Chapter 2
Fundamentals of Matching Theory

Abstract Matching theory, in economics, is a mathematical framework that allows analyzing the formation of mutually beneficial relationships over time. Prior to the seminal work of Gale and Shapley on the stable marriage and college admission problems in 1962, many matching problems were solved by the "free for all market". The "free for all market" term refers to the period before matching theory was conceived as a discipline, as well as the way in which matching problems were dealt with during the period. Economists have identified several issues such as unraveling, congestion, and exploding offers in the "free for all market". Since then, with decades of efforts devoted to developing matching algorithms (i.e., there arises a trusted third party, which collects information, runs a matching algorithm, and broadcasts the matching results), these challenges were overcome. As a result, there has been a surge in the development of matching frameworks that have become widely used in many areas, such as the national resident matching program in the United States, the college admission in Hungary, the incompatible kidney exchange market, and the partnership formation in peer-to-peer (P2P) network, among others.

2.1 Preliminaries

In order to introduce the basic notions of matching theory, first, we start by using the classical matching model *stable marriage* (SM) [2] as an illustration example. In such a model, one considers two sets of matching agents: a set of men and a set of women who seek to be matched to each other. A *preference list* is defined for each agent as an ordered list based on the preferences over the other set of agents who he/she finds acceptable. A matching consists of (man, woman) pairs. As a fundamental requirement in any matching model, the *stability* concept refers to the case that, no *blocking pair* (BP) exists in a matching. A BP is defined as a (man, woman) pair, who both have the incentive to leave their current partners and form a new marriage relation with each other. A stable matching can be achieved by using the Gale-Shapley (GS) algorithm, which is widely deployed and has been customized to generate stable matchings in many other models. The GS algorithm is an iterative procedure, where players in one set make proposals to the other set,

© Springer International Publishing AG 2017

Z. Han et al., *Matching Theory for Wireless Networks*, Wireless Networks,
DOI 10.1007/978-3-319-56252-0_2

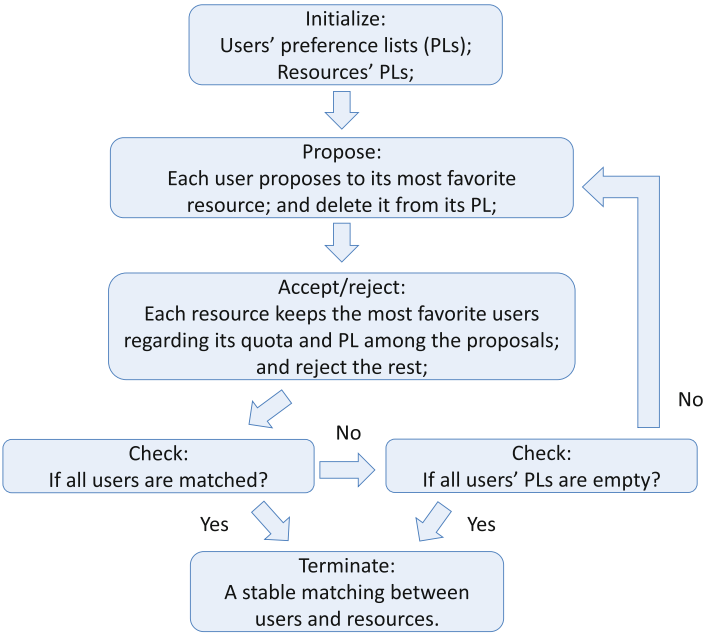


Fig. 2.1 The Gale-Shapley algorithm

whose players, in turn, decide to accept or reject these proposals, w.r.t. their quota. Players make their decisions based on their individual preferences. This process admits many distributed implementations which do not require the players to know each other's preferences [2]. The GS algorithm terminates when no further proposals are made. A flow chart illustrating the execution of GS is shown in Fig. 2.1.

With a general understanding of matching theory, the basic wireless resource management problem can be posed as a matching problem between resources and users. Depending on the scenario, the resources can be of different abstraction levels, representing base stations, time-frequency chunks, power, or others. Users can be devices, stations, or smartphone applications. Each user and resource has a quota that defines the maximum number of players with which it can be matched. The main goal of matching is to optimally match resources and users, given their individual, often different objectives and learned information. Each user (resource) builds a ranking of the resources (users) using the preference relation. The preference is evaluated based on local information, which can simply be defined in terms of an objective utility function that quantifies the QoS achieved by a certain resource-user matching. However, a preference is more generic than a utility function in that it can incorporate additional qualitative measures extracted from the information available to users and resources.

2.2 Conventional Matching Models

Matching problems can be classified in different ways. One typical classification [1] is shown in Fig. 2.2. The detailed explanations are provided as follows:

- Bipartite matching problems with two-sided preferences. Here the participating agents can be partitioned into two disjoint sets, and each member of one set ranks a subset of the members in the other set in order of preference. Example applications include assigning junior doctors to hospitals, pupils to schools and school leavers to universities.
- Bipartite matching problems with one-sided preferences. Again the participating agents can be partitioned into two disjoint sets, but this time only one set of players rank the subsets of the members in the other set in order of preferences. Example applications include campus housing allocation, DVD rental markets and assigning reviewers to conference papers.
- Non-bipartite matching problems with preferences. Here, all the participating agents form a single homogeneous set, and each agent ranks a subset of the others in order of preferences. Example applications include forming pairs of agents for chess tournaments, finding kidney exchanges involving incompatible (patient, donor) pairs and creating partnerships in P2P networks.

On the other hand, if we consider the capacity/quota allowed for each agent, we can have the following classification.

- One-to-one matching. It means each member of one set can be matched to at most one player from the opposite set. Examples include SM problem, forming roommate pairs, and so on. The one-to-one matching can be bipartite matchings with two-sided preferences (e.g., the SM problem), or bipartite matchings with one-sided preferences, or non-bipartite matchings (e.g., the stable roommate problem).

Bipartite matching with two-sided preferences	**Stable Marriage (SM); Hospital Resident (HR); Worker-Firm (WF);**
Bipartite matching with one-sided preference	Housing allocation (HA); Assigning paper to reviewers; DVD rental markets;
Non-Bipartite matching with preferences	Stable roommate (SR); Forming chess tournament pairs; Creating P2P partnerships.

Fig. 2.2 Conventional classification of matching theory

- Many-to-one matching. It means each agent of one set can be matched to more than one member from the opposite set up to the capacity, while agents from the opposite side can only be matched to one agent at most. Examples are like allocating residents to hospitals, assigning school leavers to universities. Still, the many-to-one matching can be bipartite matchings with two-sided preferences (e.g., the hospital resident allocation problem), or bipartite matchings with one-sided preferences (e.g., the student housing allocation problem), or non-bipartite matchings.
- Many-to-many matching. It means agents from both matching sets can be matched to more than one agent up to their capacities. Examples include creating partnerships in P2P networks and assigning workers to firms problem. The many-to-many matching can be bipartite matchings with two-sided preferences (e.g., the finding kidney exchange problem), or bipartite matchings with one-sided preferences, or non-bipartite matchings.

2.3 Wireless-Oriented Matching Models

To implement matching theory into the wireless resource allocations, we assume the wireless users and resources as the matching players. To capture the various wireless resource management features, we condense the rich matching literature into three new classes of problems, illustrated in Fig. 2.3, with the following properties:

1. *Class I*: *Canonical matching*: This constitutes the baseline class in which the preference of any resource (user) depends solely on the information available at this resource (user) and on the users (resources) to which it is seeking to match. This is useful to study the resource management within a single cell or for allocating orthogonal spectrum resources. Some of the applications discussed in the book belong to this category, for example, Chaps. 3 and 4.
2. *Class II*: *Matching with externalities*: This class allows finding desirable matchings when the problem exhibits "externalities", which can be translated into the interdependencies between the players' preferences. For example, in a small cell network, whenever a user is associated with a resource, the preference of other users will change since this allocated resource can create interference to other users using the same frequency. Thus, the preference of any player depends not only on the information available at this player, but also on the entire matching. We distinguish between two types of externalities: conventional externalities and peer effects. In the former, the dependence of the preferences refers to the performance change due to the interference between the matched (user, resource) pair. In the latter, the preference of a user on a resource will depend on the identity and number of other users that are matched to the same resource. Such peer effects are abundant in the wireless environment. We have also discussed how to address the external effect in Chap. 7.

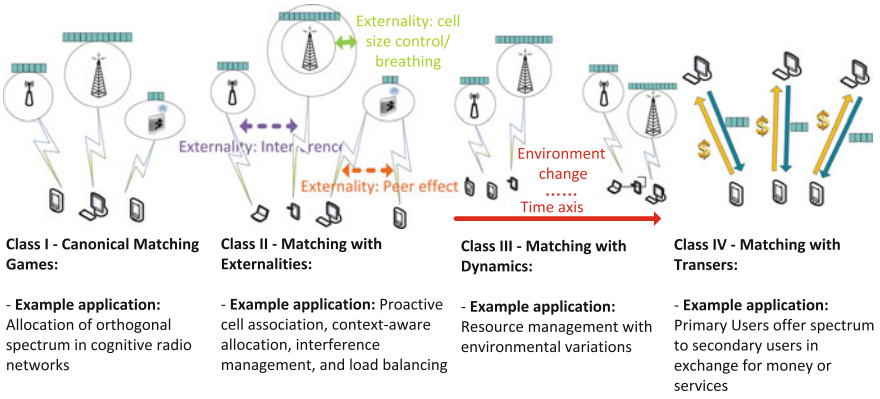


Fig. 2.3 Wireless-oriented classification of matching theory

3. *Class III*: *Matching with dynamics*: The third class, matching with dynamics, is suitable for scenarios in which one must adapt the matching processes to dynamics of the environment such as fast fading, mobility, or time-varying traffic. Here, at each time, the preferences of the players might change and, thus, the time dimension must be considered for in the matching solution. However, for a given time, the matching problem can either belong to class I or class II. The dynamic issue is discussed in the future works.
4. *Class IV*: *Matching with transfers*: The forth class, matching with transfers, there exists transactions between the matched agents. This transfer can be money, credit, service, resource and so on. For example, in the spectrum trading between the primary users (PUs) and secondary users (SUs), PUs offer spectrum resources to the SUs in exchanged for monetary gains or relay services. This type of matching problem is discussed in Chap. 8.
   The dynamic issue is also discussed in this book, such as in Chap. 5.

For class I, the preferences of one player set simply depend on the other player set. However, for class II, the preferences will now depend not only on the matched player set, but also on the entire matching, due to externalities. For class III, one can introduce a time-dependent state variable in the matching to achieve dynamic stability. Subsequently, the preferences will be time and state dependent, if the problem has both dynamics and externalities. The transition between states depends on the application being studied. For example, if the state represents the activity pattern of a licensed user, the transition would follow a classical Markov model. In contrast, if the state represents a dynamically varying fast fading channel, one can use differential equations to represent the state transition.

2.4 Stability and Information Exchange in Matching Solutions

The basic wireless resource management problem can be modeled as a matching game between resources and users. Depending on the scenarios, the resources can be of different abstraction levels, representing base stations, time-frequency chunks, power, or others. Users can be devices, stations, or smartphone applications. Each user and resource has a quota that defines the maximum number of players with which it can be matched. The main goal of matching is to optimally match resources with users, given their individual and different objectives and also their self-learned information. Each user (resource) builds a ranking of the resources (users) using a preference relation. In its basic form, a preference can simply be defined in terms of a utility function that quantifies the QoS requirements achieved by a certain matching. However, a preference is more generic than a utility function in that it can incorporate additional qualitative measures extracted from the information available to users and resources. In the wireless resource management, the matching stability implies the robustness to deviations that can benefit both the resource owners and the users. In fact, an unstable matching can for example lead to undesirable cases in which a base station can swap its least preferred user with another base station since this swap is beneficial to both the resource and the user. Having such network-wide deviations ultimately leads to unstable network operations. This concept is very useful in matching problems and is broadly applicable to all classes. In the subsequent chapters, within the context of each studied application, we will mathematically formalize the concept of stability and link it directly to the wireless network problem at hand.

Beyond stability, it is important to shed some light on the information exchange needed in general matching scenarios. In essence, the information exchange during the matching is implemented in a semi-distributed way, meaning some of the operations are made based on the players' locally collected information, while some other decisions may require the global information from the centralized agent (e.g., eNBs). Before the execution of any matching algorithm, the first step is to set up the preference lists for all players. The preference list is set up through the local information collection by each player. The collected/exchanged information can be channel state information (CSI), player location, or any other information that interests the player. After the information collection, players will rank the other type of players, according to their preferences, into descending/ascending orders. Thus, the preference list set-up is implemented in a distributed manner, without requiring any centralized agent. To explain the information exchange during the actual matching algorithm implementations, we need to first clarify the major operations that involve message exchange during the matching. Take the GS algorithm for example, most operations taken by the players, are the proposing, accepting and rejecting operations. To realize these operations, players need to maintain their preference lists and the temporary matching matrix. The proposing information is sent from one type of player to the other type through the communication signal with specific overhead

that indicates the proposing operation. Those users, who have received the proposal signals, will decide who to keep and who to reject according to the preference lists and capacity requirements, and also update their temporary matching matrices. The rejecting/accepting operations are realized by sending the communication signals with rejecting/accepting overheads to the rejected players. The players, who received such signals will update their matching matrices and then decide whether to start new proposals or not. Thus, GS algorithm can be implemented in a fully distributed way. However, there are still some operations that may require the assistance from the centralized agents in some other matching algorithms. For example, in the inter-channel cooperation (ICC) algorithm, introduced in Chap. 7, each iteration involves the detection of a BP. Such detection requires the computation of the centralized agents (e.g., eNBs), who have access to all players' preference lists. Once a BP is detected, the centralized agent will inform the involved players in this BP of the BP status through communication signals. Then the informed BP players will take the divorce and remarry actions with the related users indicated in the communication signals. The divorce and remarry actions are realized by the rejecting and accepting operations. To summarize, most operations in a matching algorithm can be implemented in a distributed way, while a small part of the operations require the assistance from the centralized agents, and thus making matching theory suitable to develop semi-distributed solutions.

Given these various challenges and opportunities of matching theory, in the next few chapters, we will explain, in detail, how one can leverage the analytical tools underlying matching theory to address a plethora of applications that address the challenges outlined in the previous chapter. In essence, for each application, we will provide a preliminary system model, then motivate how and why matching theory can be applied. Then, once a proper matching game is formulated, we will analyze the properties of this game within the context of the wireless application studied. In this regard, we will draw immediate connections between matching-theoretic concepts and wireless networking solutions. Then, we conclude each chapter by a set of simulation results that provide an insight on the benefits and limitations of matching theory, when applied in state-of-the-art wireless environments.

References

1. D.F. Manlove, *Algorithmics of Matching Under Preferences* (World Scientific, New Jersey, 2013)
2. D. Gale, L.S. Shapley, College admissions and the stability of marriage. Am. Math. Mon. **69**(1), 9–15 (1962)

http://www.springer.com/978-3-319-56251-3

Matching Theory for Wireless Networks

Han, Z.; Gu, Y.; Saad, W.

2017, VII, 116 p. 44 illus., Hardcover
ISBN: 978-3-319-56251-3