The Knowledge Engineering Review, Vol. 24:2, 111–135.
doi:10.1017/S0269888909000204 Printed in the United Kingdom

Negotiation criteria for multiagent resource allocation

KARTHIK IYER and MICHAEL N. HUHNS

Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA;
e-mail: iyerk0@gmail.com, huhns@engr.sc.edu

Abstract

Negotiation in a multiagent system is a topic of active interest for enabling the allocation of scarce resources among autonomous agents. This paper presents a discussion of the research on negotiation *criteria*, which puts in context the contributions to resource allocation from the fields of economics, mathematics, and multiagent systems. We group the criteria based on how they relate to each other as well as their historical origin. In addition, we present three new criteria: verifiability, dimensionality, and topology. The criteria are organized into five categories. The allocation category contains criteria concerning fairness and envy-freeness with respect to how resources are allocated to agents. The protocol category covers criteria for stability, strategy-proofness, and communication costs. The procedure category includes criteria about the complexity of allocation procedures. The resource category has criteria for the properties that various resources can take and how they affect allocation. The paper concludes with a discussion of the criteria for agent utility functions. The overall objectives of this paper are (1) to create a starting point for protocol engineering by future negotiation designers and (2) to enumerate the criteria and their measures that enable negotiation and allocation mechanisms to be compared objectively.

1 Introduction

The design of negotiation protocols is an area of active research in the field of multiagent systems (MAS). As software agents grow in intelligence and complexity, they are expected to perform higher-level tasks on behalf of their owner. In order to accomplish this, they will need to communicate with other agents on the Internet or local networks in order to exchange goods, services, and information. Thus, there has been research on how to design protocols and procedures in order to facilitate such communication. The focus of this paper is less on communication languages, such as the FIPA Agent Communication Language (2002), used for interaction protocols and more on various 'nice' properties the protocols should fulfill. Among these properties, MAS designers typically study the stability, strategy-proofness, and simplicity that a good protocol must fulfill. As part of our research on analyzing and synthesizing protocols and procedures for multiagent resource allocation, we realized that literature exists in mathematics and economics where researchers have looked at very similar problems for at least 50 years. MAS researchers can potentially benefit from this literature. Unfortunately, the various criteria that have been suggested are scattered and unorganized. The criteria have been synthesized for the most part, independently, in these fields. Our objective is to identify these criteria, assess their utility and independence, and provide an organization for them. In addition, several criteria that were implicit in the literature have been identified and made explicit in this paper. We hope that this document will serve as a source of reference for analyzing current and future protocols in a more systematic fashion.

1.1 Historical research

Our literature survey shows the different approaches taken by researchers from various fields. Researchers in MAS tend to focus on how to design mechanisms and protocols that discourage manipulation by agents and enable stable interactions. Economists, on the other hand, have focused on finding mechanisms that allow envy-free allocations to exist. Their focus is less on procedures that enable us to find such an allocation, and their arguments are more existential than constructive in nature. The third group consists of researchers from mathematics who have contributed to expositing resource allocation issues, as well as proposing constructive solutions to attain such allocations.

The mathematical literature discussing these problems has existed since, at least, as early as 1948, when Steinhaus (1948) explicitly discussed the problem of dividing a cake among n agents. The paradigm for discussing negotiation protocols/allocation procedures is different depending on the background of the researcher involved. Mathematicians primarily have modeled the resource as a cake or a pizza that is to be divided among various people with differing evaluations over various portions of the (edible) resource (the 'cake-cutting' problem). Based on the procedures that mathematicians have suggested, it is easy to see that two properties have been assumed to be true for the resource being divided: it is implicitly assumed that the resources are *infinitely divisible* and *recombinable*. Indivisible resources, such as a large estate being divided among heirs, have been looked into as well, but the solutions involve selling off a house (to one of the heirs or to an external party) to convert it to 'liquid' money, which is then used as a divisible (and recombinable) resource to satisfy the contending parties (Refer to Section 5.1 for further discussion of the notion of divisibility and recombinability). Fairness (where an agent in a group believes it has received at least an equal share of the resource), envy-freeness (where an agent believes its portion is as good as or better than everyone else's), and efficiency of allocation procedures, are of major concern. Boundedness (i.e., complexity of the allocation procedure) and running times are sometimes discussed, but they are not the major concern for mathematicians.

Economists on the other hand have focused mainly on existence conditions for envy-free allocations. Their treatment is set-theoretic and they put forth axioms and properties of the domain for existence of envy-free solutions. Since envy-freeness is generally difficult to achieve, there has been considerable interest in putting forth alternative criteria that keep the essence of envy-freeness but relax its constraints so that feasible solutions exist. An example of such an alternative is egalitarianism, which is explained later in this paper.

Researchers in MAS have analyzed negotiation protocols that allow for honest and fair allocations, but the emphasis has been on protocol properties, such as anonymity, strategy-proofness, and stability. Efficiency can be characterized in terms of maximization of social welfare. If we assume that the agents are autonomous and rational, no agent will prefer an allocation where it is worse off to make others better off, even if social welfare is maximized in the process. Pareto optimality, as an efficiency criterion, captures this dynamic in agent interactions. This can be visualized by a simple example: consider 100 apples that need to be divided among two agents. Agents 1 and 2 value each apple at \$1 and \$2, respectively. The solution that maximizes social welfare is one where agent 2 gets all the apples while agent 1 gets none (sw = \$200). However, every solution that allots x apples ($x \leqslant 100$) to agent 1 and $100 - x$ apples to agent 2 is Pareto optimal. Thus, Pareto optimality tends to maintain some justice (by maintaining *status quo*), even if the allocation is globally inefficient.

It is pertinent to point out here about earlier efforts (Rosenschein & Zlotkin, 1994; Weiss, 2000: Ch. 5) to collate negotiation criteria. Lomuscio *et al.* (2003) explore various issues related to negotiation, such as:

- The cardinality of negotiation;
- The required characteristics of individual agents;
- Characteristics of the environment;
- Event parameters;

- Information parameters;
- Allocation parameters.

Our work reflects the latest updates in multiagent negotiation that have occurred subsequently.

Another excellent exploration of this topic is by Chevaleyre *et al.* (2006), who discuss numerous criteria relevant to multiagent negotiation. We refer extensively to their work throughout this manuscript. Our work extends the efforts of Chevaleyre *et al.* in several ways:

1. We update the list of criteria discussed by Chevaleyre *et al.* For example, our manuscript considers criteria such as dimensionality, topology, verifiability, and infinitely divisible and recombinable resources.
2. We discuss in detail the criteria for protocols (Section 3).
3. Many criteria assumed implicitly in the literature have been made explicit in this paper. Examples include anonymity and privacy, mediated vs. arbitrated, and one-shot vs. iterative.
4. We discuss in detail the contributions by mathematicians to this field. Their work is mainly on the topic of fair division and is presented in Section 2 (Criteria for allocations).

1.2 Protocol engineering

A formal framework for the design of protocols and procedures is needed. MAS researchers and mathematicians have proposed solutions for very specific problems. These solutions, though ingenious in most cases, often cannot be generalized to solve a broader class of problems. Also, due to the scattered nature of the literature, analysis of particular protocols/procedures has not been systematic. What we envision is that protocol design, development, and testing should be conducted in a more thorough manner rather than done just to solve a specific problem. That is, a framework for *protocol engineering* should be established, which will serve two useful purposes:

1. A complete analysis of existing protocols/procedures can enable their extension to more general problem domains.
2. A systematic methodology can be provided for designing new protocols based on the criteria that need to be achieved.

Brams and Taylor (1996) have discussed the possibility of constructing allocation procedures that fulfill as many of the interesting criteria as possible. For example, can an allocation procedure be constructed that is envy-free, efficient, equitable, and truthful? So far, there seems to be no procedure that is known to fulfill these conditions. Thus, there exists a possibility that, mathematically, one may be able to prove no such procedure can exist. Different applications lead to different prioritizations of the criteria for negotiations. Therefore, before the process of designing a protocol, designers must first list the important criteria that the protocol must fulfill and then create one that satisfies the most important ones first. Consider, for example, the problem of selling perishable goods, such as fruits or flowers. The important criteria that should be fulfilled by a protocol devised for this problem are (from highest to lowest priority): time, truthful bidding, envy-freeness, and efficiency. Thus, creating an auction protocol that (1) terminates quickly (time), (2) is satisfactory to all agents (envy-free), (3) induces agents to be truthful but (4) is not efficient, might still constitute a reasonable tradeoff. Systematic analysis of requirements thus helps in choosing the right protocol. The protocol design process was first outlined by Rosenschein and Zlotkin (1994) and is visualized in Figure 1. In the following sections we present the various criteria that have been discussed in the literature. They are partitioned into the following five fundamental categories:

- Criteria for allocations;
- Criteria for protocols;
- Criteria for procedures;
- Criteria for resources;
- Criteria for utility functions.

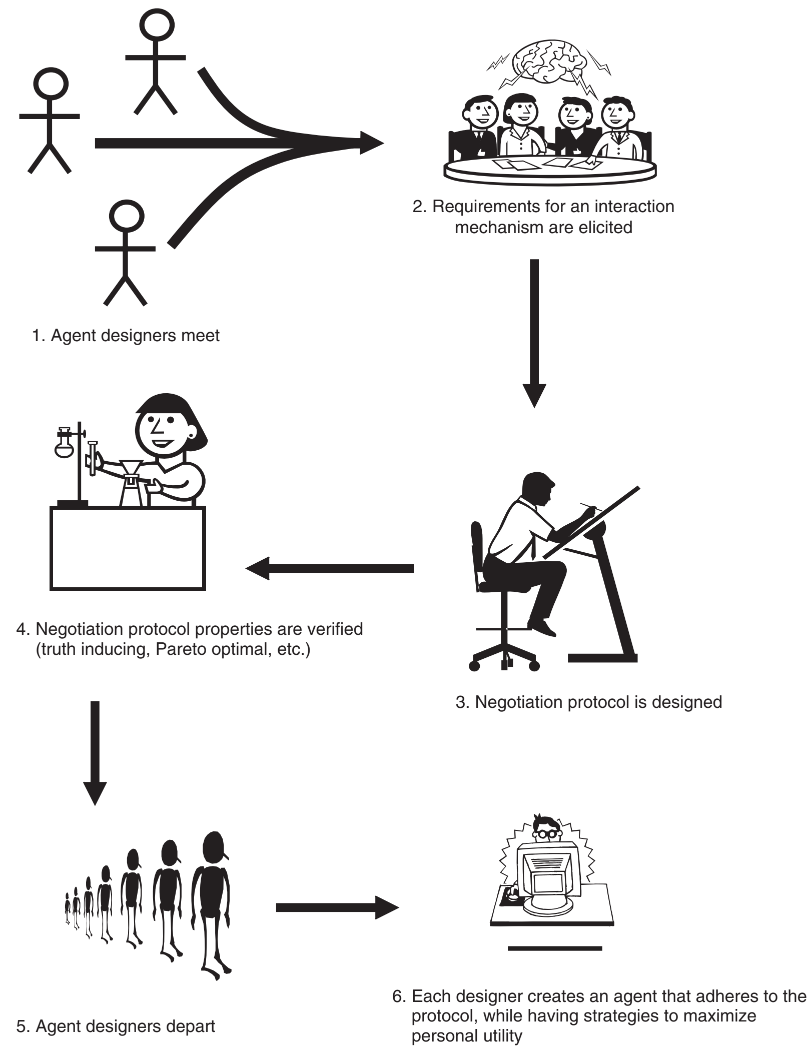

Figure 1 Engineering a protocol

2 Criteria for allocations

For a resource that will be partitioned into portions and allotted to agents, we need to determine if the agents are satisfied with the portions they receive. The following criteria will be used for this purpose:

2.1 Fairness

This is perhaps the oldest criterion mentioned in the research of fair division, and most commonly used across all disciplines. However, the concept behind its usage is imprecise. In some settings, fairness is mentioned as a diffuse concept that can be achieved by fulfilling various criteria, such as proportionality, envy-freeness, and equitability. Fairness has thus been used flexibly and we aim to clarify and fix the semantics of the term as follows: an allocation is deemed to be *fair* if every agent out of a group of size n thinks it received at least $1/n$ of the entire resource allocated. This definition has also commonly been termed as 'proportionality' (Brams & Taylor, 1996). We believe that fairness is a better term for the definition, since the many subclasses of fair (approximate fair, strong fair, etc.) will be clearer if fairness is used as the label for the overall class of such criteria.

This criterion is fulfilled by almost all protocols (or procedures) that are mentioned in the mathematical literature. The fairness criterion has been further refined in the literature and

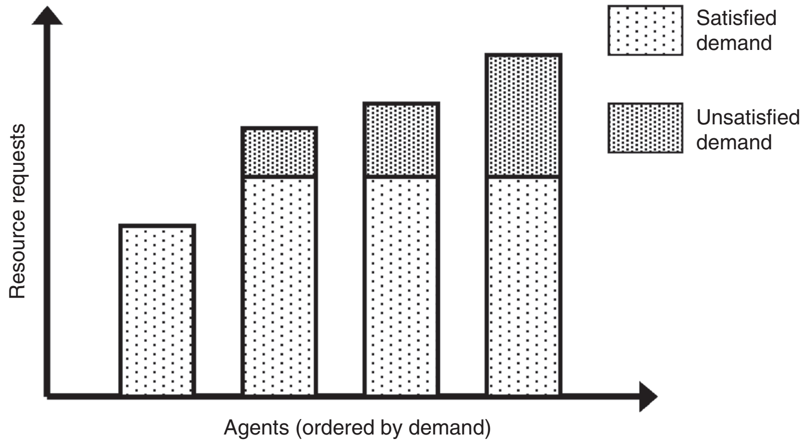

Figure 2 How max–min fairness works

sub-criteria that qualify fairness have been suggested. The following sub-criteria, in increasing order of strictness, have been mentioned:

- *Approximate fair* (Krumke *et al*., 2002): This criterion is invoked in cases where getting even simple fair allocations proves to be too difficult. Here each agent is assumed to have a tolerance ε, below which it considers the value of a resource piece to be negligible. Thus, an allocation is *approximate fair* if each agent gets a piece that is *at most* ε *smaller than* $1/n$ of the total resource allotted.
- *Simple fair*: An allocation is *simple fair* if each agent gets *at least* $1/n$ of the total resource allotted. This is the commonly accepted definition for fairness.
- *Strong fair*: An allocation is *strong fair* if each agent gets *more than* $1/n$ of the total resource allotted.
- *Max–Min fair* (Bertsekas & Gallager, 1992): This is not directly related to the fairness criteria described so far. The criterion has been suggested to solve the problem of packet scheduling in communication networks. It consists of distributing a commonly contended resource among a number of competing agents. The basic idea is to satisfy as many agents as possible and then distribute any leftovers to the agents that have larger demands. Thus, the demands of the agents are first ordered from lowest to highest. All agents receive an amount equal to the one received by the agent with the smallest demand. The remaining resources are then split equally among the remaining agents. Figure 2 shows the intuition behind this criterion.

Thus, the purpose of max–min fairness is to maximize the minimum allocated to each agent whose demand has not been fully satisfied. Some salient differences with respect to earlier fairness criteria are that each agent is assumed to get the same utility over the same portion of the resource, and the utility functions are additive in nature.

2.2 Envy-freeness

Envy-freeness is a stricter condition than fairness. Envy-freeness implies fairness, but not vice versa. The set of envy-free procedures is much smaller than ones that are simply fair. Depending on the context, fairness might not be sufficient for getting satisfactory solutions. An allocation is *envy-free* if each agent thinks that its piece is *at least as large as* the largest piece in the allocation. Thus, all envy-free solutions are fair. The procedures that have been proposed, however, involve some sort of trade-off to achieve this criterion. The most common trade-off is efficiency. Most envy-free procedures involve one agent trimming off pieces created by other agents (to create ties). These trimmings are then inaccessible to all agents till the end of the procedure and are wasted during the allocation process. Other tradeoffs include increased running times (in terms of the number of cuts or comparisons) of the procedure as well as higher complexity of the rules governing agent strategies. Envy-free solutions are still appealing[1] because once an agent considers its piece to be the largest, it will have no further incentive to try to trade its pieces with others or manipulate the process in some other way (e.g., by lying). Thus, fairness may not be a sufficiently strong condition in many real-world cases. Similar to the case of fairness, envy-freeness too has been qualified in various ways to get reasonable solutions in different situations. We define the following sub-criteria roughly categorized in increasing order of strictness.

2.2.1 Alternatives to envy-freeness

In many domains, attaining envy-free allocations might be quite appealing, but it may be infeasible to do so due to various constraints on the system. Therefore, as a compromise, suggestions have been put forth that weaken the envy-free criterion suitably so that the spirit of envy-freeness is followed as much as possible while making the criterion flexible enough to allow a larger space of feasible solutions.

- *Non-domination* (Arnsperger, 1994): An allocation is considered to have an *'absence of domination'* if for each agent i and every other agent j, there exists at least one agent k ($k \neq i, j$) that prefers agent i's portion over agent j's. This means that not everyone in the society of agents thinks agent i got a smaller portion than agent j. Thus, agent i's piece is not 'dominated' by any other agent's piece.
- *Egalitarianism* (Pazner & Schmeidler, 1978): In some economies it has been shown that among all Pareto-optimal solutions none might be envy-free. Thus, there can be conflict between the criteria of envy-freeness and Pareto optimality. Economists therefore put forth egalitarianism as an equity criterion that was close to envy-freeness in spirit while being more flexible. An allocation is said to be *egalitarian-equivalent* if there exists a piece of the resource (the same for every agent) that is considered by each agent to be indifferent to the bundle the agent actually gets in the allocation. Consider a portion of the resource (let us call it piece x). If every agent finds that the value of the portion it gets is at least as large as piece x, the allocation is egalitarian-equivalent. All envy-free solutions are egalitarian-equivalent but the converse is not true.

2.2.2 Envy-minimizing criteria

In situations where no envy-free allocations can be found, researchers have suggested approximations that will do the best that can be achieved. Thus, minimizing envy in allocations is a natural extension of the objective of getting envy-free allocations when this objective cannot be achieved. A number of indices for measuring the envy of a given allocation have been proposed.

- *Minimal ordered-pair envy* (Arnsperger, 1994): This index indicates the number of ordered pairs of agents, (i, j) in which i envies j. Thus, if x is the allocation and x_i is the portion allocated to agent i, which has utility function u_i then:

  $$\varepsilon_x = \{(i.j)|u_i(x_i) < u_i(x_j)\}$$

  is the set of ordered pairs (i, j) where i is envious of j's portion in allocation x. According to the Feldman–Kirman procedure (Arnsperger, 1994), the objective of an allocation should be:

  $$\min_x Card\, \varepsilon_x$$

  that is, the number of ordered pairs that feel envy should be minimized.
- *Minimal intensity envy* (Arnsperger, 1994): The minimal ordered-pair envy shown above has no notion of the intensity of envy present. For example, if i envies j and k, then does i envy j as

[1] Agents can be designed with one of the following priorities ranked higher than the others:
1. Agents want as much of the resource as possible (individual rationality).
2. Agents want a portion of the resource whose value is greater than or equal to that of any other agent (envy-freeness).
3. Agents want portions whose value is greater than a certain threshold (egalitarianism).

Our statement holds if priority #2 overrides the other two.

'strongly' as k? There is no way to distinguish weak envy from strong envy. The main reason intensity levels are not measured is because it requires interpersonal comparisons of utility, which would make the measure meaningless. However, there is a way to measure intensity of envy without making such comparisons. For any allocation x and any pair of agents i and j, let $\varepsilon_{x,ij}$ be a scalar that satisfies:

$$u_i(x_i) = u_i(\varepsilon_{x,ij}(x_j))$$

Thus $\varepsilon_{x,ij}$ represents the fraction of j's portion that i thinks is of the same value as its own. After suitable algebraic manipulations, it has been suggested (Arnsperger, 1994) that the objective of the allocation should be to minimize ξ_x where:

$$\xi_x = \sum_{i,j} \lambda_{x,ij}$$

and

$$\lambda_{x,ij} = (1/\varepsilon_{x,ij}) - 1$$

Thus, the objective is to minimize the intensity of global envy across all agents.

- *Worst-off envy* (Fishburn & Sarin, 1994): This is based on Rawl's idea (Rawls, 1971) of making the worst-off person as well off as possible. Let $e_{ij}[x]$ have value 1 when i envies j and be zero otherwise. Then,

$$e_i[x] = \sum_{j \neq i} e_{ij}[x]$$

is the number of people that i envies in allocation x. The index for worst-off envy is then defined as:

$$m_{\mathrm{we}}[x] = \max_i e_i[x]$$

and the objective is to find the allocation that minimizes $m_{\mathrm{we}}[x]$.

- *Median envy* (Fishburn & Sarin, 1994): Median envy measures the number of agents that envy more than half the number of other agents based on portions allocated to them. Based on the terms defined above for worst-off envy, median envy is defined as:

$$m_{\mathrm{mde}}[x] = \left|\{i : e_i[x] > (n-1)/2\}\right|$$

- *Mean envy* (Fishburn & Sarin, 1994): This index measures the average number of others that an individual envies. It is defined as:

$$m_{\mathrm{me}}[x] = \sum_i e_i[x]/n$$

Since it is the average, it ignores the variability of envy present in various allocations. This index may be combined with the Pareto-optimality condition to account for rational preferences agents make. Mean envy is a more refined measure than median envy and can be used as a secondary criterion to choose among alternative allocations, all of which are median envy-free.

- *Approximate envy-free* (Brams & Taylor, 1996): Similar to approximate fair, the criterion of approximate envy-free is used to approach envy-freeness in cases where getting simple envy-free allocations is not feasible. Let ε be a small positive number, which is the smallest portion that is of value to any agent. Anything smaller than ε is considered to be negligible by the agent. We can call an allocation *approximate envy-free* if no agent thinks its portion *is smaller by* ε *than* the largest portion in the allocation. The notion of approximate envy-free is used to propose many discrete procedures to allocate resources that compromise on the exact envy-free criterion enough to allow them to have efficient bounds on the number of resource divisions, as well the time steps required to complete the procedure. In real-world problems, such approximations might be tolerable compared to the alternative of not having envy-free procedures at all. Once ε is made small enough, the solution found would be satisfactory for all parties involved.

2.2.3 Envy-free criteria

We begin by defining some terms that will be used in this section. Let $u_i(x_i)$ denote the utility agent i derives from the portion of resource x allocated to it. Agent j ($\neq i$) denotes some other agent against whom we compare agent i's utility. We have the following versions of envy-freeness in order of increasing strictness.

- *Simple envy-free*: This is the commonly understood notion of envy-freeness. An allocation is *envy-free* if every agent gets a portion of the resource that is *at least as large as* any other agent's portion.
$$u_i(x_i) \geqslant u_i(x_j)$$

- *Strong envy-free* (Robertson & Webb, 1998): This is a stricter condition than simple envy-free. An allocation is *strong envy-free* if every agent gets a portion of the resource that is *larger than* any other agent's portion. Here the agent is not only envy-free; it actually believes that other agents are envious of its portion.
$$u_i(x_i) > u_i(x_j)$$

- *Super envy-free* (Robertson & Webb, 1998): This is a stricter condition than strong envy-free. An allocation is *super envy-free* if every agent believes that every other agent has *at most* a fair share of the resource. Naturally, here every agent not only believes it got the largest share, but also believes all other agents got a fair share at best.
$$u_i(x_j) \leqslant 1/n$$

- *Strong super envy-free* (Robertson & Webb, 1998): This is a stricter condition than super envy-free. An allocation is *strong super envy-free* if every agent believes that every other agent has *less than* a fair share of the resource.
$$u_i(x_j) < 1/n$$

- *Opportunity envy-free* (Arnsperger, 1994): This is a suggested measure for fairness that comes from the domain of social justice. It tries to formalize the notion that for equality among individuals to be attained, rather than apportion the resources equally, all agents be given equal means to fulfill their goals. Thus, one agent will be envy-free of another if it gets the same means as everyone else to fulfill its goals. The opportunity set C_i for agent i is the set of allocations x_i that are feasible given its initial endowments. Let x^* be an allocation such that $x_i^* \in C_i$ for each i. An allocation is *opportunity envy-free* if for every pair of agents i, j,
$$u_i(x_i^*) > u_i(x_j) \quad \forall x_j \in C_j$$

  Thus, agent i does not envy the opportunity available to any other agent. Opportunity envy-free implies envy-freeness but the converse is not true. This is a stricter condition, therefore, than envy-freeness.

2.3 Exact

This can be taken as a special case where many criteria mentioned earlier intersect. An allocation is *exact* if every agent believes *everyone* has received *exactly* $1/n$ of the total resource allocated. The set of allocations that are exact is the subset of the intersection of the sets of allocations that are approximate fair, fair, egalitarian, approximate envy-free, envy-free, and super envy-free (refer to Figure 3). In the context of divisible resources, Dubins and Spanier (1961) have mentioned the existence of exact allocations given that each agent has a measure function over the entire resource. There are, however, no constructive methods known to find such allocations.

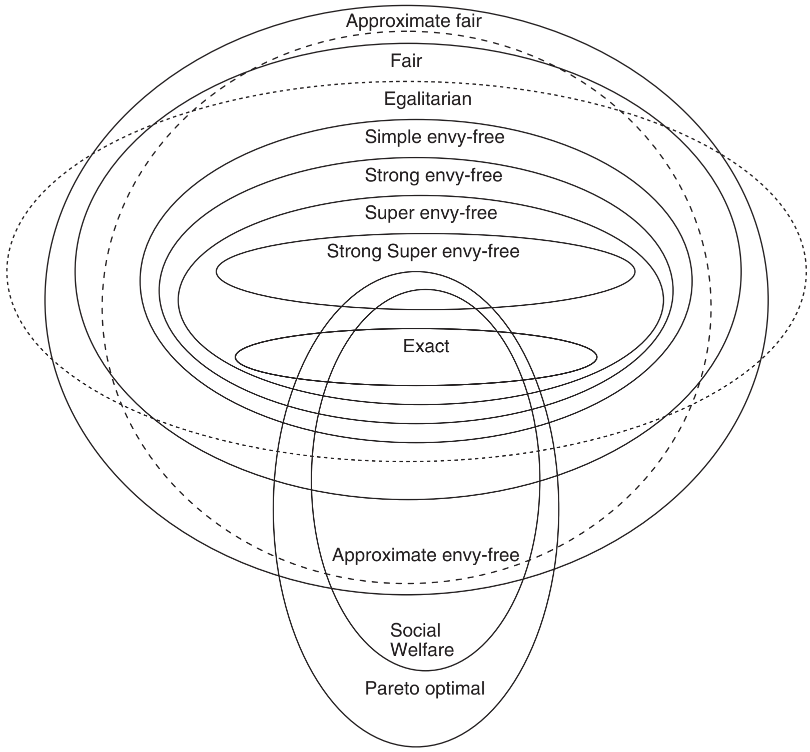

Figure 3 Relationships among the various allocation criteria

2.4 *Equitability*

This is a criterion put forth by Brams and Taylor (1996) for their Adjusted Winner (AW) procedure for allocating a resource. Since AW is a specific two-person procedure to divide heterogeneous resources and requires certain assumptions about the properties of the agent utility functions, equitability is also valid only under such conditions. The setup involves two agents each with different utilities for each item in a bundle of resources. Both agents are required to allot points to each item in the bundle, reflecting their relative values to each agent, such that the points sum to 100. Then the equitability condition states that the goods should be divided among the two agents in such way that both agents get goods worth an equal number of points; that is, not only should both agents think they got a larger share than the other; they should also think their share is larger by the same amount. This step requires that agents have utility functions that are additive and linear. Linearity means the agents' marginal utilities are constant. While envy-freeness and equitability both address the question of whether one agent believes it did at least as well as the other agent, the difference is that envy-freeness involves intra-agent comparison of utilities of different portions to a particular agent, whereas equitability involves a more controversial interpersonal comparison of utilities of different agents for a particular portion. Thus the objective is to give each agent portions such that their point totals are the same. However, it requires that comparisons between different utility functions be made, which might not hold up in real-world scenarios. This criterion can be applied lexicographically after efficiency to obtain better satisfaction for the participating agents.

2.5 *Efficiency*

Efficiency is deemed as an important criterion in many allocations right after they are checked for fairness and envy-freeness. Researchers in welfare economics sometimes prefer this criterion to be fulfilled over envy-freeness. However, MAS researchers may find efficiency to be secondary to strategy-proofness and stability, among others. Efficiency criteria indicate whether there has been wastage in the allocation procedure. Wastage mainly happens in envy-free procedures that trim the portions of the resource to create ties for the largest piece. The trimmed portions are generally wasted. Efficiency criteria, therefore, typically run afoul of envy-freeness. Some simple retrofitting of the procedures can reduce wastage. For example, the allocation procedure can be run iteratively, first on the whole resource, then on the leftovers, then on the leftovers of the leftovers, *ad infinitum*. Some procedures are however created from the beginning to be envy-free as well as efficient. While this is ideal, the rules tend to be quite complex and almost no procedures exist for allocating to general n number of agents. There are generally two schools of thought about how to approach efficiency. Those who adhere to a utilitarian theory have generally connected efficiency to the maximization of social welfare. Another approach is to view efficiency in terms of Pareto optimality. If agent utility functions are known, then social welfare can be computed and used. However, if one assumes that agent utilities are private information, then efficiency is more naturally measured in terms of Pareto optimality.

- *Social welfare*: The roots of social welfare are based on the theory of utilitarianism, which seeks quantitative maximization of good consequences for a population. Each agent is assumed to have a utility function that reflects the satisfaction it receives from obtaining a portion of the resource. The utility functions can be distinct for each agent and generally are not comparable to those of another agent. Thus, while utility theory can say if an agent is *happier* in receiving one portion over another, it cannot say if one agent would be *happier* than another agent in doing so. Social welfare refers to the overall welfare of society. It is defined as the summation of the utilities of all agents in the society. Ironically, social welfare tends to let individuals get monopoly over resources rather than benefiting society as a whole. Consider the simple example mentioned earlier. If 100 apples need to be distributed between agents 1 and 2, who value them at \$1 and \$2, respectively, then social welfare is maximized if all the apples are allotted to agent 2 leaving agent 1 with no share. Thus, social welfare functions may be at odds with equity and justice criteria. Since cardinal comparisons between utilities are involved, some researchers have preferred to measure efficiency using the Pareto-optimality measure.
- *Pareto optimality*: This is an alternative proposition for efficiency that does away with inter-utility comparisons. Rather than utilize cardinal measures of utility, it uses ordinal preferences to rank allocations. An allocation is *Pareto optimal* if there is no other allocation that makes every agent at least as well off and at least one agent strictly better off. This requirement fits well in game theory and MAS, where players (or agents) are assumed to be autonomous and rational. It is quite reasonable to assume that given an initial distribution of resources, no agent will agree to any alternative distribution where it has to sacrifice its welfare, even if it improves the condition of society as a whole. Pareto optimality thus tends to maintain the *status quo*. Depending on the perspective, however, this is seen as desirable or undesirable. Continuing on our previous example of dividing apples among agents, any solution which grants x apples to agent 1 and $100-x$ apples to agent 2 ($0 \leqslant x \leqslant 100$) is Pareto optimal. The social welfare solution of agent 2 getting all the apples is also a member of the set of Pareto-optimal solutions. Thus, social welfare solutions are a subset of Pareto-optimal solutions. As can be seen, Pareto optimality does not necessarily promote equity in distributions and monopolistic allocations are ranked as high as fair ones.

The discussion shows that efficiency is so far seen as a binary variable. If we interpret efficiency in terms of Pareto optimality, then an allocation is either Pareto optimal or is not. It is inefficient if it is not Pareto optimal, hence the question of whether an allocation is efficient can only have one of two answers: Yes or No, that is, efficiency is a binary variable in this case.

If we interpret efficiency in terms of social welfare, then we determine if an allocation maximizes social welfare or not. An allocation that does not maximize social welfare is inefficient. Again, the question of whether an allocation is efficient or not can be answered by a binary variable.

This definition is too constraining and is not able to judge procedures that can improve upon efficiency while being inefficient in the traditional sense. There is a clear need for creating a measure of efficiency that can rank procedures based on the amount of resource wasted.

Table 1 Valuations of allocations from the perspective of agent 1

Portion	1	2	3	Approximate fair?	Approximate envy-free?	Fair?
Allocation 1	$1/3 - \varepsilon$	1/3	$1/3 + \varepsilon$	✓	×	×
Allocation 2	$1/3 - \varepsilon/2$	1/3	$1/3 + \varepsilon/2$	✓	✓	×
Allocation 3	1/3	1/3	1/3	✓	✓	✓
Allocation 4	1/3	$1/3 - 2*\varepsilon$	$1/3 + 2*\varepsilon$	✓	×	✓

The allocations show how approximate fair and approximate envy-free relate to each other.

2.6 Relationship between various allocation criteria

The relation between the various allocation criteria is shown in Figure 3. The salient features of the figure are explained further. The set of approximate-fair allocations is the largest set that encompasses all the other criteria (except egalitarianism). It can be shown that the set of approximate envy-free allocations is a proper subset of approximate-fair allocations. However, the set of fair allocations is not a proper subset of the set of approximate envy-free allocations or vice versa. Consider three agents and let ε be a small positive number that is the smallest portion of a resource that has any value for an agent. Consider the valuations of the distributions from the perspective of agent 1 (refer to Table 1). It is to be noted that the examples show that the portion of the resource allocated to agent 1 fulfills a particular criterion (say, being approximate envy-free). They are from the point of view of agent 1. In order for the allocation (the entire resource) to fulfill the criterion (of approximate envy-freeness), it is required that all agents believe the said criterion is fulfilled for their respective portions.

3 Criteria for protocols

This category mainly considers issues of negotiation from the MAS point of view. Rosenschein and Zlotkin (1994) formalized the idea of negotiation in MAS. Criteria such as stability, simplicity, and strategy-proofness are of primary concern. We discuss these and other related criteria in this section.

3.1 Symmetric (Rosenschein & Zlotkin, 1994)

An owner of an agent would not want the negotiation process to be arbitrarily biased against that agent. A protocol is called *symmetric* if it is not biased against any agent for arbitrary or inappropriate reasons. Thus, for example, an agent should not be awarded a contract because the owner of the agent happened to be a close relative of the contractor. Exactly what constitutes an inappropriate criterion depends on the specific domain. A good way to ensure symmetry is to hide the identity of the agent from the other parties involved in the negotiation.

3.2 Simplicity (Rosenschein & Zlotkin, 1994)

Protocol designers should endeavor to create simple protocols that make low computational demands on agents and minimize the communication overhead. If protocols encourage very unpredictable behavior among agents, then agent designers will be encouraged to create agents that model their opponents and attempt to compensate for the modeled behavior. This may, however, induce more unpredictability, because the opponents' models will be imprecise. As a result, each agent would be burdened with more computation, and overall system efficiency would be lowered as agents waste resources trying to outguess each other. Simplicity, therefore, becomes a strong normative requirement. Simplicity can be derived if protocols are stable. Since the optimal outcome is well known, there is nothing better to do than just carry it out.

3.3 Stability (Rosenschein & Zlotkin, 1994)

Systems that are stable can reduce agent computational load and communication costs. From a game-theoretic point of view, stability means systems that have at least one Nash equilibrium. A system is *stable* if no agent can do better by moving away from its current strategy while other agents maintain their current strategies. Thus, protocols need to be created that encourage agents to keep strategies that are in equilibrium with each other. Stability cannot be enforced but can only 'emerge' voluntarily from agents, because it is assumed that agents are autonomous and rational. Since the agents are not necessarily cooperative, the protocols need to induce stable behavior from the agents so that the entire system can behave predictably.

3.4 Anonymity and privacy[2]

Privacy deals with the issue of an agent keeping its proprietary information hidden from competing agents. The Internet is an open network and agents may try to attain privacy through anonymity, whereby agents do not reveal their identities. Anonymity can protect agents from being profiled by opponents, who might then exploit the information in future negotiations. Open-cry auctions tend to reveal the identity of agents, since they are held publicly. In an English auction, for example, every agent's true utility is revealed except the winner's. This is because every rational agent will stop bidding once the price of an item has exceeded the true utility for that agent. Anonymity can be assured in a sealed-bid setting, however. The auctioneer need to only inform every agent (that is, other than the winner) that it has lost. The winning bidder and bid need not be communicated to every agent. In this way, anonymity can improve the privacy of personal agent information. Privacy is therefore an important criterion in negotiation protocol design.

3.5 Strategy-proof (Rosenschein & Zlotkin, 1994)

A protocol is said to be *strategy-proof* if each agent declaring its true evaluation is a dominant strategy. This is a game-theoretic criterion. A typical example of a strategy-proof mechanism is the Vickery auction, where the protocol requires that agents submit sealed bids to the auctioneer and the agent quoting the highest price wins, but is required to pay only the amount of the second highest bid. Alternatively, the term incentive compatibility has been used. A negotiation mechanism is called *incentive compatible* when the strategy of telling the truth (or behaving according to one's true goals) is in equilibrium. Strategy-proof protocols lead to outcomes where lying by a particular agent will negatively affect its own utility while all other agents do no worse than getting what they would have received if the agent were truthfully revealing its utility.

If a negotiation protocol is known to be strategy-proof, then agent designers will not attempt to manipulate the system. There would be no benefit in creating sophisticated agents that attempt to outwit other agents. Agents that try to guess other agents' moves in an attempt to misrepresent and manipulate the system often end up making poorer choices. Since lot of agents would then end up with lower utilities than if they had made simpler choices, the efficiency of allocation is lower when mechanisms are not strategy-proof. Similarly counterproductive agents might choose portions that are less than fair in order to 'spite' another agent.

An example of such a situation is provided by Brams and Taylor (1996). Section 1.4 explains how the two-person divide-and-choose procedure is not strategy proof and the agents can end up making suboptimal choices in an attempt to 'outwit' each other.

While strategy-proof mechanisms allow fairer and more efficient outcomes, it may not be preferred by participating agents, because it tends to make their private information public, thus making them more prone to manipulation in future negotiations.

[2] We do not provide any concrete references to the criteria as we encountered only passing discussions on these criteria in the existing literature.

3.6 Centralized vs. distributed[2]

As it happens, while MAS are discussed as an example of distributed systems, the interaction protocols that are designed for sharing resources are centralized in nature. Distributed protocols are generally more difficult to create and many protocols suggested may involve some degree of centralization at subtler levels. The main drawback of centralized protocols is that the system is prone to a single point of failure. Thus, in any auction mechanism, if the central auctioneer goes offline, the entire system goes down. Distributed negotiation schemes are by contrast less susceptible to failures due to any particular agent going offline. They are fault tolerant. In order to achieve a decentralized protocol however, some redundancy needs to be built into the system in order that other agents may assume the role of the failed agent. Thus, redundancy is required in terms of roles, resources, and information. The trade-off is that distributed systems may arrive at less globally efficient outcomes and take more time to reach stability. However, distributed systems can also protect the privacy of agent information better. The following statement in Chevaleyre *et al*. (2006) sums it up aptly, 'The distributed model seems more natural in cases where finding optimal allocation may be computationally infeasible. Even small improvements over initial allocation of resources would be considered a success. Stepwise improvements over the *status quo* are naturally modeled in a distributed negotiation framework'. The English auction and the Dutch auction are examples of centralized systems, whereas the contract net protocol is an example of a distributed mechanism. This discussion of centralized vs. distributed is equally applicable to the discussion about criteria for procedures in Section 4. However, this topic has been researched more extensively by the MAS community with relation to the design of protocols, so is included here.

3.7 Mediated vs. arbitrated[2]

Arbitration is a process by which contending parties can submit their dispute to an arbitrator who renders a judgment on how the dispute is to be resolved. The judgment is then binding on the parties involved. The arbitrator is expected to be impartial and unbiased. Arbitration is also the process involved when disputes are settled through the legal court system. In multiagent societies, an auctioneer can be considered an arbitrator. While most auctions have clear criteria on winner determination (e.g., lowest bidder wins in a call for proposals), if various other properties need to be considered, such as swift job completion or quality of service, the winner will depend on the opinion of the arbitrator. This is used less often in MAS due to the inherent subjectivity involved in judgments. Mediation, on the other hand, involves an unbiased mediator who *facilitates* the settlements of disputes. A mediator does not pass judgment or opinions, but rather clarifies the rules of the negotiation and acts as a catalyst between opposing interests. The one holding the knife in a moving-knife procedure[3] is performing the role of a mediator. Arbitration is *adjudicatory*, as opposed to mediation, which is *advisory*.

Procedures that require either a mediator or arbitrator suffer from the weakness that the bias (or lack thereof) of the arbitrator/mediator cannot be proven definitively. Generally, bilateral negotiation protocols do not need any mediators. Multilateral negotiation, on the other hand, features some form of a mediator, who sifts through the various offers to devise an allocation satisfactory to all agents.

3.8 Communication complexity (Endriss & Maudet, 2005)

Communication complexity is more of a concern for MAS researchers than mathematicians or economists. Centralized protocols tend to be more efficient in communications (since all agents

[3] The moving-knife procedure is so-called, because it is assumed that a knife moves smoothly over a cake demarcating an increasingly larger portion until an agent calls 'Stop', to indicate the portion is satisfactory to the agent.

have to talk only to the auctioneer) than distributed ones. Similarly, one-shot protocols communicate more frugally than iterative protocols. Endriss and Maudet have discussed communication complexity of a protocol based on an earlier proposal by Yao (1979). Communication complexity can be defined as: 'the maximal number of bits exchanged when following the protocol (in the worst case)'. Endriss and Maudet mention another result by Feldman (1973) that deals with the communication complexity of agent negotiation, 'If the costs of arranging a multilateral deal were proportional to the number of pairs in a group of agents, then communication costs would rise quadratically $O(n^2)$. If the costs were proportional to the number of subgroups in a group, then the communication costs would rise exponentially $O(2^n)$'. The communication costs are studied as a combination of the number of deals and communications to agree on a deal that would be needed to finalize an allocation. Endriss and Maudet have given a detailed analysis of the communication complexity for various classes of utility functions that allow any rational deal to be implemented.

3.9 Time (Kraus & Wilkenfield, 1991)

Time has not been studied seriously as a criterion for negotiation protocols. This is because the environment is assumed to be static when the negotiation takes place. The assumption can be wrong in many ways. If the resource is perishable, then agents can benefit more from a quick but inefficient protocol rather than a lengthy but efficient one. If, by the end of the negotiation, the resource has lost its utility by a significant amount, then the objective of profiting from ownership of the resource is defeated. There are other time constraints possible. If agents count the time required for negotiation as time lost in following up other potential opportunities, then it makes sense to keep the negotiation short. Also, the resources may be required for accomplishing a specific task that has a completion deadline. A negotiation protocol is useless if it achieves a good bargain in acquiring a good but the agent no longer needs it. Agents' attitudes towards negotiation time directly influence the kinds of agreements they will reach. Kraus *et al.* (1995) have studied this issue and shown that the agreements can be reached without delay. Thus the inefficiency of negotiating in many steps is avoidable.

3.10 One-shot vs. iterative[2]

A majority of the protocols put forth in the literature are iterative. Iterative protocols allow the gradual discovery of other agents' valuations. Thus, each agent can adjust its offer according to the latest information it has on the other agents' bids. This allows for efficient outcomes. The open-cry English auction and Dutch auctions are examples of iterative protocols. Such protocols can however be costly in terms of communication overhead. They also require more time to complete. One-shot negotiation protocols offer an advantage in this respect. Examples include the first-price sealed bid and the second-price sealed bid (also called Vickery auctions). Only one round of messages are sent by the agents, after which a mediator or auctioneer decides who wins the resource. One-shot protocols can lead to inefficient outcomes, because agents do not have any chance to adjust their offers after knowing their opponents bids. They, however, protect the privacy of agent bids. A one-step protocol has been proposed by Rosenschein and Zlotkin (1994) as an alternative to the multi-step Monotonic Concession Protocol. The Monotonic Concession Protocol put forth by Rosenschein and Zlotkin is succinctly explained by Jennings *et al.* (2001) as follows: 'Negotiation proceeds in a sequence of rounds, where at every round, each agent puts forward a proposal. If the proposals overlap, then agreement has been reached. If the proposals do not overlap, then negotiation proceeds to a further round, where the agents either make a concession or else put forward the proposal they made in the preceding round. If neither agent makes a concession, then negotiation terminates with a "conflict deal". This simple protocol ensures that negotiation either monotonically proceeds towards a solution, or else terminates'. Since it is guaranteed that the Monotonic Concession Protocol will lead to a stable outcome, Rosenschein proposes that the agents jump all steps and mutually agree on the final outcome in the first step itself. In such cases, a one-shot protocol is as efficient as the iterative protocol while having very low communication and computational needs. Thus, if the domain is uncertain and dynamic, then iterative protocols need to be used to allow agents to 'discover' alternative allocations. On the other hand, domains that are simple and well known allow construction of one-shot protocols.

4 Criteria for procedures

Mathematicians have generally dominated in providing procedures or 'recipes' for allocating resources. Their approach to solving fair division problems bears remarkable similarity to the way MAS research characterizes agent preferences. Many assumptions explicitly made in MAS (such as rationality and autonomy) are implicitly considered by mathematicians when suggesting resource allocation procedures. Other criteria, such as envy-freeness and efficiency, which generally belong in the domain of economics, are considered explicitly. Economists suggest the criteria in a normative tone or to show existence results, whereas mathematicians generally try to incorporate the criteria into their solutions. In both cases, however, the semantics of the terms are the same. We therefore do not revisit criteria such as envy-freeness and efficiency, as they have been discussed earlier. The criteria that follow are more exclusively focused on evaluating the procedural aspects of resource allocation.

4.1 Bounded vs. unbounded number of agents

Is the procedure for resource allocation applicable to n agents in general? This is an important criterion for evaluating the applicability of procedures in real-world situations. The well known divide-and-choose methodology (Brams & Taylor, 1996) is applicable to only two agents. While it has many nice features, such as simplicity, envy-freeness, and Pareto optimality, its biggest limitation is that it applies only to systems of exactly two agents. Procedures (Stewart, 1998) have been suggested that extend it to an unbounded number of agents while trying to preserve its essence, but invariably they end up compromising on at least one of the features. They most commonly lose efficiency and/or envy-freeness, while still being fair. They also tend to be more complex. Thus, to get procedures to scale to a general n number of agents, a number of compromises have to be made. Our literature survey shows specific envy-free, efficient solutions for $n = 3$ by Stromquist (1980) and Levmore-cook (Brams & Taylor, 1996), and $n = 4$ by Fink (1964). However, the only known envy-free, efficient procedure for any n is by Brams *et al.* (1997). It is an ingenious solution based on the notion of 'irrevocable advantage'. It is, however, quite complex and involves an arbitrarily large number of cuts. It is to be noted that there exist many alternative procedures if only fairness is demanded. The simplest example of an n-agent fair procedure is the moving-knife procedure, which is also Pareto optimal. Banach and Knaster (Steinhaus, 1948) have suggested a discrete procedure, based on trimmings, that is fair. However, it is inefficient.

4.2 Discrete vs. continuous[4] (Even & Paz, 1984; Robertson & Webb, 1998)

Discrete vs. continuous is another important distinction that needs to be considered. In general, discrete procedures are algorithmic and therefore implementable on a computer (but they might need a large number of inputs from every agent to discern their preferences). On the other hand, it is not clear how continuous procedures can be used in an automated fashion. A number of issues come into play if a machine wields the knife (say, in the moving-knife procedure). For example, the continuous resource may need to be discretized. The moving knife's position may be stated as the number of millimeters from the leftmost point of the cake. However, there can be ordering

[4] We use the comparison 'discrete vs. continuous' for characterizing procedures. This comparison could also apply to the discussion in Section 5.1 on resources. However, they carry different semantics in the two places. In order to reduce confusion, we use the term 'divisibility' when discussing resources.

issues if two or more agents call cut at the same millimeter length. This case never arises in the original procedure, because the resource was continuous and infinitely divisible (hence calls from two agents can be strictly ordered). Another issue is that of synchronization of agent messages. The moving-knife procedure is basically analogous to the English auction. So it is essential that the time order of agent messages be preserved. Unrelated externalities like network congestion can affect the ordering of the bids: an agent with a slow computer or internet connection might find its message reaching the mediator later than another's, although it sent its bid earlier. Many of these issues can be obviated if discrete procedures are implemented, as they are not dependent on the time order of messages, but carry out the processing in steps. They are also more amenable to analysis. Thus, various features like time and space complexity can be studied to find the discrete procedure most suited to the problem at hand. The major difference is that evaluation of portion sizes by all agents takes place simultaneously in continuous procedures, but is spread over binary comparisons in discrete procedures. Consequently, the running time for discrete procedures is much longer than for continuous ones. The number of cuts required is similarly larger in discrete procedures. This is because, for evaluating each portion, continuous procedures 'demarcate' the portion by the position of the knife. This is not true in discrete procedures. Thus, many continuous procedures have nice features like frugality and C-efficiency, described next.

4.3 Frugality (efficiency in number of cuts) (Brams & Taylor, 1996)

A procedure for n players is called *frugal* if it uses $n-1$ cuts to complete the resource allocation. All allocation procedures need to make comparisons before coming up with a feasible solution. Since continuous algorithms do not need to cut resources into portions while making such comparisons, they are generally very efficient in the number of cuts made. The moving-knife procedure does the job in the least number of cuts ($n-1$) possible. Since at any given time, k agents (the ones that have not yet received a portion) evaluate the size of the portion to the left of the cake simultaneously and continuously, the number of comparisons made is infinite. Discrete procedures, however, run in steps and can compare only two pieces at every step. Each comparison is made between portions actually cut out from the resource. Typically, the procedures iterate in the following manner, as shown in Figure 4.

There is only one discrete procedure known to complete in $n-1$ cuts. The literature shows that there are many continuous procedures that are frugal, while all but one of the discrete procedures are non-frugal. The divide-and-choose procedure is the only discrete procedure that is frugal, because it can accomplish a satisfactory division of a resource between two agents with just one cut.

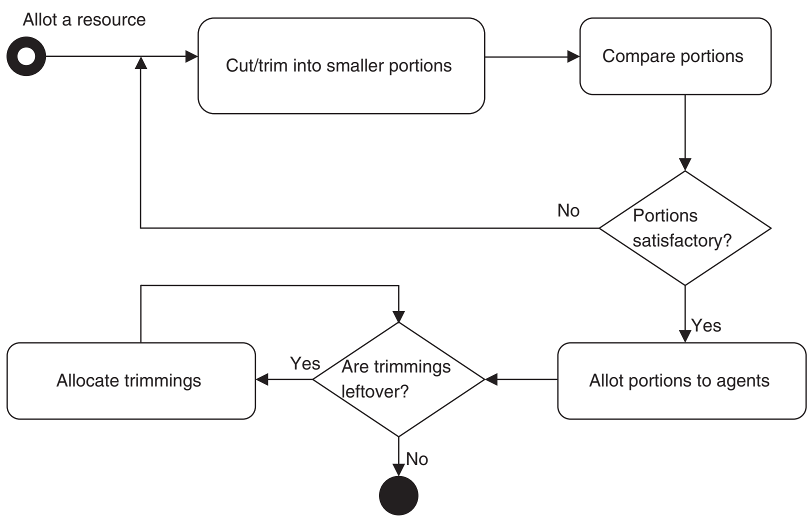

Figure 4 A general view of how discrete allocation procedures work

It is worth mentioning that the Selfridge–Conway discrete procedure[5] for three agents manages to allocate resources in a maximum of five cuts, and the 4-agent procedure proposed by Brams, Taylor and Zwicker (1997) produces an envy-free, efficient allocation with a maximum of 11 cuts. However, when the generalized procedure is extended to any $n > 4$ agents, it will allocate completely only after an arbitrary number of cuts.

4.4 C-efficiency (efficiency in frugal procedures) (Brams & Taylor, 1996)

The discussion of frugality showed the difficulty of obtaining discrete procedures that can complete an allocation in the fewest cuts ($n-1$) possible. Brams and Taylor discuss the notion of a weaker form of efficiency connected to frugal procedures. An allocation procedure for n players is called a C-procedure if it results in a set of $n-1$ cuts of a rectangular cake, all of which are parallel to the left and right edges of the cake. We can then define C-efficiency as follows: a C-procedure for n players is C-efficient if there is no other C-procedure for n players that yields an allocation that is strictly better for at least one player and as good for all others. That is, a C-procedure that is Pareto optimal is C-efficient. Brams and Taylor use this definition to connect envy-freeness and efficiency. They show that any envy-free C-procedure is C-efficient.

4.5 External payments

Most procedures that have been proposed involve agents redistributing resources among themselves. There are no external monetary compensations involved. This is quite true in cases where there is a single resource that is infinitely divisible. In such cases, the procedure can make exceedingly thin slices of the resource to create equitable divisions. However in some practical situations, the problem might consist of a bundle of heterogeneous resources that are indivisible. A typical example is the problem of divvying up inheritances, such as a house, car, and gold among heirs. Since the house and car are indivisible resources, procedures that apply to divisible resources cannot apply. But if we assume an extra pool of cash or gold available to each of the agents, then it is possible to arrive at envy-free allocations. Knaster's proposal (Steinhaus, 1948) suggests an internal bidding procedure where each heir places bids for each item reflecting its relative value in his or her eyes. The agent that bids highest wins the item. To achieve envy-freeness, the payments made to a common pool are divided among the agents so that the total of the object value and money value is the same for every agent. Procedures incorporating external payments can allow for envy-free and efficient division of resources. They do so by making money the 'divisible' resource to work around the indivisibility issue. The obvious drawback of such schemes is that agents are required to have an extra pool of cash available to make payments. Brams and Taylor put forth two point allocation schemes that do away with cash when dividing indivisible resources. The implicit assumption is that there exists at least one divisible resource along which envy-free and efficient allocations can be created. An additional difficulty is that they cannot predict ahead of time (before agents reveal their true utilities) which resource is the one that has to be divisible.

4.6 Verifiability (Iyer & Huhns, 2005)

Iyer and Huhns (2005) have proposed the notion of verifiability to evaluate procedures. This criterion is used to determine if an allocation generated by a procedure is reproducible or not. Due to their essential nature, none of continuous procedures are verifiable, that is, if a different agent runs the procedure again it will not be able to get the same division of resources. Formally,

[5] The Selfridge–Conway procedure is mentioned by Brams and Taylor in their book *Fair Division: From Cake-Cutting to Dispute Resolution*, on page 116. Regarding their source for the procedure, they state, 'Such a procedure was found about 1960 by John L. Selfridge and, later but independently, by John H. Conway […] Although never published by either, the procedure was quickly and widely disseminated by Richard K. Guy and others'.

a procedure is said to be *verifiable* if the allocation of the resource is invariant to the bias of the mediator or the agent. Thus, any agent that runs the procedure must get the same results as the mediator. In such a case, mediator bias does not play any role. This is quite unlike the case of continuous procedures like the moving-knife one, where the procedure is vulnerable to manipulation by the mediator. Therefore, no special qualifications are required of the mediator if the procedure is vulnerable. Such procedures are therefore self-mediating and do not require an outsider (with good reputation and unbiased judgment) to divide the resource. If procedures are verifiable, disputes can be quickly settled by simply executing the procedure once again. Unfortunately, auctions, sealed bids and most discrete procedures are not verifiable. Iyer and Huhns (2005) have proposed an n agent allocation procedure that is fair, unbiased, and verifiable.

4.7 Computational complexity

A lot of effort has been expended by mathematicians in devising procedures that fulfill criteria such as envy-freeness and efficiency. But little has been mentioned about the running time of various procedures. Continuous procedures are not generally studied because there is no clear way to implement them on a computer. As a basic observation, it can be said that most continuous procedures involve some form of a knife moving at a constant rate from one end of a cake to the other. A cut is made whenever an agent calls cut. We do not consider the operation of cutting to be a significant parameter in time calculations (although the number of cuts is a useful metric). Thus, in continuous algorithms, only the time taken by the knife to traverse the entire cake is to be considered and the time for execution of the algorithm is independent of the number of agents involved and can be taken to be a constant. We emphasize discrete algorithms more, because they are programmable and amenable to traditional complexity analysis.

There have been attempts (Even & Paz, 1984; Magdon-Ismail *et al.*, 2003) to determine the minimum number of cuts required to divide the resource fairly. A given (discrete) resource allocation procedure can be evaluated on two aspects: space complexity and time complexity.

Space complexity deals with the number of comparisons that a procedure needs to make in order to produce a satisfactory allocation. Space can be thought of as the amount of memory needed to store information about various portions of the resource in order for comparisons to be made. We therefore define space as the number of portions whose information needs to be held in order for the allocation to be determined. Magdon-Ismail, Busch and Krishnamoorthy show that sorting can be reduced to cake cutting, that is, any algorithm that performs fair division can sort. Since it is known that sorting algorithms generally take $\Omega(n \log n)$ comparisons, they deduce the same lower bound on computational complexity for resource allocation procedures as well. For envy-free procedures, they determine the complexity to be $\Omega(n^2)$. While *all known* procedures are verified to have such properties, they are unable to confirm that *every* procedure will have such properties.

Time complexity roughly corresponds to the number of steps an algorithm takes to compute a feasible allocation. This is generally dependent on the number of binary comparisons required, which is in turn dependent on the number of portions. The information about the portions in turn needs to be saved in memory. Thus, the time complexity corresponds to the space complexity for all (discrete and centralized) procedures.

4.8 Division independence (Chambers, 2005)

Chambers (2005) has put forth the property of division independence as a normative requirement of any division procedure. An allocation procedure satisfies *division independence* if the union of the portions allocated to an agent from various subparcels of a parcel is identical to the portion allocated from the initial parcel. The basic interest in this property is when the total amount of a resource available is initially unknown. Thus, the allocation procedure can be run on the fly with each new increment to the resource, with the guarantee that the combined portions so created will be the same as if the complete resource were initially available. Another appeal of such a property is that one can potentially create a distributed version of the procedure that be run in parallel on small chunks of the resource, since the resulting combined parcels would be the same allocation as the centralized version would have found. Unfortunately, Chambers has shown that this criterion negates other necessary criteria such as fairness and allows only dictatorial rules to exist. Division independence is therefore too constraining to be applied to realistic situations.

5 Criteria for resources

Mathematicians refer to resource allocation as the 'cake-cutting' problem. The resources are assumed to be like cakes and the problem is to divide the cake (resource) satisfactorily among *n* agents. This is because cakes are assumed to be divisible and recombinable. If a discrete good, like a house, needs to be distributed among heirs, then most procedures that were designed for cake cutting are not applicable. That is because such goods are indivisible. We describe below the various criteria that can be used to discover the kind of resource we are dealing with and hence appropriately devise allocation procedures for it.

5.1 Divisibility

Divisibility pertains to whether the resource is continuous or discrete in nature. Examples of indivisible resources include a house or a car. The constraints for procedures that allocate indivisible resources are stronger. A resource is called *indivisible* if it can be handed out only in integer units. Fractions of such resources will have no value for the agents. Divisibility on the other hand means that the resource is continuous and portions, however small, can be created. This implies that the resource is *infinitely divisible*, that is, any portion, however small, holds a finite value for every agent. In general, it is easier to find solutions for divisible resources rather than indivisible ones. For indivisible resources, a common trick that is employed is to append a divisible resource (like money) and to allocate them together, such that each agent believes it got a fair share. Any procedure that can allocate indivisible resources can work for divisible resources as well. All that needs to be done is to create portions of a continuous resource that can be treated as discrete units of a (divisible) resource. Thus, procedures for indivisible resources are a proper subset of the ones for divisible resources. A further qualification of divisible resources is whether they are recombinable or not. If a resource is *recombinable*, then a portion that is constituted from many smaller parts will hold a finite value to every agent. That is why cakes are the most commonly used paradigms for discussing resource allocation procedures. Cakes are assumed to be both infinitely divisible and recombinable. An example of a resource that is infinitely divisible but not recombinable is land. It is important to determine the type of resource we are working with before we can decide on the procedures that would be applicable to apportion it. For example, a procedure for cake cutting that guarantees certain properties like envy-freeness and efficiency (Jones, 2002) will not work for resources like land division. This is because, unlike cake, land is *not* recombinable.

Another example of a resource that is infinitely divisible but *not* recombinable would be the time available on a common central processing unit (CPU) for scheduling jobs. While ever-finer time slices can be created, two non-contiguous time slices cannot be recombined. It is preferable for an agent to receive a large contiguous chunk of time rather than many small, scattered chunks of time on the CPU due to the overhead of starting and stopping jobs. Table 2 shows examples of various categories of resources.

5.2 Homogeneous vs. heterogeneous

Resources are *homogeneous* if they are composed of the same kind of material in every part. A cake covered entirely with chocolate icing is considered homogeneous. Barring external issues (like is it the edge of the cake or the center?) every agent is then supposed to value every part of the resource as having the same value. *Heterogeneous* resources have different materials or compositions—and thus different values—in different parts. A cake, one half of which is covered in

Table 2 Various problem domain types

Problem domain properties	Infinitely divisible	Combinable
A single row of seats in a theatre	–	–
Time scheduling, Land	×	–
Circular cake	×	×

chocolate icing and the other half in vanilla, is considered heterogeneous. Agents value the different parts of the cake differently. Most procedures assume the resource to be heterogeneous.

5.3 Sharable or not (Chevaleyre et al., 2006)

A *sharable* resource can be allocated to a number of agents at the same time. An example of a sharable resource would be a picture taken by a satellite that can be allocated to several different agents at the same time. In general, any resource that is some type of information that needs to be transmitted can be shared simultaneously among several agents. Then, the only constraints are on the equipment (e.g., the camera that takes the picture) that is used to generate this information. Since most resources (unlike information) are exhausted on being shared, the canonical sense is that resources are generally not sharable.

5.4 Measurability (Bartle, 1995)

Measurability is a notion with a basis in set theory. A *measure* is a function that assigns a number representing a 'size', 'area', or 'volume' to subsets of a given set. Such a function therefore has normative requirements to fulfill:

1. It should be non-atomic, that is, the empty set should measure zero.
2. The measure should be countably additive.
3. Monotonic: the measure function should be non-decreasing in value.

A simple example is the counting measure defined as $\mu(S) =$ number of elements in S. Measurability is more of a theoretical concern, which does not affect the domain of resource allocation. It can be safely assumed that all real-world resources are measurable.

5.5 Dimensionality

An implicit assumption in the literature of fair division is that all resources are one-dimensional. The dimensionality of resources has never been considered as a parameter. Although cakes are three-dimensional, all procedures assume the knife that cuts the portions is parallel to one of the axes and that all agents agree on the same axis. There is a possibility, therefore, that more efficient allocations exist when the knife moves along one axis than another. But they may not be found, because the cake is essentially looked upon as a one-dimensional resource. The literature on land division, however, shows that resources like land have been considered as two-dimensional. The procedures begin an allocation by transforming the two dimensions to one. For example, consider a piece of land contended by many agents. If an agent marks a rectangular portion as its region of interest, the obvious treatment would be to calculate the area of the rectangle and compare it to the areas of other agents' rectangles in order to come up with a suitable allocation. Since 'area' might not be a common measure for all agents, some procedures allocate resources based on how much utility an agent gets from a portion of the (two-dimensional) resource. The process of transforming a two-dimensional preference into one dimension is therefore passed on to the agent, that is, the agent evaluates its preference for a two-dimensional portion by labeling it with a one-dimensional 'utility'. Thus, even where two-dimensional resources have been considered, the essential allocation is one-dimensional. A procedure

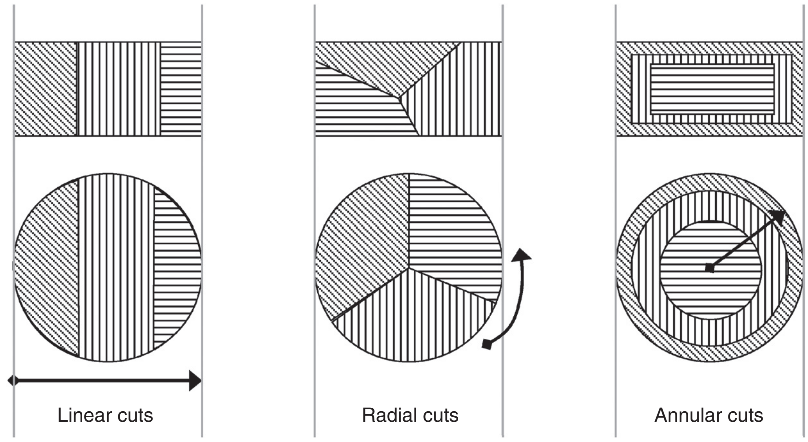

Figure 5 Cake-cutting using different topologies for rectangular and circular shaped cakes. The arrows show the direction of knife movement. The figure shows example allocations for three agents

for allocating two-dimensional resources has been proposed by Iyer and Huhns (2007) that calculates the allocation based on both dimensions. There is no discussion found in the literature for allocating resources of three or higher dimensions. It should be noted that dimensionality of a resource need not refer just to its physical dimensions. *Dimensionality* of a resource is the number of independent variables along which a resource can be defined. An example of a two-dimensional resource allocation problem is the allocation of bandwidth from a common Internet connection to various agents. One dimension can be the bandwidth of the connection while the other dimension can be the hours of the day when it is needed. Allotting resources with their natural dimensionality in mind might reasonably increase the efficiency of the allocation.

5.6 Topology

Topology is the qualitative study of geometric shapes. The precise shape of an object is not the primary focus. Rather, topologists study the features of objects that are invariant under continuous deformation. Thus, topologically, a sphere, a cube, and a tetrahedron are the same (no holes in the solid, hence homeomorphic to a 2-sphere). The topology of a resource is an important factor that affects the quality of an allocation. This is an aspect that has been overlooked in the current literature on negotiation. By default, the implicit assumption is that all resources are like an infinite flat two-dimensional Euclidean plane. This is true of the typical cake-cutting domain. Thus, the portions allocated are linear slices of the cake. The circular cake has been studied too. The circular cake exposes procedures that are topologically different. Thus, many circular cake-cutting procedures cut radial slices rather than cutting linearly. Figure 5 shows how cake-cutting procedures can approach the apportioning of resources.

Unlike a linear cake, a circle has no endpoints, which makes it topologically different. Hence, procedures arbitrarily mark one point on the circumference as the start and end point. Then a convention has to be chosen for the direction of rotation. Similarly, in two dimensions a flat rectangular paper has a different topology than a strip of paper wrapped around a cylinder (as shown in Figure 6). This is because the cylindrical surface has one dimension unbounded and the other bounded. We have options to create procedures that take advantage of the different topologies. There is, therefore, a strong appeal to study the topology of the resources before creating allocation procedures.

5.7 Resources vs. tasks (desirable vs. undesirable) (goods vs. bads) (Chevaleyre et al., 2006)

Tasks, or chores, can be considered to be resources that give negative utility to agents. Resources, typically, are of positive utility to agents. Thus, agents will try to maximize the resources they get while minimizing the tasks they have to perform. Criteria like envy-freeness can be suitably

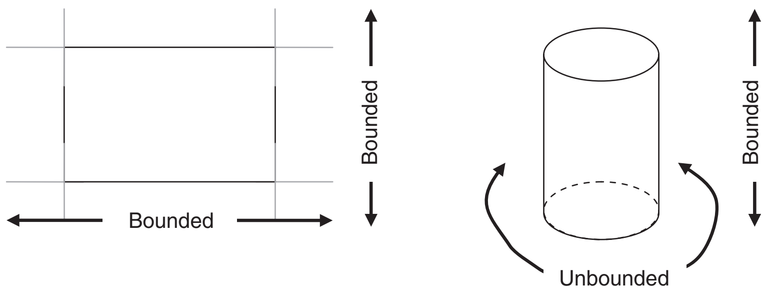

Figure 6 Different resources can have different topologies. Procedures should be designed with these topologies in mind to generate more efficient solutions for negotiating resource allocations

modified to account for tasks. Thus, an allocation is *envy-free* if each agent thinks that's its piece is *at least as small as* the smallest piece in the allocation. Tasks can be considered dual to resources. However, an important characteristic of tasks is that they are often coupled with constraints regarding their coherent combination. To begin a task, the completion of another task may be required as a precondition. Thus, clearly, procedures for resources allocation cannot be simply retrofitted for the allocation of arbitrary kinds of tasks.

5.8 Single vs. multiunit (Chevaleyre et al., 2006)

Multiunit resources are not individually distinguishable and agents should not be able to discriminate one unit from another. They have exactly the same features and characteristics. In such cases, allocations and agent preferences can be represented in a compact manner. However, more expressive languages are required to represent different quantities of the resource. Single unit resources are individually identifiable and agents can discriminate between units due to their different characteristics. Any multiunit problem can be transformed into a single unit problem by applying unique labels to each of them. The reverse is also true: a single unit problem can be converted to a degenerate multiunit problem. In order to convert a single-unit problem to a multiunit problem we do the inverse of the above procedure, that is, we remove labels (or names) that distinguish individual units, so that all units are now identical. It is this process of removing names we state as: 'converting a single unit problem to a degenerate multiunit problem'. Thus, removal of distinguishing labels entails the loss of information about the unique properties of individual units. This also means that all agents will be forced to value every unit equally, whereas, if it remained a single unit problem, they can assign different values to different units. Hence, a multiunit allocation procedure will perform worse (in terms of efficiency) than a procedure specifically designed for single units.

5.9 Static vs. perishable (Chevaleyre et al., 2006)

Perishable resources lose their value over time. Food is a good example of a perishable resource. Some protocols, such as the open-cry descending bid auction (Dutch auction), evolved to be able to trade quickly for perishable items like flowers. Normally, protocol designers assume the resource to be static, that is, the resource does not degrade or diminish with time.

5.10 Consumable vs. resusable (Chevaleyre et al., 2006)

Resources such as fuel and food are consumable. Once the resource has been made use of, it cannot be used again. Reusable resources can be reused after being put to work for a particular agent. The distinction as to whether the resource is consumable or perishable can be a matter of perspective. A satellite (used for weather imagery) is reusable but the specific time slot when it is needed (say 15th February 2005, from 9 am to 10 am) is not.

6 Criteria for agent preferences[6]

Agent preferences can be classified broadly into two types: ordinal (Sections 6.1–6.5) and cardinal (Section 6.6). We discuss ordinal preferences briefly here. The reader is encouraged to read a more detailed overview provided on this topic by Chevaleyre *et al*. (2006). In general, ordinal preference representation is appealing when agents can easily specify their preference between two alternative allocations.

6.1 Prioritized goals (Chevaleyre et al., 2006)

Agents can specify a list of goals (in decreasing order of their importance). A procedure can then be used to rank various alternative allocations based on how well they fit the *goal base* of the agent. This ranking can be determined by various criteria, such as best-out ordering, discrimin ordering, or leximin ordering.

6.2 Ceteris paribus (Chevaleyre et al., 2006)

Let V be a set of propositional variables that express three propositional formulas C, G, and G'. The *ceteris paribus* desire can be stated as: $C\colon G > G'[V]$. In plain terms this means that all other (irrelevant) options being equal, the agent prefers goal state G being true, even if goal state G' is false, rather than the converse. The preference relation can then be induced by obtaining a transitive closure of the union of preference relations induced by individual preference statements.

Next, we discuss the types of cardinal preference representation. The following subcategories of cardinal preferences are known:

6.3 Bidding languages (Chevaleyre et al., 2006)

Here the agents state the amounts they are willing to pay for every alternative allocation in which they are interested. Based on the language (OR or XOR) being used, an agent can determine the bid made by another agent for a particular bundle. For example, in the OR bidding language, the 'value of a bundle is taken to be the maximal value that can be obtained when computing the sum over disjoint bids for subsets of the bundle' (Chevaleyre *et al*., 2006).

6.4 Straight-line programs (SLPs)(Chevaleyre et al., 2006)

SLPs consist of a set of vertices in a (acyclic) graph. A vertex can be either of the *input* type (used for specifying the allocation as input) or of the *gate* type (used for computing a Boolean function based on its input). Some gates are outputs that are used to read the result (the computed utility of the allocation). SLPs have the advantage of being fully expressive, while being exponentially smaller in the number of bits required to encode the utility functions.

6.5 Weighted propositional formulas (Chevaleyre et al., 2006)

This is the cardinal analog of prioritized goals. Rather than rank goals to specify their importance, agents assign a 'weight' to specify the importance of each goal. One can then use various measures to rank alternative allocations based on how well they fulfill various goals in a goal base.

6.6 Utility functions

A succinct way to represent cardinal preferences is by using a utility function to assign a score to alternative allocations. Our literature survey shows that a number of assumptions have been made,

[6] An excellent exposition of how agent preferences can be represented is discussed by Chevaleyre *et al.* (2006). The reader is referred to their paper for further details on the topic of agent preferences. We briefly enumerate the types of agent preferences in this paper for the sake of completeness. In addition, we add a few other types to round off this section.

on the type of utility functions that agents can have, in order for allocation procedures to work. It is unknown if the properties of the procedures hold when agents' utility functions do not respect these assumptions. We mention the typical restrictions on agent utility functions mentioned in the literature:

6.6.1 *Non-atomic*

This requirement specifies that the agent utility functions smoothly decrease to zero as the amount of the resource tends to zero, that is, there should be no 'atoms' in the resource, where portions smaller than the 'atom' are of zero value to the agent.

6.6.2 *Additive*

This is a common requirement for utility functions and helps simplify a lot of analysis. If A and B are two disjoint subsets, then simple additivity states that:

$$v(A \cup B) = v(A) + v(B) \text{ for all disjoint } A, B \in \Sigma$$

6.6.3 *k-additive (Chevaleyre et al., 2006)*

Chevaleyre *et al.* define a k-additive utility function as follows: 'Given $k \in N$, a utility function is said to be k-additive if and only if there exists a coefficient α_T for each set of resources T of size at most k such that:

$$u(R) = \sum_{T \subseteq R} \alpha_T$$

The coefficient α_T represents the synergistic value of owning all items in T together, beyond the utility associated with any of the proper subsets of T'.

6.6.4 *Concave*

Concave domains are a subset of subadditive domains, where utility functions have the property:

$$v(A \cup B) \leqslant v(A) + v(B) \text{ for all disjoint } A, B \in \Sigma$$

They are less commonly modeled than simple additivity, because it is difficult to prove results in this domain. However, they more realistically reflect the utility of obtaining additional resources in the real world.

6.6.5 *Continuous*

Utility functions are assumed to be continuous. This means that for every amount x of the resource, the agent can assign a unique value, $v(x)$, for that amount. This simplifies analysis, but may not be necessarily true in the real world. In addition, utility functions are said to be smooth if there are no 'kinks' in the function value anywhere, that is, the function is differentiable at all points of the curve.

7 Conclusion

The goal of this paper is to explicate and organize criteria that are applicable to protocol and procedure design for negotiated resource allocation among autonomous entities. As part of our research in multiagent resource allocation, we realized that much investigation has been done on similar topics by others, including mathematicians and economists. Our endeavor has been to summarize this research into a single document that discusses criteria explicitly mentioned in the research as well as identifies criteria that were implicitly assumed. In addition, new criteria have been proposed: verifiability, dimensionality, and topology. We hope that future negotiation protocol designers will find this useful as a starting point for protocol engineering and have an objective basis for comparing alternative mechanisms.

Acknowledgements

We would like to sincerely thank our two anonymous referees for their valuable input to improve various aspects of the paper. Their detailed comments on each aspect of the paper helped us in presenting this material in a clear and succinct manner.

References

Arnsperger, C. 1994. Envy-freeness and distributive justice. *Journal of Economic Surveys* **8**(2), 155–186.

Bartle, R. G. 1995. *The Elements of Integration and Lebesgue Measure*. Wiley-Interscience.

Bertsekas, D. P. & Gallager, R. G. 1992. *Data Networks*. Prentice Hall.

Brams, S. J. & Taylor, A. D. 1996. *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press.

Brams, S. J., Taylor, A. D. & Zwicker, W. S. 1997. A moving-knife solution to the four-person envy-free cake-division problem. *Proceedings of the American Mathematical Society* **125**(2), 547–554.

Chambers, C. P. 2005. Allocation rules for land division. *Journal of Economic Theory* **121**, 236–258.

Chevaleyre, Y., Dunne, P. E., Endriss, U., Lang, J., Lemaitre, M., Maudet, N., Padget, J., Phelps, S., Rodriguez-Aguilar, J. A. & Sousa, P. 2006. Issues in multiagent resource allocation. *Informatica* **30**(1), 3–31.

Dubins, L. E. & Spanier, E. H. 1961. How to cut a cake fairly. *American Mathematical Monthly* **68**, 1–17.

Endriss, U. & Maudet, N. 2005. On the communication complexity of multilateral trading: extended report. *Journal of Autonomous Agents and Multi-Agent Systems* **11**(1), 91–107.

Even, S. & Paz, A. 1984. A note on cake cutting. *Discrete Applied Mathematics* **7**, 285–296.

Feldman, A. M. 1973. Bilateral trading, processes, pairwise optimality, and Pareto optimality. *Review of Economic Studies* **40**(4), 463–473.

Fink, A. M. 1964. A note on the fair division problem. *Mathematics Magazine* **37**(5), 341–342.

FIPA. 2002. *FIPA ACL Message Structure Specification*. `http://www.fipa.org/specs/fipa00061/SC00061G.html`

Fishburn, P. C. & Sarin, R. K. 1994. Fairness and social risk I: unaggregated analyses. *Management Science* **40**(9), 1174–1188.

Iyer, K. & Huhns, M. 2005. Multiagent negotiation for fair and unbiased resource allocation. *OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2005*. Springer.

Iyer, K. & Huhns, M. N. 2007. *A Procedure for the Allocation of Two-Dimensional Resources in a Multiagent System*. USC CSE Technical Report TR-2007-003, Department of Computer Science and Engineering, University of South Carolina.

Jennings, N. R., Faratin, P., Lomuscio, A. R., Parsons, S., Wooldridge, M. J. & Sierra, C. 2001. Automated negotiation: prospects, methods and challenges. *Group Decision and Negotiation* **10**(2), 199–215.

Jones, M. A. 2002. Equitable, envy-free, and efficient cake cutting for two people and its application to divisible goods. *Mathematics Magazine* **75**(4), 275–283.

Kraus, S. & Wilkenfield, J. 1991. Negotiations over time in a multiagent environment. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*. Morgan Kaufman, 56–61.

Kraus, S., Wilkenfeld, J. & Zlotkin, G. 1995. Multiagent negotiation under time constraints. *Artificial Intelligence* **75**(2), 297–345.

Krumke, S., Lipmann, M., De Paepe, W. E., Poensgen, D., Rambau, J., Stougie, L. & Woeginger, G. J. 2002. How to cut a cake almost fairly. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 263–264.

Lomuscio, A., Wooldridge, M. & Jennings, N. R. 2003. A classification scheme for negotiation in electronic commerce. *Group Decision and Negotiation* **12**(1), 31–56.

Magdon-Ismail, M., Busch, C. & Krishnamoorthy, M. 2003. Cake-cutting is not a piece of cake. In *Proceedings of the 20th International Symposium on Theoretical Aspects of Computer Science*. Springer, 596–607.

Pazner, E. A. & Schmeidler, D. 1978. Egalitarian equivalent allocations: a new concept of economic equity. *Quarterly Journal of Economics* **92**(4), 671–687.

Rawls, J. 1971. *Theory of Justice*. Harvard University Press.

Robertson, J. & Webb, W. 1998. *Cake Cutting Algorithms*. A K Peters Ltd.

Rosenschein, J. S. & Zlotkin, G. 1994. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. MIT Press.

Steinhaus, H. 1948. The problem of fair division. *Econometrica* **16**, 101–104.

Stewart, I. 1998. Your half's bigger than my half! *Scientific American* **3**(4), 112–114.

Stromquist, W. 1980. How to cut a cake fairly. *American Mathematical Monthly* **87**(8), 640–644.

Weiss, G. (ed.) 2000. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press.

Yao, A. C.-C. 1979. Some complexity questions related to distributive computing (Preliminary Report). In *Proceedings of the eleventh annual ACM symposium on Theory of computing*. ACM Press, 209–213.