students with resources and the colleges with users because one student can go only to one college as well as one resource is allocated to a single user.

In wireless communications, this scenario oulined above occurs in many different settings. It could correspond to the allocation of sub-carriers in an OFDM downlink transmission system. There the set of carriers is matched to a set of active users in one cell or sector. Another example from wireless communications are ad-hoc networks. There, a set of transmitter nodes want to transfer their information to a set of receiver nodes. In order to enable communication a matching of transmitter nodes to receiver nodes is found. Here, the distributed nature of the matching problem requires special attention.

2.1. Resource Allocation and Matching Market

Each user has preferences on the resources based on her local information. In wireless communications, the local information contains channel quality information and it is given in terms of SINR values. Denote the channel quality of user $k \in \mathcal{K}$ on resource $n \in \mathcal{N}$ as $\alpha_{k,n} \geq 0$. Thus each user has a preference relation $\succ_k$ over the subsets of resources. A resource $n \in \mathcal{N}$ is acceptable to user k if the SINR leads to a user utility larger than zero, i.e., $\phi(\alpha_{k,n}) > 0$. The mapping $\phi : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ maps the channel quality to a utility function taking the local context and information of the user into account. For Shannon capacity it is $\phi(x) = \log(1 + x)$ or for finite modulation and coding schemes it is usually a step function. This can lead to non-strict preferences.

Each resource has also preferences on the users based on local information. In wireless communications, the local information could contain channel quality, buffer state, or any context related information available for resource allocation. Each resource $n \in \mathcal{N}$ has a preference relation P_n over the set of users and being unused (n). A user $k \in \mathcal{K}$ is acceptable to resource $n \in \mathcal{N}$ if kP_nn.

A resource allocation problem is specified by the tuple

$$(\mathcal{N}, \mathcal{K}, \boldsymbol{P}_{\mathcal{N}}, \succ_{\mathcal{K}}, \boldsymbol{q}) \tag{1}$$

consisting of the set of resources $\mathcal{N}$, the set of users $\mathcal{K}$, the set of preference relations of the resources $\boldsymbol{P}_{\mathcal{N}} = \{P_n\}_{n \in \mathcal{N}}$, the set of preference relations of the users $\succ_{\mathcal{K}} = \{\succ_k\}_{k \in \mathcal{K}}$, and the quota q_k, $1 < k < K$ describing how many resources a user k can have at most. Usually, there is not a single utility function associated with one resource but rather there is an overall utility function defined to describe the interest of the system operator, e.g., the sum utility or proportional fair utility. In economics, these functions which map the state to a real value are called *social welfare functions* because they rank the resulting (social) operating point from lowest to highest [14].

Definition 1 *A matching μ is a function from the set $\mathcal{N} \cup \mathcal{K}$ into the set of unordered families of elements of $\mathcal{N} \cup \mathcal{K}$ such that:*

1. *$|\mu(n)| = 1$ for every resource $n \in \mathcal{N}$ and $\mu(n) = n$ if $\mu(n) \notin \mathcal{K}$;*
2. *$|\mu(k)| = q_k$ for every user $k \in \mathcal{K}$ and if the number of resources in $\mu(k)$, say r is less than q_k, then $\mu(k)$ contains $q_k - r$ copies of k;*
3. *$\mu(n) \in \mathcal{K}$ if and only if $n \in \mu(K)$.*

The notation $\mu(\cdot)$ has different meanings depending on the argument. If the argument is a user k, then $\mu(k)$ gives a set of matched resources. If the argument is a resource n, then $\mu(n)$ maps to the matched user. Usually, these matchings are represented graphically, i.e.,

$$\mu_1 : \begin{array}{ccc} k_1 & k_2 & (n_4) \\ n_1\ n_3\ k_1 & n_2 & n_4 \end{array} \tag{2}$$

represents a matching at which user k_1 with quota $q_{k_1} = 3$ is matched with two resources, n_1 and n_3, user k_2 with quota one is matched with one resource n_2, and user k_3 as well as resource n_4 is unmatched.

Given a matching μ, the user utilities can be easily computed for user $k \in \mathcal{K}$ as

$$u_k(\mu) = \sum_{n \in \mu(k)} \phi(\alpha_{k,n}) \tag{3}$$

and the system utility described above can be computed as

$$\begin{aligned} u_s(\mu) &= \psi(\phi_1(\alpha_{\mu(n_1),n_1}, \mu), \phi_2(\alpha_{\mu(n_2),n_2}, \mu), ..., \\ &\quad \phi_N(\alpha_{\mu(n_N),n_N}, \mu)) \end{aligned} \tag{4}$$

where $\psi : \mathbb{R}^N \to \mathbb{R}+$ maps from the individual resource qualities to an overall performance measure, e.g., the for the sum utility $\psi_s(\boldsymbol{x}) = \sum_{n=1}^N x_n$, for proportional fair utility $\psi_f(\boldsymbol{x}) = \prod_{n=1}^N x_n$, or for max-min fair utility $\psi_m(\boldsymbol{x}) = \min(x_1, ..., x_N)$. The per resource utility functions $\phi_n(\alpha_{k,n}, \mu)$ take into account local information of the resources as well as the matching μ itself. Note that we do not make further assumptions on the functions ϕ_n and ψ. This is in contrast to monotonic increasing and concave utility functions used e.g. in [15].

Another representation for the matching μ is by a binary $K \times N$ matrix $\boldsymbol{A}$ where a zero at position i, j means that user i and resource j are not matched whereas a one means that user i and resource j are matched. Then, the constraints imply that each row sum is either one or zero and the sum of column k is smaller than or equal to q_k. The matching matrix $\boldsymbol{A}_1$ for the example of the matching μ_1 from (2) is given by

$$\boldsymbol{A}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{5}$$

The advantage of the matching matrix representation is that the system utilities can be written compactly, e.g., the sum Shannon capacity achieved with matching μ_1 can be computed as

$$u_s(\mu_1) = || \log(1 + \boldsymbol{M} \odot \boldsymbol{A}_1)||_F^2, \tag{6}$$

where $\log$ is applied component-wise and the $K \times N$ matrix $\boldsymbol{M}$ contains the channel qualities $\boldsymbol{M}_{kn} = \alpha_{k,n}$ and $\odot$ is the Schur-product.

The following definitions on stability of a matching μ can be found in [2]. The matching μ is *blocked* by resource n and user k if resource n strictly prefers k to $\mu(n)$ and either (i) k strictly prefers n to some $n' \in \mu(k)$ or (ii) $|\mu(k)| < q_k$ and n is acceptable to k. A matching is *individually rational* if for each resource $n \in \mathcal{N}$ it holds $\mu(n) P_n n$ or $\mu(n) = n$ and for each user $k \in \mathcal{K}$ it holds (i) $|\mu(k)| \leq q_k$ and (ii) $n \succ_k n$ for every $n \in \mu(k)$. A matching is *stable* if it is individually rational and not blocked. A resource allocation mechanism is a systematic way of assigning resources to users. A *stable mechanism* is a mechanism that yields a stable matching for every resource allocation problem $(\mathcal{N}, \mathcal{K}, P_{\mathcal{N}}, \succ_{\mathcal{K}}, \boldsymbol{q})$.

This notion of stability differs from the notion of stability used in queuing systems. Here, stability implies that there is not a single pair of resource and user which prefer being matched to each other instead of matched to their current partner. The interest in being matched is purely based on the preference relations of the users and resources. At first sight there is no operational meaning of this notion of stability. Consider the scenario in which the resources do not belong to one entity (e.g. to one base station) but to different receivers, e.g., in an ad-hoc network, who can autarkic decide with whom they will be matched. Then an unstable matching would leave room for some pairs to improve their situation by bilateral cooperation.

2.2. Basic Characterization of Matchings in Wireless Communications

First, we describe three representative resource allocation algorithms used in wireless communications, namely the maximum throughput allocation, the proportional fair allocation, and the score-based scheduling. It turns out that none of them necessarily leads to a stable matching.

The score-based scheduler is proposed in [16]. The idea is that each user assigns a score to each resource and numbers the resources according to their scores. The scheduler then allocates the user with the best score to each resource. This scheduler is shown to have a good performance versus fairness tradeoff.

Proposition 2 *The score-based scheduling mechanism described above does not necessarily lead to a stable matching.*

Proof Since this proof is by contradiction, it also illustrates how the score-based scheduler works. Consider a scenario with two users $\{A, B\}$ and three resources $\{1, 2, 3\}$. Let the channel matrix M be given by $\boldsymbol{M} = \begin{pmatrix} 2 & 1 & 0 \\ 4 & 5 & 1 \end{pmatrix}^T$. The score matrix is then given by $\boldsymbol{S} = \begin{pmatrix} 0 & 1 & 3 \\ 1 & 0 & 2 \end{pmatrix}^T$ where 3 means that resource 3 is not acceptable to user 1. The resulting score based matching μ_s is given by $\mu_s(1) = A, \mu_s(2) = B, \mu_s(3) = B$ or $\mu_s(A) = \{1\}$ and $\mu_s(B) = \{2, 3\}$. This matching μ_s is blocked by resourse 1 and user B because resource 1 stricly prefers user B to $\mu(1) = A$ and user B stricly prefers resource 1 to $3 \in \mu_s(B)$. ■

The maximum sum-utility matching is defined as the solution to the following combinatorial programming problem:

$$u_s(\mu_r) = \max_{\mu} \sum_{n=1}^N \phi(\alpha_{\mu(k),n}) \quad \text{s.t. } \mu \text{ is a matching.} \tag{7}$$

The constraint that μ is a matching contains also that the user's quotas q_k must be fulfilled. For a large number of resources and users, this problem is difficult to solve.

Proposition 3 *The sum-utility maximal matching μ_r is not necessarily stable.*

Proof Consider the following counter example $\boldsymbol{M} = \begin{pmatrix} 5 & 4 & 3 \\ 4 & 2 & 0 \end{pmatrix}^T$. The maximum sum-utility is achieved by the matching $\mu_r(1) = A, \mu_r(2) = B, \mu_r(3) = A$. However, this matching is blocked by resource 2 and user A because resource 2 strictly prefers user A to $\mu(2) = B$ and user A strictly prefers resource 2 to $3 \in \mu(A)$. ∎

The proportional fair matching is defined as the solution to the following problem similar to (7):

$$u_p(\mu_p) = \max_{\mu} \prod_{n=1}^{N} \phi(\alpha_{\mu(k),n}) \quad \text{s.t. } \mu \text{ is a matching.} \qquad (8)$$

Again, this is a computational complex combinatorial problem which is difficult to solve for a large number of users and resources.

Proposition 4 *The proportional fair matching μ_p is not always stable.*

Proof Consider the same channel matrix as above, i.e., $\boldsymbol{M} = \begin{pmatrix} 5 & 4 & 3 \\ 4 & 2 & 0 \end{pmatrix}^T$. The solution to (8) for $\boldsymbol{M}$ is given by the matching $\mu_p(1) = B$, $\mu_p(2) = A$, and $\mu_p(3) = A$. This matching is blocked by resource 1 and user 1. ∎

In conclusion, the three counter examples indicate that the usual scheduling algorithms used in wireless communications do not lead to stable matchings. The interpretation is that in these instable cases, there is a resource-user pair which would like to change their matchings. If the resources are administrated non-centrally or they operate on their own will, this resource-user pair could simply destroy the outcome of the scheduling algorithm by swapping their allocated match with each other.

3. STABLE MATCHINGS AND THEIR PROPERTIES

3.1. Deferred Acceptance Algorithms

The question whether there exists always a stable matchings was first answered positive and constructively in [1] by describing the algorithm which computes one stable matching: Every resource allocation problem has at least one stable matching because the so-called deferred acceptance algorithm finds one of these stable matchings [1]. There are two variants available: the resource proposing and the user proposing algorithm. We describe both algorithms in the following pseudo code adapted to the resource allocation problem at hand. For a textual description, the interested reader is referred to [2]. In the algorithm, note the double meaning of $\mathcal{W}^t$. If the argument is a user k, it returns the set of resources rejected by this user. If the argument is a resource n, it returns the set of users who have rejected resource n.

Result: Find stable matching for resource allocation problem $(\mathcal{N}, \mathcal{K}, \boldsymbol{P}_{\mathcal{N}}, \succ_{\mathcal{K}}, \boldsymbol{q})$
Input: $\boldsymbol{P}_{\mathcal{N}}, \succ_{\mathcal{K}}$
Init: Initialize the ordered set of temporarily accepted resources $\mathcal{A}^t(k)$ for step t and the set of users who rejected resource n in step t, $\mathcal{W}^t(n)$;
Step $t = 1$: Resource $n \in \mathcal{N}$ applies for user $k \in \mathcal{K}$ with $k = \arg\max_{k' \in \mathcal{K}} \phi(\alpha_{k',n})$. Denote the resources who apply for user k as $n_1^k, ..., n_m^k$. User k keeps the first q_k best ranked resources, i.e., $\mathcal{A}^1(k) = \{n_1^k, ..., n_{q_k}^k\}$ with $n_l^k P_k n_{l+1}^k$ for $l = 1..., m-1$. Update the set of rejected resources $\mathcal{W}^1(k) = \{n_{q_k+1}^k, ..., n_m^k\}$ and the set of rejected users $\mathcal{W}^1(n) = \{k \in \mathcal{K} : n \in \mathcal{W}^1(k)\}$;
Step t: All resources not yet assigned $n \in \mathcal{N} \setminus \bigcup_{k \in \mathcal{K}} \mathcal{A}^{t-1}(k)$ apply for their next best user, i.e., $k \arg\max_{k' \in \mathcal{K} \setminus \mathcal{W}^{t-1}(n)} \phi(\alpha_{k',n})$. Denote the resources who apply for user k as $\tilde{n}_1^k, ..., \tilde{n}_{m'}^k$. User k keeps the first q_k best ranked resources from $\mathcal{S} = \{n_1^k, ..., n_m^k\} = \mathcal{A}^{t-1}(k) \cup \{\tilde{n}_1^k, ..., \tilde{n}_{m'}^k\}$ and updates $\mathcal{A}^t(k) = \{n_1^k, ..., n_{q_k}^k\}$ with ordered elements $n_l^k P_k n_{l+1}^k$ for $l = 1, ..., m-1$;
Output: Stable matching μ_{rp}

Algorithm 1: Resource Proposing Deferred Acceptance Algorithm

The algorithm terminates either when all resources are matched to users or every unmatched resource has been rejected by every acceptable user. Therefore, the algorithm terminates after a finite number of steps. The resource proposing deferred acceptance algorithms results in a stable matching.

In order to continue the examples from above, the resource proposing deferred acceptance algorithm leads for the matrix $\boldsymbol{M} = \begin{pmatrix} 2 & 1 & 0 \\ 4 & 5 & 1 \end{pmatrix}^T$ to the following matchings: $q = 1; \mu_{rp}(A) = 2, \mu_{rp}(B) = 1$, $q = 2; \mu_{rp}(A) = \emptyset, \mu_{rp}(B) = \{1, 2\}$, $q = 3; \mu_{rp}(A) = \emptyset, \mu_{rp}(B) = \{1, 2, 3\}$.

In contrast to the resource proposing acceptance algorithm the user proposing algorithm differs since one user k can have up to q_k resources allocated. Therefore, for each user k a number q_k of virtual users with identical preferences are created. This results in $K_C = \sum_{k=1}^{K} q_k$ users and a one-to-one matching problem $(\mathcal{N}, \mathcal{K}_c, \boldsymbol{P}_{\mathcal{N}}, \succ_{\mathcal{K}_c})$. Then, the user-proposing acceptance algorithm can be defined similarly as the resource proposing variant.

The example with $\boldsymbol{M}^T = \begin{pmatrix} 2 & 1 & 0 \\ 4 & 5 & 1 \end{pmatrix}$ gives for $q = 2$ a virtual matrix $\boldsymbol{M}' = \begin{pmatrix} 2 & 2 & 4 & 4 \\ 1 & 1 & 5 & 5 \\ 0 & 0 & 1 & 1 \end{pmatrix}$ and the user proposing deferred acceptance algorithm results in $\mu_{up}(A) = \emptyset$ and

$\mu_{up}(B) = \{1, 2\}$ which corresponds to the resource proposing stable mechanism. This does not happen by chance as shown below in Theorem 5.

In general, the outcomes of the two algorithms are not equal. In fact, there are multiple stable matchings possible and the two algorithms give the extreme points of the set of stable matchings. A characterization of the set of stable matchings can be found in [2, Section 3.1]. The user proposing algorithm gives the user optimal stable matching μ_U, i.e., this is the matching which is preferred by all users to any other stable matching. The resource proposing algorithm gives the resource optimal stable matching μ_R. All other matchings lie between these two, in the following sense $\mu_U \succ_R \mu \succ_R \mu_R$ and vice versa $\mu_U \prec_U \mu \prec_U \mu_R$.

As a final corollary, we observe that the resource optimal stable matching is not necessarily sum-utility optimal.

3.2. Uniqueness of Stable Matching

It depends on the user and resource preferences whether there are multiple stable matchings or not. Since the structure of our preferences and underlying utility functions does not fulfill the non crossing condition (NCC), we obtain multiple stable matchings in general [17].

We dicuss two cases. In the first case, there will be multiple stable matchings and in the second case, we provide a sufficient condition for the uniqueness of the stable matching. Both cases are motivated by two wireless communications scenarios.

First, we consider a interference scenario in which the local information of the users and the resources lead to multiple stable matchings: Let the users have access to their channel quality $\alpha_{k,n}$, e.g., by pilot based channel estimation. Their preferences are thus based purely on $\alpha_{k,n}$. The resources have additional context information, namely the complete SINR including transmit power constraint with spectral shaping mask constraints $p_{k,n}$ and interference level z_n. Their preferences are based on the SINR of user k on resource n given by

$$\text{SINR}_{k,n} = \frac{\alpha_{k,n} p_{k,n}}{\sigma^2 + z_n}. \tag{9}$$

For an anecdotal example, assume four users and four resources. Let $q_k = 1$ for all users, $\sigma^2 = 1$ and the channel gain matrix $\boldsymbol{M}$ be given below in (10). Let the interference vector $\boldsymbol{z} = [z_1, z_2, z_3, z_4] = [1, 1, 1, 1]$ and the transmit power spectral mask constraints given below in (10).

$$\boldsymbol{M} = \begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 4 & 1 & 2 \\ 2 & 1 & 4 & 3 \\ 1 & 2 & 3 & 4 \end{bmatrix}, \; \boldsymbol{P} = \begin{bmatrix} 1/4 & 2/3 & 3/2 & 4 \\ 2/3 & 1/4 & 4 & 3/2 \\ 3/2 & 4 & 1/4 & 2/3 \\ 4 & 3/2 & 2/3 & 1/4 \end{bmatrix}. \tag{10}$$

	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	μ_8	μ_9	μ_{10}
R1	1	2	1	2	3	2	3	4	3	4
R2	2	1	2	1	1	4	4	3	4	3
R3	3	3	4	3	4	1	1	1	2	2
R4	4	4	3	4	2	3	2	2	1	1

Table 1. Ten stable matchings for the resource allocation problem example.

The resulting SINR matrix is given by

$$\text{SINR} = \boldsymbol{P} \odot \boldsymbol{M} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}. \tag{11}$$

Note that this example leads exactly to the preferences in [2, Example 2.17] provided by Knuth. Therefore, there are *ten* stable matchings where the resources $1, 2, 3, 4$ are matched to the users according to the Table 1.

Next, let us consider the case where there are neither intercell interference nor spectral mask constraints. Then, both preference relations are based on the same channel gain matrix $\boldsymbol{M}$ containing $\alpha_{k,n}$. The utility of the user k on resource n is simply $\phi(\alpha_{k,n})$ and the same for the utility of resource n for user k. in this case, there is a unique stable matching, even though the (sufficient) NCC is not satisfied.

Theorem 5 *If the preferences of the users and resources are strict and depend on the same common coefficients $\alpha_{k,n}$, then there is only one stable matching μ^*.*

3.3. Pareto Optimality of Stable Resource Matchings

Definition 6 *A matching μ is weak Pareto Optimal (PO) if there is no other matching ν with $\boldsymbol{u}(\nu) \geq \boldsymbol{u}(\mu)$ where the inequality is component-wise and strict for one user.*

Remember that the utility of user k is given by

$$u_k(\mu) = \sum_{n \in \mu(k)} \underbrace{\phi(\alpha_{k,n})}_{\ell(k,n)} \tag{12}$$

where $\phi(\alpha_{k,n})$ describes the utility of user k on resource n.

In [2, Theorem 2.27] it is shown that in one-to-one stable matchings for the marriage market the man-optimal stable matching is weak PO for the men, i.e., there is no individually rational matching (stable or not) which is preferred by all men. Unfortunately, the strong PO does not hold, i.e., there are individual men in the man-optimal stable matching who prefer another matching.

Proposition 7 *Any stable resource-user matching μ is weak PO to an unstable matching ν.*

Proof Assume the contrary, i.e., that there is a unstable ν which is PO to a stable μ. There are two possibilities: either ν is unstable because of lack of individually rationality or because it is blocked.

The first case is easy. Assume that resource l is not individual rational, then the utility of resource l can be improved by removing user $\nu(l)$ and matching another user, say $j = \mu(l)$. This increases the utility of user j and $u_j(\nu) < u_j(\mu)$.

The second case is more difficult: Assume that the unstable ν is blocked by resource n and user k. This is so if resource n strictly prefers user k to $\nu(n)$ and either (i) k stricly prefers n to some $n' \in \nu(k)$ or (ii) $|\nu(k)| < q_k$ and n is acceptable to k. For case (i), we construct a stable matching μ by interchanging resource n and n' for user k, i.e.,

$$\mu(k) = (\nu(k) \setminus n') \cup n.$$

This leads to the new utility of user k

$$\begin{aligned} u_k(\mu) &= \sum_{m\in\mu(k)} \ell(k,m) = \ell(k,n) + \sum_{m\in\mu(k),n\neq m} \ell(k,m) \\ &> \ell(k,n') + \sum_{m\in\nu(k),m\neq n} \ell(k,m) \\ &= \sum_{m\in\nu(k)} \ell(k,m) = u_k(\nu). \end{aligned} \tag{13}$$

The inequality in (13) holds because user k strictly prefers resource n to n'. Since all other utilities are left unchanged this shows for case (i) that $\boldsymbol{u}(\mu) \geq \boldsymbol{u}(\nu)$. For case (ii), let $k' = \nu(n)$ with utility $\ell(k',n)$. This implies

$$u_k(\mu) = \sum_{m\in\nu(k)} \ell(k,m) + \ell(k,n) > \sum_{m\in\nu(k)} \ell(k,m) = u_k(\nu)$$

and

$$\begin{aligned} u_{k'}(\mu) &= \sum_{m\in\nu(k')} \ell(k',m) - \ell(k',n) \\ &< \sum_{m\in\nu(k')} \ell(k',m) = u_{k'}(\nu). \end{aligned}$$

From these two inequalities follow that neither $\boldsymbol{u}(\mu) \geq \boldsymbol{u}(\nu)$ nor $\boldsymbol{u}(\mu) \geq \boldsymbol{u}(\nu)$. This completes the proof. ■

4. PERFORMANCE OF RESOURCE-USER STABLE MATCHINGS

In this section, we apply the resource proposing and user proposing deferred acceptance algorithm in the wireless communications context. Since we assume a special structure of the underlying preference relations, these results cannot be applied to general markets.

Define the set of preliminary accepted resources in step t by user k by $\mathcal{A}^t(k)$. The set of all preliminary accepted resources in step t is thus $\mathcal{A}^t = \bigcup_{k\in\mathcal{K}} \mathcal{A}^t(k)$. Define the set of all users who were rejected by resource n in step t as $\mathcal{W}^t(n)$. The set of resources which apply for user k in step t is given by

$$\mathcal{V}^t(k) = \left\{ n \in \mathcal{N} \setminus \mathcal{A}^t : k = \arg \max_{k'\in\mathcal{K}\setminus\mathcal{W}^t(n)} \ell_{k',n} \right\}. \tag{14}$$

Each user k solves in step t the following optimization problem

$$\mathcal{A}^{t+1}(k) = \arg \max_{\mathcal{S}\subseteq\mathcal{A}^t(k)\cup\mathcal{V}^t(k),|\mathcal{S}|\leq q_k} \sum_{n\in\mathcal{S}} \ell_{k,n}. \tag{15}$$

From Proposition 3 follows that there are (strong) PO matchings that are not stable, e.g., the maximum sum utility matching. If we restrict the outcome of the resource allocation problem to the set of stable matchings, then an important question is how to find the sum utility maximal stable matching.

Proposition 8 *The resource-proposing deferred acceptance algorithm yields the stable matching with maximum sum-utility.*

Remark: The reason for this behavior lies in the preferences of users and resources. These are coupled by their respective utility functions. In general many-to-one matchings, this result does not hold. Note further, that this result does not follow from the weak PO because some resources may prefer another stable matching.

Proof In step 1 of the algorithm, the resources apply for their best users, i.e., resource n applies for $\arg\max_k \ell(k,n)$. The users decide on the set of resources they keep. User k keeps the set of maximum size q_k which gives highest sum-utility. Therefore, the sum-utility is maximized in the first step.

For step t of the algorithm, we can rewrite (15) for user k as

$$\mathcal{A}^{t+1}(k) = \arg \max_{\mathcal{S}\subseteq\mathcal{V}^t(k)\cup\mathcal{V}^{t-1}(k)\cup\ldots\cup\mathcal{V}^0(k),|\mathcal{S}|\leq q_k} \sum_{n\in\mathcal{S}} \ell_{k,n}, \tag{16}$$

because the proposals which were rejected in former steps $t-1, ..., 0$ will be also again rejected in step t. Therefore, the set is not changed if we add these sets to $\mathcal{V}^t(k)$. The sum utility after step $t+1$ is given by

$$u_s^t = \sum_{k=1}^{K} \sum_{n\in\mathcal{A}^{t+1}(k)} \ell_{k,n}.$$

If t is increased to some T for which $\mathcal{V}^T(k) = \emptyset$, it can be observed that the set of all resources that have applied for user k is given by

$$\mathcal{V}^\infty(k) = \mathcal{V}^0(k) \cup \mathcal{V}^1(k) \cup ... \cup \mathcal{V}^{T-2}(k) \cup \mathcal{V}^{T-1}(k). \tag{17}$$

User k chooses the best set of maximal q_k resources from $\mathcal{V}^{\infty}(k)$, i.e., define

$$\mathcal{A}^{\infty}(k) = \arg \max_{\mathcal{S} \subseteq \mathcal{V}^{\infty}(k), |\mathcal{S}| \leq q_k} \sum_{n \in \mathcal{S}} \ell_{k,n}.$$

This clearly maximizes the sum-utility. Note, that by construction, the best q_k resources in $\mathcal{V}^{\infty}(k)$ contain disjunct resource sets, i.e.,

$$\mathcal{A}^{\infty}(1) \cap \mathcal{A}^{\infty}(2) \cap ... \cap \mathcal{A}^{\infty}(K) = \emptyset. \tag{18}$$

To finish the proof, the question remains whether it is possible to construct these sets $\mathcal{V}^{\infty}(1), ..., \mathcal{V}^{\infty}(K)$ better in terms of sum-utility. Assume, that we can get a better sum-utility by swapping one resource n' from some $\mathcal{A}^{\infty}(k)$ to $\mathcal{A}^{\infty}(k')$. This implies that at some time t, the resource n' has not applied for k but for k'. But the resources have applied for the best user in each step, thus $\ell_{k,n'} \geq \ell_{k',n'}$. Therefore, this swap gives lower sum-utility which is a contradiction and concludes the proof. ■

4.1. Performance Improvement with Non-Strict Preferences

Finite modulation and coding schemes lead to weak user preferences. Consider the scenario in which one user has three acceptable resources each which identical achievable rates (e.g. 4-QAM and 1/3 codec). The question is how to assign the 'best' resource to the user. One naive approach is to randomly choose between equal prioritized resources. However, this could lead to very inefficient matchings. There might even be matchings resulting which are not Pareto efficient for the resources. Using the framework of [18], it is possible to improve the utility of all resources by a rotation algorithm.

In order to improve the matching, [18] proposes a *stable improvement cycle algorithm* based on the construction of a directed graph Γ where for each pair of users k_1 and k_2 there is an edge $k_1 \to k_2$ if and only if there is a resource i which is matched to k_1 under μ and i belongs to the set of highest priority resources of user k_2.

5. APPLICATION

Let us consider a wireless ad-hoc networks in which there are a total of K nodes which deconstruct into two disjunct sets, a set of transmitters nodes $\mathcal{T}$ and a set of receiver nodes $\mathcal{R} = \{1, ..., K\} \setminus \mathcal{T}$. The channel conditions depend only on the location (distance) and are given by $\alpha_{kl}(\boldsymbol{S}(t))$ with node location vector $\boldsymbol{S}(t) = [S_1(t), ..., S_K(t)]$ and $k \in \mathcal{T}$ and $l \in \mathcal{R}$.

One suboptimal scheduling policy is to randomly select transmit/receive node pairs. Another approach is to allow nodes to send transmission requests and allow an arbitrator to determine which requests are granted [12]. Several such

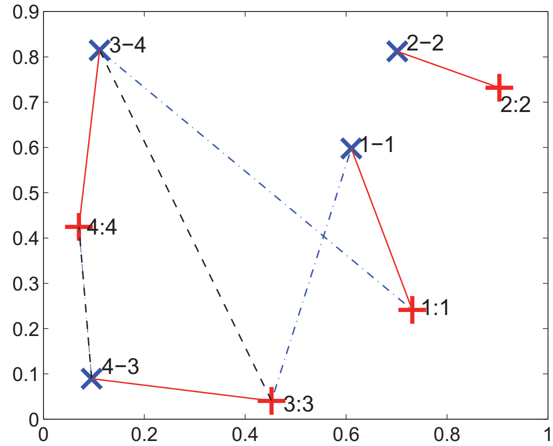

Fig. 1. Example 4×4 ad-hoc network with random (blue, dash-dotted), stable (black, dashed), and best unstable (red, solid) matching.

rounds of arbitration can take place to improve the scheduling decision. If such an arbitrator is not available, the transmitter- or receiver proposing deferred acceptance algorithm is a good candidate to find a stable matching. The transmitters announce their transmission requests to the receivers which in turn reject or accept temporarily the offers. The resulting matching will be stable such that all transmitters and receivers voluntarily accept the outcome. However, there is a price to be paid compared to the best unstable matching.

Consider the following example with eight nodes, four transmitting and four receiving, all dropped uniformly and independently over a 1x1 planar area. The channel gains α_{kl} are computed as the inverse Euclidean distance between transmit and receive node. Figure 1 shows the node locations of the transmitters (cross) and the receivers (plus) as well as the random, stable, and best unstable matching.

The channel gain matrix for the example in Figure 1 is given by

$$\boldsymbol{M}^T = \begin{pmatrix} 7.0544 & 3.0593 & 1.4032 & 2.3427 \\ 9.6052 & 21.1034 & 1.5761 & 0.9384 \\ 2.9795 & 1.5203 & 1.3982 & 7.7266 \\ 3.1126 & 1.8238 & 6.5012 & 8.8563 \end{pmatrix} \tag{19}$$

The gain matrix is related to the channel gains $[\boldsymbol{M}]_{kl} = \alpha_{kl}$ defined above. In Figure 1, the numbers at the transmitter nodes show the best unstable matching and the numbers at the receiver nodes show the stable matching. The resulting matchings are given below. The random matching is

$$\mu_{random}: \begin{matrix} 1 & 2 & 3 & 4 \\ 3 & 2 & 1 & 4 \end{matrix} \tag{20}$$

with resulting sum rate $R_{rand} = 5.76$ bits per channel use at

Tx1	2	1	4	3	Rx1	1	2	4	3
Tx2	2	1	4	3	Rx2	2	1	3	4
Tx3	4	2	1	3	Rx3	4	1	2	3
Tx4	4	3	1	2	Rx4	4	3	1	2

Table 2. Preference lists of transmitter 3 and 4 and receivers 3 and 4 for the channel gain matrix $\boldsymbol{M}$ in (19).

10 dB SNR. The stable matching is

$$\mu_{stable}: \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{array} \tag{21}$$

with resulting sum rate $R_{stab} = 6.07$ bits per channel use at 10 dB SNR. The best unstable matching is

$$\mu_{best}: \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 1 & 2 & 4 & 3 \end{array} \tag{22}$$

with resulting sum rate $R_{best} = 6.56$ bits per channel use at 10 dB SNR. The preference lists are collected in table 2.

Let us briefly repeat why μ_{best} is an unstable matching. This can be observed studying the preference lists of the transmitters and the receivers, in particular the preference list of transmitters three and four and receivers three and four in table 2. For μ_{best} the fourth transmit and fourth receive node are better off if they were matched and thus they block the matching.

6. REFERENCES

[1] D Gale and L Shapley, "College admissions and the stability of marriage," *American Mathematical Monthly*, vol. 69, pp. 9–15, 1962.

[2] A E Roth and M A O Sotomayor, *Two-Sided matching - A study in game-theoretic modeling and analysis*, Cambridge University Press, 1990.

[3] H. I. Ergin, "Efficient resource allocation on the basis of priorities," *Econometrica*, vol. 70, no. 6, pp. 2489–2497, Nov. 2002.

[4] F. Kojima and M. Manea, "Axioms for deferred acceptance," *Econometrica, will appear*, 2009.

[5] F. Kojima and P. A. Pathak, "Incentives and stability in large two-sided matching markets," *American Economic Review*, vol. 99, no. 3, pp. 608–627, June 2009.

[6] Z Han and K J R Liu, *Resource Allocation for Wireless Networks: Basics, Techniques, and Applications*, Cambridge University Press, 2008.

[7] K Seong, M Mohseni, and J M Cioffi, "Optimal resource allocation for OFDMA downlink systems," *Proc. IEEE ISIT*, 2006.

[8] Y. Yaffe, A. Leshem, and E. Zehavi, "Stable matching for channel access control in cognitive radio systems," in *Proc. IEEE CIP*, 2010.

[9] Y J Zhang and K B Letaief, "An efficient resource-allocation scheme for spatial multiuser access in mimo/ofdm systems," *IEEE Trans. on Communications*, vol. 53, pp. 107–116, 2005.

[10] N. Y. Ermolova and B. Makarevitch, "Practical approaches to adaptive resource allocation in OFDM systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2008, pp. Article ID: 160307, 2008.

[11] M. Ergen, S. Coleri, and P. Varaiya, "QoS aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless access systems," *IEEE Trans. on Broadcasting*, vol. 49, no. 4, pp. 362–370, Dec. 2003.

[12] L. Georgiadis, J. M. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in , Now Publishers*, vol. 1, pp. 1–144, 2006.

[13] Y. Su and M. van der Schaar, "Multiuser multimedia resource allocation over multicarrier wireless networks," *IEEE Trans. on Signal Processing*, vol. 56, no. 5, pp. 2102–2116, May 2008.

[14] K. J. Arrow, *Social Choice and Individual Values*, Yale University Press; 2nd edition, 1951.

[15] F. Kelly, "Charging and rate control for elastic traffic," *Europ. Trans. on Telecommunications*, vol. 8, pp. 33–37, 1997.

[16] T Bonald, "A score-based opportunistic scheduler for fading radio channels," in *Proc. European Wireless*, 2004.

[17] S Clark, "The uniqueness of stable matchings," *Contributions to Theoretical Economics*, vol. 6, pp. 28, 2006.

[18] A Erdil and H Ergin, "What's the matter with tie-breaking? improving efficiency in school choice," *American Economic Review*, vol. 98, pp. 669–689., 2008.