Coalitions among Computationally Bounded Agents

Tuomas W. Sandholm
sandholm@cs.wustl.edu
Washington University, Department of Computer Science
One Brookings Drive, Campus Box 1045, St. Louis, MO 63130-4899

Victor R. Lesser
lesser@cs.umass.edu
University of Massachusetts at Amherst, Department of Computer Science
Amherst, MA 01003

Abstract

This paper analyzes coalitions among *self-interested* agents that need to solve *combinatorial optimization problems* to operate efficiently in the world. By colluding (coordinating their actions by solving a joint optimization problem) the agents can sometimes save costs compared to operating individually. A model of *bounded rationality* is adopted where computation resources are costly. It is not worthwhile solving the problems optimally: solution quality is decision-theoretically traded off against computation cost. A normative, application- and protocol-independent theory of coalitions among bounded-rational agents is devised. The optimal coalition structure and its stability are significantly affected by the agents' algorithms' performance profiles and the cost of computation. This relationship is first analyzed theoretically. Then a domain classification including rational and bounded-rational agents is introduced. Experimental results are presented in vehicle routing with real data from five dispatch centers. This problem is $\mathcal{NP}$-complete and the instances are so large that—with current technology—any agent's rationality is bounded by computational complexity. [1]

Abbreviated title: Coalitions among Computationally Bounded Agents

[1]Supported by ARPA contract N00014-92-J-1698. The content does not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred. Tuomas Sandholm was also supported by a University of Massachusetts Graduate School Fellowship, the Finnish Science Academy, Finnish Culture Foundation, Finnish Culture Foundation Rank Xerox Fund, Information Technology Research Foundation, Transportation Economic Society, Leo and Regina Wainstein Foundation, Jenny and Antti Wihuri Foundation, Honkanen Foundation, Ella and George Ehrnrooth Foundation, and the Thanks to Scandinavia Foundation. A short early version of this paper appeared in [44].

1 Introduction

Automated negotiation systems with self-interested agents are becoming increasingly important. One reason for this is the *technology push* of a growing standardized communication infrastructure—Internet, WWW, NII, EDI, KQML [8], FIPA, Telescript [14], Java, *etc*—over which separately designed agents belonging to different organizations can interact in an open environment in real-time and safely carry out transactions. The second reason is strong *application pull* for computer support for negotiation at the operative decision making level. For example, we are witnessing the advent of small transaction commerce on the Internet for purchasing goods, information, and communication bandwidth [21, 30]. There is also an industrial trend toward virtual enterprises: dynamic alliances of small, agile enterprises which together can take advantage of economies of scale when available (e.g., respond to more diverse orders than individual agents can), but do not suffer from diseconomies of scale.

Multiagent technology facilitates the automated formation of such dynamic coalitions at the operative decision making level. This automation can save labor time of human negotiators, but in addition, other savings are possible because computational agents can be more effective at finding beneficial short-term coalitions than humans are in strategically and combinatorially complex settings.

This paper discusses coalition formation in inherently distributed combinatorial problems—e.g., resource and task allocation and multiagent planning and scheduling—in situations where agents may have different goals, and each agent is trying to maximize its own good without concern for the global good. Such self-interest naturally prevails in negotiations among independent businesses or individuals. In building computer support for coalition formation in such settings, the issue of self-interest has to be dealt with.

In *cooperative distributed problem solving* [7, 5], the system designer imposes an interaction *protocol* [2] and a *strategy* (a mapping from state history to action; a way to use the protocol) for each agent. The approach is usually *descriptive*: the main question is what social outcomes follow given the protocol and *assuming that the agents use the imposed strategies.* On the other hand, in *multiagent systems* [34, 23, 5, 47, 41, 44], the agents are provided with an interaction protocol, but each agent will choose its own strategy. A self-interested agent will choose the best strategy for itself, which cannot be explicitly imposed from outside. The protocols need to be designed *normatively*: the main question is what social outcomes follow given a protocol which

[2]By protocol we do not mean a low level communication protocol, but a negotiation protocol which determines the possible actions that agents can take at any point of the negotiation. An example protocol is the *sealed-bid first-price auction*, where each bidder is free to submit one bid to take responsibility for a task, which is awarded to the lowest price bidder at the price of his bid. The analog of a protocol is called a *mechanism* in game theory [11, 24].

guarantees that each agent's desired local strategy is best for that agent—and thus the agent will use it. The normative approach is required in designing robust non-manipulable multiagent systems where the agents may be constructed by separate designers and/or may represent different real world parties.

Interactions of self-motivated agents have been widely studied in microeconomics—especially in game theory [28, 11, 24, 33]. Most of that work assumes perfect rationality of the agents [49, 18], e.g., flawless and costless deduction.

We extend the normative approach of game theory to settings where the agents lack full rationality because they cannot enumerate or evaluate all alternative solutions to a coalition's optimization problem.[3] Instead, they have to search for good solutions. Such search incurs expenses in terms of CPU time. Therefore it is unreasonably costly to attempt to find optimal solutions to hard problems. Instead, solution quality needs to be traded off against the cost of computation.

1.1 Example application: distributed vehicle routing

The methods presented in this paper are needed in settings where the agents are self-interested, and there is an underlying intractable combinatorial problem that limits the agents' rationality because the problem cannot be solved optimally in practice. Applications with these two characteristics include distributed vehicle routing among independent dispatch centers, manufacturing planning and scheduling among multiple agile enterprises, meeting scheduling, scheduling of patient treatments across hospitals, classroom scheduling, planning and scheduling of multi-contractor software projects, multiagent information gathering on the World Wide Web, and allocating bandwidth in multi-provider multi-consumer computer networks, to name just a few. The methods developed in this paper are domain independent. However, to make the concepts more concrete, the distributed vehicle routing problem will be used as an example throughout the paper.

The distributed vehicle routing problem that we study is structured in terms of a number of geographically dispersed dispatch centers of different companies. Each center is responsible for certain deliveries and has a certain number of vehicles to take care of them. So each agent—representing a dispatch center—has its own vehicles and delivery tasks. The local problem of each agent is a heterogeneous fleet multi-depot routing problem with the following constraints.

- Each vehicle has to begin and end its tour at the depot of its center (but neither the pickup nor the drop-off locations of the orders need to be at the depot).

[3] Others in game theory have examined the effects of computational limits on rational play in settings where agents play a combinatorially trivial game, but complexity stems from numerous repetitions of that same game, see e.g. [31].

- Each vehicle has a maximum load weight constraint. These differ among vehicles.
- Each vehicle has a maximum load volume constraint. These also differ among vehicles.
- Each vehicle has the same maximum route length (prescribed by law).
- Every delivery has to be included in the route of some vehicle.

The objective is to minimize transportation costs: the domain cost is the sum of the route lengths of the vehicles in the solution that has been reached.

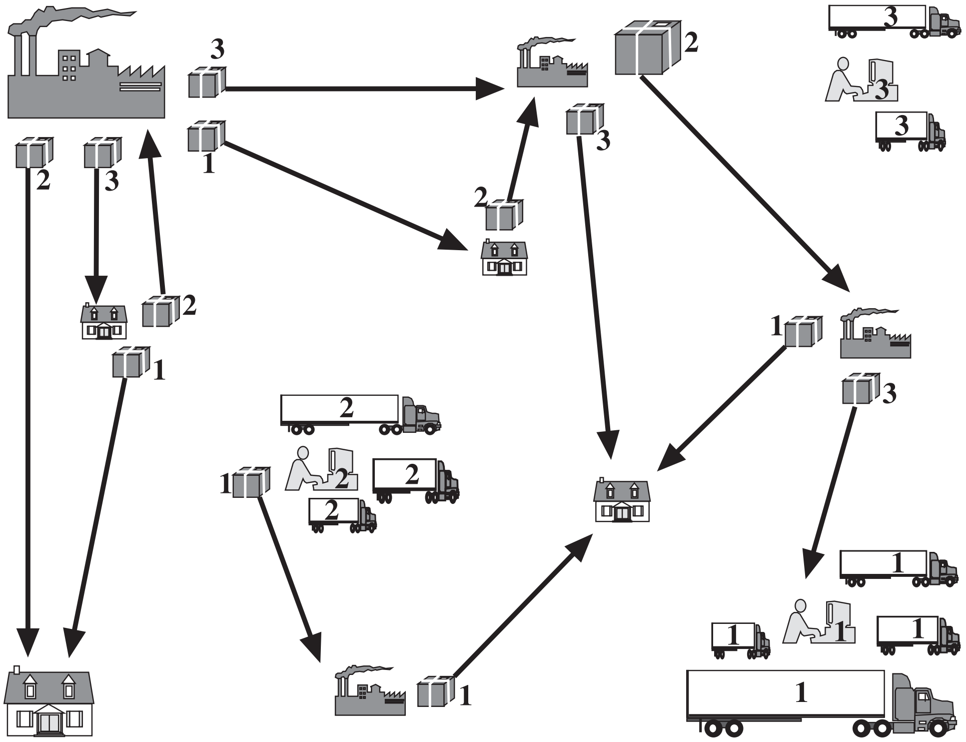

Figure 1: *Small example problem instance of distributed vehicle routing. This instance has three dispatch centers represented in the figure by computer operators. They receive the delivery orders and route the vehicles. Each parcel is numbered according to the dispatch center that is responsible for delivering it. No routing solution is shown.*

The problem is $\mathcal{NP}$-hard, because ΔTSP can be trivially reduced to it.[4] It is in $\mathcal{NP}$, because the cost and feasibility of a solution can easily be checked in polynomial

[4] ΔTSP is a Traveling Salesman Problem where the distances between cities satisfy the triangle inequality.

time. Thus, the problem is $\mathcal{NP}$-complete. Moreover, the problem instances in our experiments are so large that even the smallest ones are too hard to solve optimally—unlike the one in Figure 1.

The geographical operation areas of the centers overlap. This creates the potential for multiple centers to handle a delivery. The cost of handling it may vary between agents because it can often be less expensively integrated into adjacent routes than remote ones while honoring the weight, volume and route length constraints. So it can often be incorporated with lowest cost into the routing solution of the center that happens to have adjacent routes. The asymmetric costs among agents for handling a delivery often make it beneficial to reallocate delivery tasks among agents. This allows considerable cost savings from coordination among the agents.

Distributed vehicle routing is a real world problem, and the problem instances used in the experiments of this paper were collected from five real dispatch centers. They represent one week delivery order and vehicle data.[5] The collected data is characterized in Table 1. Our prior work has already focused on different aspects of automated negotiation in this domain [37, 40, 44, 38, 39, 26, 45, 42], and lately other researchers have studied an almost identical problem, yet with randomly generated instances and with a non-normative approach [9]. Also, simpler routing problems have often been used as example applications in recent multiagent systems research [34, 59, 52, 57].

Table 1: *One week of real vehicle and delivery data used in the experiments.*

Dispatch center	Number of delivery orders	Number of vehicles	Average delivery length
1	65	10	121 km
2	200	13	169 km
3	82	21	44 km
4	124	18	145 km
5	300	15	270 km
All	771	77	187 km

[5]Company A owned the first three centers and company B owned the last two. Even though some of the dispatch centers were owned by the same company, in practice they acted self-interestedly because they had their own fiscal goals. The centers were located around Finland.

1.2 Coalition formation setting

In many domains, self-interested real world parties—e.g., companies or individual people—need to solve combinatorial optimization problems to operate efficiently. Often they can save costs by coordinating their activities with other parties. For example when the planning activities are automated, it can be useful to automate the coordination activities as well. This can be done via a negotiating software agent representing each party. In such automated negotiations among self-interested agents, the following questions arise: what coalitions should the agents form, are they stable, and how should costs be divided within each coalition? Coalition formation includes three activities:

- *Coalition structure generation*: formation of coalitions by the agents such that agents within each coalition coordinate their activities, but agents do not coordinate between coalitions. Precisely this means partitioning the set of agents into exhaustive and disjoint coalitions. This partition is called a *coalition structure* (CS). For example, in the vehicle routing problem, coalition structure generation involves choosing which dispatch centers will work together as coalitions.

- *Solving the optimization problem* of each coalition. This means pooling the tasks and resources of the agents in the coalition, and solving this joint problem. For example, in the vehicle routing problem this means solving a routing problem with the delivery orders and vehicles of all member agents. The coalition's objective is to maximize monetary value: money received from outside the system for accomplishing tasks minus the cost of using resources.[6]

- *Dividing the value* of the generated solution among agents. This value may be negative because agents incur costs for using their resources.

These activities interact. For example, the coalition that an agent wants to join depends on the portion of the value that the agent would be allocated in each potential coalition.

This paper addresses these coalition formation activities with a special emphasis on settings where the optimization problem cannot be solved exactly due to computational limitations. The paper is organized as follows. Section 2 describes our model of bounded rationality where computation cost precludes enumerating and evaluating all solutions to a coalition's optimization problem. Section 3 studies the optimal

[6]In some problems, not all tasks have to be handled. This can be incorporated by associating a cost with each omitted task. Then problem solving also involves the selection of tasks to handle. The theory of this paper applies to such cases but in our example application, all tasks have to be handled, and no payments from outside the system are received for them.

coalition structure, and Section 4 analyzes its stability. Section 5 presents experimental results in the distributed vehicle routing domain with real data. Externalities and agents with different problem solving capabilities are discussed in Section 6. Section 7 presents related research, and Section 8 concludes and describes future research directions.

2 Computation unit cost and algorithm as limits to rationality

Coalition formation has been widely studied [20, 56, 32, 53, 52, 59, 22], but to our knowledge, only among *rational agents* which can solve the coalition's optimization problem exactly, immediately, and without computation cost. This section describes how our model differs because it takes into account the cost of computation.

Let us call the entire set of agents A. Say that the lowest cost achievable by agents $S \subseteq A$ working together, but without any other agents, is c_S^R. For example, in a task allocation setting this is the minimum cost to handle the tasks of agents S with the resources of agents S. A coalition game in characteristic function form—i.e., a *characteristic function game* (CFG, Fig. 3)—is defined by a characteristic function v_S^R, which defines the *value* of each coalition S:

$$v_S^R = -c_S^R. \tag{1}$$

The superscript R emphasizes that we mean the *rational* value of the coalition, i.e., the maximum value that is reachable by the coalition given its optimization problem. A rational agent can solve this combinatorial problem optimally without any deliberation costs such as CPU time costs or time delay costs.

However, if the problem is hard and the instance is large, it is unrealistic to assume that it can be solved without deliberation costs. This paper adopts a *specific model of bounded rationality* [54, 15], where each agent has to pay for the computational resources (CPU cycles) that it uses for deliberation. A fixed computation cost $c_{comp} \geq 0$ per CPU time unit is assumed.[7] The domain cost associated with coalition S is denoted by $c_S(r_S) \geq 0$, i.e., it depends on (decreases with) the allocated computation resources r_S, Fig. 2 left. For example in the vehicle routing problem, the domain cost

[7]In practice, CPU time can already be bought, e.g., on supercomputers. Similarly, the developing infrastructure for remotely executing agents provides an equivalent setting. For example in Telescript [14], the remotely executing agents pay Teleclicks for CPU time to the owner of the host machine. In this paper, the market for CPU time is assumed to be so large that the demand of the agents that we are studying has negligible impact on the price of a CPU time unit. It is also assumed that this price is common to all agents, which corresponds to an open CPU cycle market.

is the sum of the lengths of the routes of the coalition's vehicles.[8] The functions $c_S(r_S)$ can be viewed as *performance profiles* of the problem solving algorithm. They are used to decide how much CPU time to allocate to each computation. With this model of bounded rationality, the value of a coalition with bounded-rational agents can be defined. Each coalition minimizes the sum of solution cost (i.e., domain cost, which decreases as more computation is allocated) and computation cost (which increases as more computation is allocated):

$$v_S(c_{comp}) = -\min_{r_S}[c_S(r_S) + c_{comp} \cdot r_S].^9 \tag{2}$$

This coalition value decreases as the CPU time unit cost c_{comp} increases, Fig. 2 right. Intuitively, as the unit cost of computation increases, agents need to pay more for the computation or they have to use less computation and acquire worse solutions accordingly. Our model also incorporates a second form of bounded rationality: the base algorithm may be incomplete, i.e., it might never find the optimal solution. If the base algorithm is complete, the bounded-rational value of a coalition when $c_{comp} = 0$ equals the rational value ($v_S(0) = v_S^R$). In all, the bounded-rational value of a coalition is determined by three factors:

- The *domain problem*: tasks and resources of the agents (e.g. trucks and delivery orders in our vehicle routing problem). Among rational agents this is the only determining factor.
- The *execution architecture* on which the problem solving algorithm is run. Specifically, the architecture determines the unit cost of computation, c_{comp}.
- The *problem solving algorithm*. Once the coalition formation game begins, the algorithm's performance profiles are considered fixed. This model incorporates the possibility that agents design different algorithms for different possible allocations of computation resources. We make no assumptions as to how effectively the algorithm uses the execution architecture. This is realistic because in practice it is often hard to construct algorithms that optimally use the architecture. For example, Russell and Subramanian have devised algorithms that are optimal for the architecture in simple settings, but in more complex settings they had to resort to an asymptotic criterion of optimality [35].

[8] In games where the agents receive revenue from outside—e.g., for handling tasks—this revenue can be incorporated into $c_S(r_S)$ by subtracting the coalition members' revenues from the coalition's domain cost.

[9] Throughout this chapter on coalition formation, *min*-operators are used due to their familiarity, although strictly speaking the value of such a *min*-operator may be undefined because $c_S(r_S)$ need not be continuous. Thus, to be precise, *inf*-operators should be used.

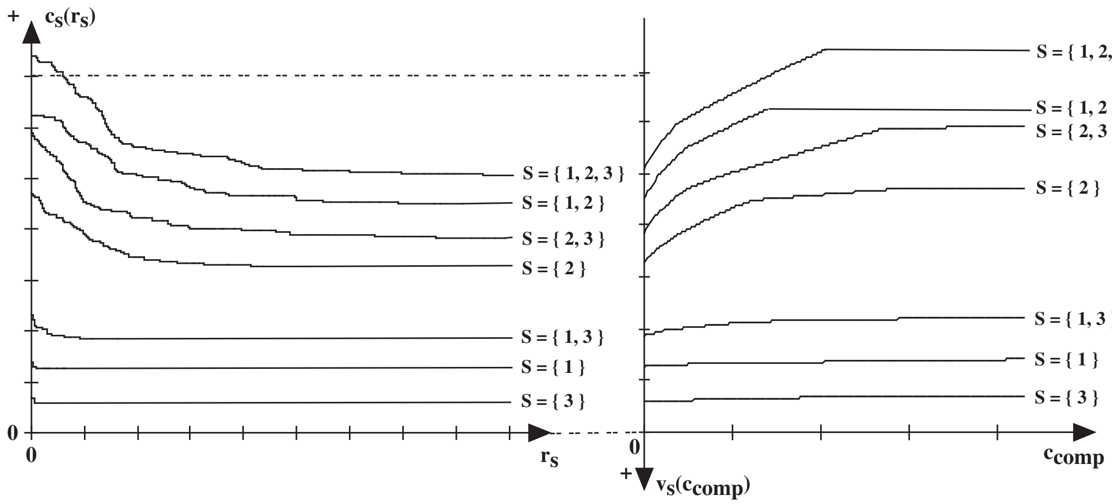

Figure 2: *Example experiment from the vehicle routing domain with agents 1, 2, and 3. Left: performance profiles, i.e., solution cost as a function of allocated computation resources. The curves become flat when the algorithm has reached a local optimum. Right: bounded-rational coalition value as a function of computation unit cost. The value of each coalition is negative because the cost is positive. The curves become flat at a computation unit cost c_{comp} that is so high that it is not worthwhile to take any iterative refinement steps: the initial solutions are used (their computation requirements are assumed negligible).*

2.1 Discussion of this model of bounded rationality

Conceptually we allow the agents to use *design-to-time algorithms* [12, 58, 13]: once an agent has decided how much CPU time r_S it will allocate to a computation, it can design an algorithm that will find a solution of cost $c_S(r_S)$. The design-to-time framework is used instead of the *anytime* framework [43, 4, 17, 58] because to devise a normative theory of self-interested agents, the possibility that they design their algorithms to time has to be accounted for. With deterministic performance profiles, for any desired CPU time allocation or solution quality, a noninterruptible design-to-time algorithm can be constructed that performs no worse than an interruptible anytime algorithm. In the worst case, the design-to-time algorithm may actually consist of executing the anytime algorithm. However, there are cases where the algorithm can be beneficially tailored for a specific CPU time allocation or solution quality, and in such cases the design-to-time algorithm will outperform the anytime algorithm.

We assume that the performance profiles exactly predict the solution cost attained for any given CPU time allocation. So, we have relaxed the assumption that the base level algorithm is optimal (complete and costless), but instead we assume that the deliberation controller (meta-level reasoner) is exact and costless. However, we do not assume that the deliberation controller composes the optimal sequence of base level computation actions since even the most advanced methods for such composition rely on assumptions that often do not hold in practice [36]. Assuming that the meta-level exactly and costlessly predicts the solution cost is more realistic than assuming optimality of the base level, but it still does not match reality exactly. In practice there is uncertainty in each performance profile: the meta-level is not exact.[10] Secondly, the performance profile depends on several features of the problem instance, and computing the mapping from the instance to the performance profile [43] may take considerable time, thus making the meta-level costly. In the limit, the base algorithm would be run at the meta-level to determine what it would achieve for a given time setting. Our assumptions regarding the meta-level enable us to analyze bounded rationality at the base level in isolation from uncertainty of the performance profiles. They also allow us to sidestep the problem of having a meta-meta-level controlling the meta-level, a meta-meta-meta-level controlling the meta-meta-level, and so on *ad infinitum*.

For now—this is relaxed in Section 6—we assume that the agents solve the combinatorial optimization problems equally well and that this is common knowledge. For any coalition's problem and for any setting of CPU time, the cost of the solution potentially generated by each agent is the same. The agents need not generate the same solutions, only the same quality.

With such shared deterministic performance profiles, each agent knows the value $v_S(c_{comp})$ of each potential coalition S up front. Therefore coalition formation is best off taking place before any computation. This guarantees that no computation is wasted. After collusion, each coalition computes its solution using the optimal amount of CPU time r_S as defined by Equation 2. Because in our model, rationality is bounded by CPU time cost, it costs the same for one agent to use nt CPU time units as it costs n agents to use t units. Therefore, it is best if a coalition's optimization problem is solved by a single agent. This is trivially true since an agent could simulate distributed problem solving among n agents for time t by using a local algorithm for nt. Conversely, it is not always possible (due to redundancy *etc.*) for n agents solving the problem for time t to reach a solution of the same quality as one agent using nt can reach. The computing agent can be arbitrarily chosen from within the coalition, and the coalition pays that agent its true cost for computing. This cost along with the domain solution cost contribute—as was defined in Equation 2—to $v_S(c_{comp})$, which

[10] If the performance profiles are only probabilistically known, anytime algorithms may be desirable due to their flexibility with respect to termination time. In general, for optimal meta-reasoning, the remaining part of a probabilistic performance profile should be conditioned on the algorithm's performance on that problem instance on previous CPU time steps [43, 58, 16].

is divided among the agents in the coalition as will be presented later. To summarize, with our model of bounded rationality it is best to centralize computation within each coalition but computation may be distributed across coalitions.[11]

3 Social welfare maximizing coalition structure

Outcomes of a game can be analyzed with respect to *social welfare*, which is defined as the sum of the agents' payoffs. The payoff that agent i gets is denoted by $x_i \in \Re$. The sum of the agents' payoffs has to equal the sum of the values of the coalitions in the coalition structure that formed.

A game is called *superadditive* if any pair of coalitions is best off by merging into one coalition:

Definition 3.1 *A game is* **superadditive** *if* $v^R_{S \cup T} \geq v^R_S + v^R_T$ *for all disjoint coalitions* $S, T \subseteq A$.[12]

When computation cost is ignored, superadditivity almost always holds, because at worst, the agents in the composite coalition can use the solutions that they had when they were in separate coalitions. A game can be non-superadditive only if the collusion process itself involves some cost, e.g., anti-trust penalties. The concept of superadditivity is important because it implies optimality of a specific coalition structure. Precisely, all superadditive games are *grand coalition games*, i.e., the agents are best off by forming the grand coalition where all agents operate together. In other words, in such games, $\{A\}$ is a social welfare maximizing coalition structure for rational agents. Some non-superadditive games are *subadditive*, Fig. 3:

Definition 3.2 *A game is* **subadditive** *if* $v^R_{S \cup T} < v^R_S + v^R_T$ *for all disjoint coalitions* $S, T \subseteq A$.

In subadditive games, the agents are best off by operating alone, i.e., $\{\{a_1\}, \{a_2\}$, ..., $\{a_{|A|}\}\}$ is a social welfare maximizing coalition structure for rational agents. Some games are neither superadditive nor subadditive because the characteristic function fulfills the condition of superadditivity for some coalitions and the condition of subadditivity for others. In other words, some coalitions are best off merging while others

[11]In our model, there is no cost to real-time. This corresponds to reality in settings where the domain cost is (practically) unaffected by the real-time that is used when consuming the optimal amount of CPU time. This can occur either because the amount of real-time is short, or because the domain cost is insensitive to real-time. To model settings where these conditions are not met, the cost of real-time should be incorporated into the model of bounded rationality. Under such a model, it might no longer be optimal to centralize the computation within each coalition because distributed computing may reduce the amount of real-time used.

[12]Definitions 3.1, 3.2 and 4.1 are from game theory.

are not. In such cases, the social welfare maximizing coalition structure varies. The grand coalition may be the optimal coalition structure even in games which are not superadditive. Similarly, every agent operating alone may be optimal even in games which are not subadditive.

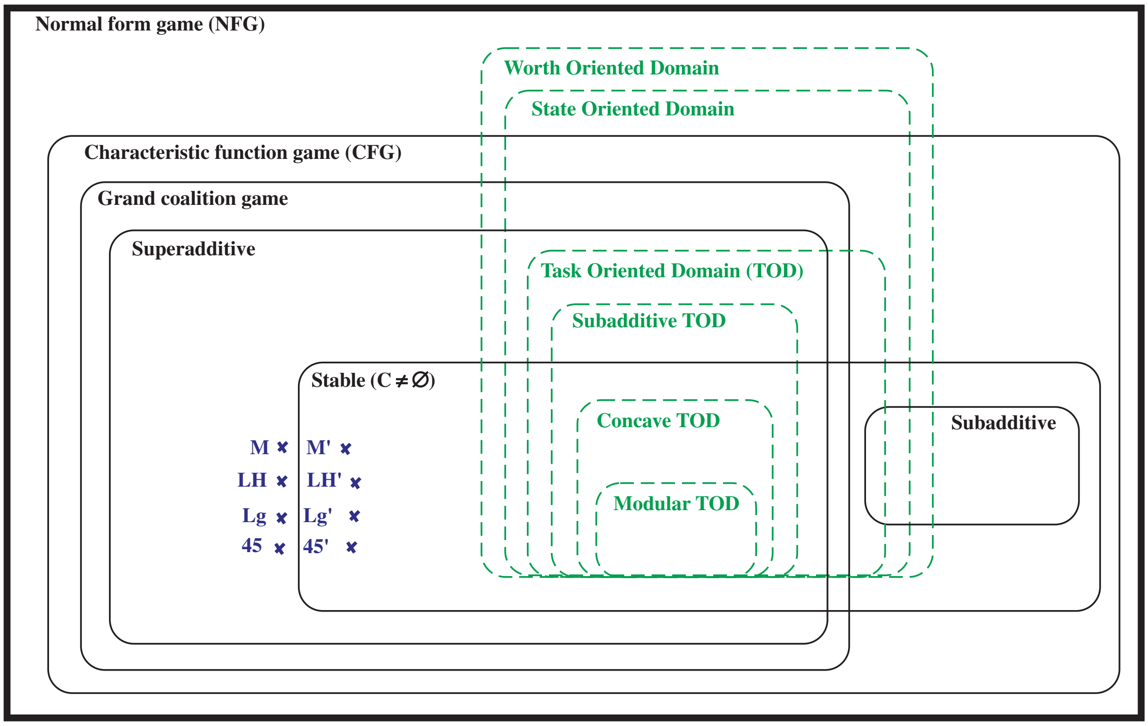

Figure 3: *Venn diagram of negotiation domains for rational agents. Normal lines show the classification from game theory. Dotted lines show the domain classification of Rosenschein and Zlotkin [34] . They use "Subadditive" to mean that an agent's cost for handling tasks is subadditive in tasks. We use subadditive to refer to coalition value functions that are subadditive in agents. The figure does not reflect the fact that Rosenschein and Zlotkin do not allow side payments.*

With bounded-rational agents, the coalition values incorporate the computation costs as described earlier. Now we generalize the concept of superadditivity to allow for bounded-rational agents. A game is called *bounded-rational superadditive* (BR-SUP) if any pair of coalitions with bounded-rational agents is best off by merging into one coalition. In other words, a game is BRSUP if the best value that one coalition can reach given the computation cost plus the best value that another coalition can reach given the computation cost is never greater than the best value that these coalitions can reach as a composite coalition given the computation cost:

Definition 3.3 *A game is* **bounded-rational superadditive (BRSUP)** *for com-*

putation unit cost c_{comp} *if* $v_{S \cup T}(c_{comp}) \geq v_S(c_{comp}) + v_T(c_{comp})$ *for all disjoint coalitions* $S, T \subseteq A$.[13]

Every BRSUP game is a *bounded-rational grand coalition game*. In such games, bounded-rational agents are best off by all working together, i.e., by forming the grand coalition. In other words, in such games, $\{A\}$ is a social welfare maximizing coalition structure for bounded-rational agents. There also exist bounded-rational grand coalition games which are not BRSUP: the grand coalition may be the optimal coalition structure although not all local poolings are beneficial.

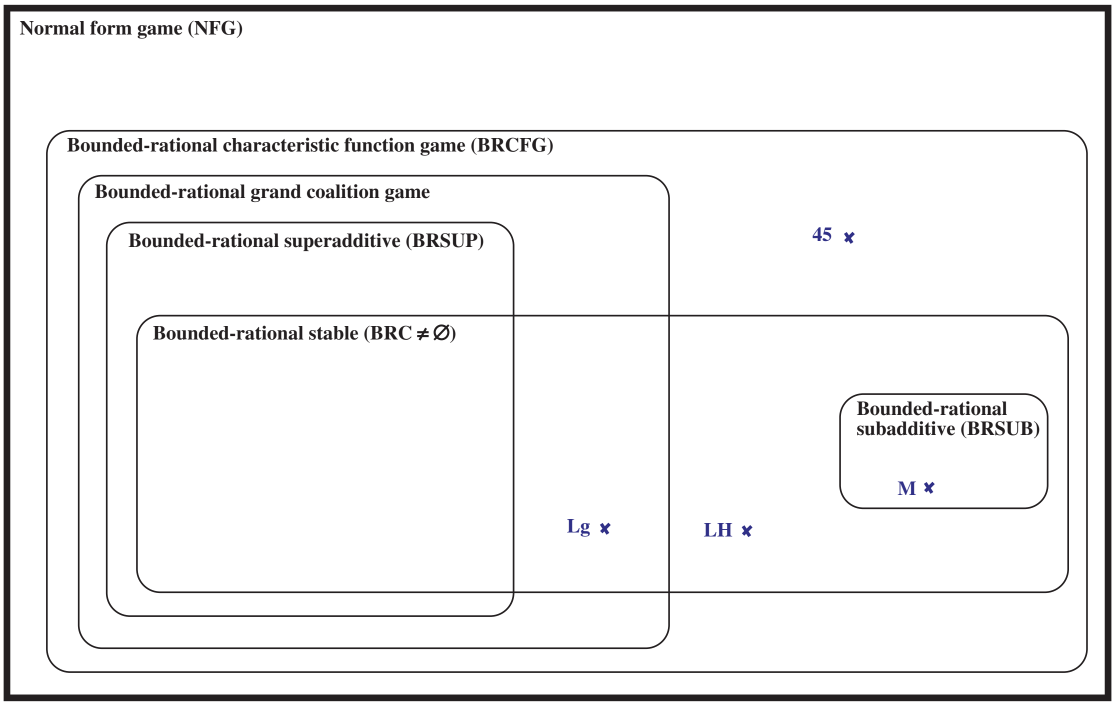

Figure 4: *Venn diagram of negotiation domains for bounded-rational agents. A particular game can lie in any region of this space, and it can simultaneously lie in any region of Figure 3. Therefore, the domain classifications for bounded-rational and rational agents can be merged by observing that each region of one classification intersects with each region of the other classification. By "region" we mean any area enclosed by lines, not just the named classes. To enhance readability, the two classifications are presented in separate figures.*

Bounded-rational superadditivity does not always coincide with superadditivity.

[13]The classic definition of rational superadditivity (Def. 3.1) is a special case of bounded-rational superadditivity (Def. 3.3) where the agents have complete algorithms (ones that find the optimal solution given enough computation time) and computation is costless ($c_{comp} = 0$).

In general, for a given computation unit cost, a game can be superadditive, BRSUP, both, or neither.

Only some non-BRSUP games are bounded-rational subadditive:

Definition 3.4 *A game is* **bounded-rational subadditive (BRSUB)** *for computation unit cost* c_{comp} *if* $v_{S \cup T}(c_{comp}) < v_S(c_{comp}) + v_T(c_{comp})$ *for all disjoint coalitions* $S, T \subseteq A$.

In BRSUB games, the agents are best off by operating alone, i.e., $\{\{a_1\}, \{a_2\}, ..., \{a_{|A|}\}\}$ is a social welfare maximizing coalition structure for bounded-rational agents. This coalition structure may be optimal even in games that are not BRSUB. In games that are neither BRSUP nor BRSUB, the optimal coalition structure varies, and several coalition structures may be equally good with respect to social welfare.

If the algorithm's performance profiles and the unit cost of computation are known, the bounded-rational values of potential coalitions can be computed according to Equation 2. Based on these values, all different coalition structures can be evaluated, and an optimal coalition structure determined. However, some general results hold which may make this enumeration process unnecessary. The rest of this section analyzes the relationship between the shape of the performance profiles and the class of the game. Specifically, the question is: what types of performance profiles make a game BRSUP (or BRSUB) for all computation unit costs? If the agents have these types of performance profiles, they know the optimal coalition structure irrespective of the unit cost of computation. This is important for example when agents are sent to execute on a remote host where the unit cost of computation is unknown.

Bounded-rational superadditivity depends on the performance profiles and the unit cost of computation. The next theorem states a natural condition on the performance profiles. If the condition holds, the game is BRSUP for any computation unit cost. Proofs of theorems are presented in appendix A.

Theorem 3.1 BRSUP (sufficient condition). *If* $c_{S \cup T}(r_S + r_T) \leq c_S(r_S) + c_T(r_T)$ *for all disjoint coalitions* $S, T \subseteq A$ *and all computation allocations* $r_S, r_T \geq 0$, *then the game is BRSUP for all computation unit costs.*

The condition states that the domain cost for coalition S after allocating a certain amount r_S of computation plus the domain cost to another coalition T after allocating a certain amount r_T of computation is never less than the domain cost of these coalitions combined after allocating $r_S + r_T$. This is always achievable in theory because in the worst case, the algorithm can allocate r_S on the problem of S and then allocate r_T on the problem of T separately. In other words, if the algorithm decomposes the problem by solving each agent's problem separately, bounded-rational superadditivity is trivially guaranteed by bounded-rational additivity. However, this trivial decomposition does not allow the agents to benefit from cooperation.

Some other problem decomposition—i.e., decision regarding which coalitions' problems to allocate computation on and how much—may guarantee bounded-rational superadditivity while allowing some benefits from cooperation. However, given a large coalition, it may be difficult to find an efficient decomposition.

Often, the algorithm that is used on the composite problem does not apply this type of problem decomposition. The real desideratum is not necessarily to generate algorithms that guarantee bounded-rational superadditivity (and thus the superiority of the grand coalition over other coalition structures), but algorithms that provide the highest social welfare (for the best coalition structure, which need not be the grand coalition). Sometimes these two goals are in conflict. Whether the algorithm's performance profiles actually satisfy the conditions for bounded-rational superadditivity without using a decomposition method depends on the problem, the specific instances under study, and the algorithm itself.

In general, a game can be bounded-rational superadditive for all computation unit costs even if the condition of Theorem 3.1 does not hold on the performance profiles:

Theorem 3.2 *$[c_{S\cup T}(r_S + r_T) \leq c_S(r_S) + c_T(r_T)$ for all disjoint coalitions $S, T \subseteq A$ and all computation allocations $r_S, r_T \geq 0] \not\Leftarrow$ Game is BRSUP for all computation unit costs.*

It is reasonable to assume that the performance profile $c_S(r)$ is decreasing in r if the agent can inexpensively store the best solution it has arrived at so far. Furthermore, $c_S(r)$ is often convex in r: greater savings are achieved in the early stages of computation and the savings per time unit decrease as problem solving proceeds. We conjecture that performance profiles of design-to-time algorithms are almost always convex. On the other hand, performance profiles of anytime algorithms are typically not convex at points where the base algorithm switches from one mode to another. One example is completing an iterative refinement algorithm by running an exhaustive complete algorithm after the refinement phase. Another example is switching from using one refinement operator (e.g., 2-swap in TSP [25, 40]) to using another refinement operator (e.g., 3-swap in TSP). Furthermore, refinements often decrease solution cost in a step-wise, noncontinuous manner rendering the performance profiles locally nonconvex—as in our experiments (Fig. 2 left). If the algorithm is stochastic, these step-related nonconvexities are reduced as the performance profile is averaged over multiple runs. The performance profiles in our experiments exhibited an overall convex nature, but also had true local nonconvexities (because the design-to-time algorithms were constructed from anytime algorithms, and were not tailored for each time setting separately, Section 5). Convexity is significant because with convex performance profiles, a domain is BRSUP for all computation unit costs if *and only if* the condition of Theorem 3.1 on the performance profiles holds:

Theorem 3.3 BRSUP (necessary and sufficient condition). *Let $c_U(r)$ be* **decreasing** *and* **convex** *in r for every coalition $U \subseteq A$. Now, $[c_{S \cup T}(r_S + r_T) \leq c_S(r_S) + c_T(r_T)$ for all disjoint coalitions $S, T \subseteq A$ and all computation allocations $r_S, r_T \geq 0] \Leftrightarrow$ Game is BRSUP for all computation unit costs.*

Analogous to Theorem 3.1, there is an easy sufficient condition on the performance profiles that guarantees that the game is bounded-rational subadditive for all computation unit costs:

Theorem 3.4 BRSUB (sufficient condition). *If $c_{S \cup T}(r_S + r_T) > c_S(r_S) + c_T(r_T)$ for all disjoint coalitions $S, T \subseteq A$ and all computation allocations $r_S, r_T \geq 0$, then the game is BRSUB for all computation unit costs.*

This implies that the agents would be best off by operating separately regardless of the execution platform.

Again, a game can be bounded-rational subadditive for all computation unit costs even if the condition of Theorem 3.4 does not hold on the performance profiles. Unlike in the case of bounded-rational superadditivity, the implication does not turn into an equivalence even for convex performance profiles:

Theorem 3.5 *$[c_{S \cup T}(r_S + r_T) > c_S(r_S) + c_T(r_T)$ for all disjoint coalitions $S, T \subseteq A$ and all computation allocations $r_S, r_T \geq 0] \not\Leftarrow$ Game is BRSUB for all computation unit costs. This holds even if all performance profiles are decreasing and convex.*

4 Stability of the coalition structure

In the previous section we presented methods for determining the social welfare maximizing coalition structure. In this section we analyze the *stability* of that structure. Specifically, can the social good be distributed so that every subgroup of agents is better off staying in the social welfare maximizing coalition structure than by separating into a new coalition (individual agents and the group of all agents are also considered subgroups here)? The *core* (C) is the solution concept that satisfies this requirement [20, 56, 32]. The core of a game is a set of *payoff configurations* $(\vec{x}, CS)$, where each $\vec{x}$ is a vector of payoffs to the agents in such a manner that no subgroup is motivated to depart from the coalition structure CS. Given payoffs according to $\vec{x}$, the value of each subgroup is no greater than the sum of the payoffs that the agents of that subgroup get under CS. Obviously, only coalition structures that maximize welfare can be stable in the sense of the core because from any other coalition structure, the group of all agents would prefer to switch to a social welfare maximizing one. The core can be formally defined as follows:

Definition 4.1 Core $C = \{(\vec{x}, CS) | \forall S \subset A, \sum_{i \in S} x_i \geq v_S^R \text{ and } \sum_{i \in A} x_i = \sum_{j \in CS} v_{S_j}^R\}$.

The core is the strongest of the classical solution concepts in coalition formation. It is often too strong: in many cases it is empty [20, 56, 32, 59]. In such games there is no way to divide the social good so that the coalition structure becomes stable: any payoff configuration is prone to deviation by some subgroup of agents. The new solution that is acquired by the deviation is again prone to deviation and so on. There will be an infinite sequence of steps from one payoff configuration to another. To avoid this, explicit mechanisms such as limits on negotiation rounds, contract costs, or some social norms need to be in place in the negotiation setting.

Another problem is that the core may include multiple payoff vectors and the agents have to agree on one of them. An often used solution is to pick the *nucleolus* which, intuitively speaking, corresponds to a payoff vector that is in the center of the set of payoff vectors in the core [20, 56, 32]. A further problem with the core is that the constraints in the definition become numerous as the number of agents increases. This is due to the combinatorial subset operator in the definition.

Now we generalize the core to allow for bounded-rational agents. The classic definition (4.1) corresponds to the special case where the agents' algorithms are complete, and computation is costless.

Definition 4.2 *The* **bounded-rational core (BRC)** *for computation unit cost* c_{comp} *is* $BRC(c_{comp}) = \{(\vec{x}, CS) | \forall S \subset A, \sum_{i \in S} x_i \geq v_S(c_{comp}) \text{ and } \sum_{i \in A} x_i = \sum_{j \in CS} v_{S_j}(c_{comp})\}$.

If the BRC is not empty, bounded-rational agents can divide the social good among themselves in a way that no subgroup is motivated to break away from CS. Sometimes the BRC is empty, but this does not always coincide with the core being empty. There are games where the BRC and the core exist, games where either one of them exists separately, and games where both are empty.

If the agents are best off working separately, the coalition structure with separate agents is stable:

Theorem 4.1 BRC in BRSUB games. *Game is BRSUB for computation unit cost* $c_{comp} \Rightarrow BRC(c_{comp}) \neq \emptyset$.

In domains that are not BRSUB, the BRC is sometimes empty. The condition $C \neq \emptyset$ can be converted into necessary and sufficient conditions on the v_S^R values in games where the grand coalition maximizes social welfare [51, 6]. We convert the condition $BRC(c_{comp}) \neq \emptyset$ into conditions on the $v_S(c_{comp})$ values analogously. Let $B_1, ..., B_p$ be distinct, nonempty, proper subsets of A. The set $\mathcal{B} = \{B_1, ..., B_p\}$ is called *balanced* if there are positive coefficients $\lambda_1, ..., \lambda_p$ such that $\forall i \in A, \sum_{\{j | i \in B_j\}} \lambda_j = 1$. A *minimal balanced set* includes no other balanced sets.

Theorem 4.2 BRC in bounded-rational grand coalition games (necessary and sufficient condition). *[Analogous to Shapley [51]]. In games where $\{A\}$ is the social welfare maximizing coalition structure for bounded-rational agents for some c_{comp}, $BRC(c_{comp}) \neq \emptyset$ iff for every minimal balanced set $\mathcal{B} = \{B_1, ..., B_p\}$, $\sum_{j=1}^{p} \lambda_j v_{B_j}(c_{comp}) \leq v_A(c_{comp})$.*

Example. In any 3-agent game where $\{A\}$ is the social welfare maximizing coalition structure for bounded-rational agents for some c_{comp}, $BRC(c_{comp}) \neq \emptyset$ iff $v_{\{1\}}(c_{comp})+v_{\{2,3\}}(c_{comp}) \leq v_{\{1,2,3\}}(c_{comp})$ and $v_{\{2\}}(c_{comp})+v_{\{1,3\}}(c_{comp}) \leq v_{\{1,2,3\}}(c_{comp})$ and $v_{\{3\}}(c_{comp})+v_{\{1,2\}}(c_{comp}) \leq v_{\{1,2,3\}}(c_{comp})$ and $v_{\{1\}}(c_{comp})+v_{\{2\}}(c_{comp})+v_{\{3\}}(c_{comp}) \leq v_{\{1,2,3\}}(c_{comp})$ and $\frac{1}{2}v_{\{1,2\}}(c_{comp}) + \frac{1}{2}v_{\{1,3\}}(c_{comp}) + \frac{1}{2}v_{\{2,3\}}(c_{comp}) \leq v_{\{1,2,3\}}(c_{comp})$. All but the last inequality are implied by the fact that $\{A\}$ is the social welfare maximizing coalition structure.

Example. In any 4-agent game where $\{A\}$ is the social welfare maximizing coalition structure for bounded-rational agents for some c_{comp}, $BRC(c_{comp}) \neq \emptyset$ iff the 41 inequalities of Table 2 hold. Constraints 1, 2, 3 and 5 correspond to partitions of A (every λ equals 1). They are thus implied by the fact that $\{A\}$ is the social welfare maximizing coalition structure.

Table 2: *Conditions for existence of the BRC in a 4-agent bounded-rational grand coalition game. The last column shows the number of constraints generated from the constraint by permuting the agents (including the presented permutation).*

Id	Constraint	#
1	$v_{\{1,2\}}(c_{comp}) + v_{\{3,4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	3
2	$v_{\{1,2,3\}}(c_{comp}) + v_{\{4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	4
3	$v_{\{1,2\}}(c_{comp}) + v_{\{3\}}(c_{comp}) + v_{\{4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	6
4	$\frac{1}{2}v_{\{1,2,3\}}(c_{comp}) + \frac{1}{2}v_{\{1,2,4\}}(c_{comp}) + \frac{1}{2}v_{\{3,4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	6
5	$v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp}) + v_{\{3\}}(c_{comp}) + v_{\{4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	1
6	$\frac{1}{2}v_{\{1,2\}}(c_{comp}) + \frac{1}{2}v_{\{1,3\}}(c_{comp}) + \frac{1}{2}v_{\{2,3\}}(c_{comp}) + v_{\{4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	4
7	$\frac{1}{2}v_{\{1,2,3\}}(c_{comp}) + \frac{1}{2}v_{\{1,4\}}(c_{comp}) + \frac{1}{2}v_{\{2,4\}}(c_{comp}) + \frac{1}{2}v_{\{3\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	12
8	$\frac{2}{3}v_{\{1,2,3\}}(c_{comp}) + \frac{1}{3}v_{\{1,4\}}(c_{comp}) + \frac{1}{3}v_{\{2,4\}}(c_{comp}) + \frac{1}{3}v_{\{3,4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	4
9	$\frac{1}{3}v_{\{1,2,3\}}(c_{comp}) + \frac{1}{3}v_{\{1,2,4\}}(c_{comp}) + \frac{1}{3}v_{\{1,3,4\}}(c_{comp}) + \frac{1}{3}v_{\{2,3,4\}}(c_{comp}) \leq v_{\{1,2,3,4\}}(c_{comp})$	1

In BRSUP games, a subset of the above inequalities suffices. Let us call a minimal balanced set *proper* if no two of its elements are disjoint.

Theorem 4.3 BRC in BRSUP games (necessary and sufficient condition). *[Analogous to Shapley [51]]. In a game that is BRSUP for some c_{comp}, $BRC(c_{comp}) \neq$*

$\emptyset$ *iff for every proper minimal balanced set* $\mathcal{B} = \{B_1, ..., B_p\}$, $\sum_{j=1}^{p} \lambda_j v_{B_j}(c_{comp}) \leq v_A(c_{comp})$. *Furthermore, this set of inequalities is minimal: no smaller set is sufficient [analogous to [6]].*

Example. In a 3-agent game that is BRSUP for some c_{comp}, $BRC(c_{comp}) \neq \emptyset$ iff $\frac{1}{2}v_{S_{\{1,2\}}}(c_{comp}) + \frac{1}{2}v_{S_{\{1,3\}}}(c_{comp}) + \frac{1}{2}v_{S_{\{2,3\}}}(c_{comp}) \leq v_{S_{\{1,2,3\}}}(c_{comp})$.

Example. In a 4-agent game that is BRSUP for some c_{comp}, $BRC(c_{comp}) \neq \emptyset$ iff the 11 conditions acquired from Table 2's constraints 4, 8, and 9 are satisfied.

Next we present conditions on the performance profiles that are *sufficient* to guarantee that the BRC exists. According to Theorem 4.1, the conditions on the performance profiles that guarantee bounded-rational subadditivity (Theorem 3.4) form one such set of conditions. The following set suffices for games where $\{A\}$ is a social welfare maximizing coalition structure for bounded-rational agents:

Theorem 4.4 BRC in bounded-rational grand coalition games (sufficient condition). *In games where* $\{A\}$ *is a social welfare maximizing coalition structure for bounded-rational agents for some* c_{comp}, [*for every minimal balanced set* $\mathcal{B} = \{B_1, ..., B_p\}$, $(\forall B \in \mathcal{B}, \forall r_B \geq 0)$ $\sum_{j=1}^{p} \lambda_j c_{B_j}(r_{B_j}) \geq c_A(\sum_{j=1}^{p} \lambda_j r_{B_j})$] $\Rightarrow BRC(c_{comp}) \neq \emptyset$.

In games where $\{A\}$ is a social welfare maximizing coalition structure for bounded-rational agents for *all* $c_{comp}(\geq 0)$, the above conditions guarantee existence of the $BRC(c_{comp})$ for *all* $c_{comp}(\geq 0)$. This would mean stability of the grand coalition for any execution platform. In BRSUP games, fewer conditions suffice:

Theorem 4.5 BRC in BRSUP games (sufficient condition). *In a game that is BRSUP for some* $c_{comp} \geq 0$, [*for every proper minimal balanced set* $\mathcal{B} = \{B_1, ..., B_p\}$, $(\forall B \in \mathcal{B}, \forall r_B \geq 0)$ $\sum_{j=1}^{p} \lambda_j c_{B_j}(r_{B_j}) \geq c_A(\sum_{j=1}^{p} \lambda_j r_{B_j})$] $\Rightarrow BRC(c_{comp}) \neq \emptyset$.

Again, if the game is BRSUP for *all* $c_{comp}(\geq 0)$, the above conditions guarantee existence of the $BRC(c_{comp})$ for *all* $c_{comp}(\geq 0)$. This would mean stability of the grand coalition for any execution platform.

Example. In a 3-agent game that is BRSUP $\forall c_{comp}$, $[(\forall r_{\{1,2\}} \geq 0, \forall r_{\{1,3\}} \geq 0, \forall r_{\{2,3\}} \geq 0), \frac{1}{2}c_{\{1,2\}}(r_{\{1,2\}}) + \frac{1}{2}c_{\{1,3\}}(r_{\{1,3\}}) + \frac{1}{2}c_{\{2,3\}}(r_{\{2,3\}}) \geq c_{\{1,2,3\}}(\frac{1}{2}r_{\{1,2\}} + \frac{1}{2}r_{\{1,3\}} + \frac{1}{2}r_{\{2,3\}})] \Rightarrow \forall c_{comp}, BRC(c_{comp}) \neq \emptyset$.

5 Experimental results

Although one can construct problems, problem instances, and algorithm performance profiles to populate any region of the Venn diagrams of coalition games (Figures 3 and 4), this does not mean that real world cases uniformly populate this space. The role of the experiments of this section is to analyze where a particular real world problem, its instances, and a sensible iterative refinement algorithm fall in the space of coalition games. Some quite surprising results appeared.

Coalition formation among bounded-rational agents was tested on the vehicle routing problem using the large-scale real world vehicle and delivery order data described earlier. The domain cost $c_S(r_S)$ for a coalition S was the sum of the route lengths of the vehicles of that coalition (while handling all of its orders) in the solution that had been reached after computation r_S. So, the rational value (v_S^R) of each coalition is defined by the tasks (delivery orders) and the resources (vehicles, depots) of the agents in the coalition. The problem instances in our example are so large that even the smallest ones are too hard to solve optimally. Therefore, rational coalition formation algorithms for vehicle routing problems [27] are unusable in this case.

To analyze a game, we ran the same algorithm on the vehicle routing problem of each subgroup of agents separately and thus acquired a performance profile for each potential coalition. The algorithm first generates an initial solution by giving each vehicle one long delivery and then, in order, giving each vehicle the delivery that can be added to its route with the least cost without violating the constraints. The second phase of the algorithm is based on iterative refinement. At each step, a delivery (chosen from a randomly ordered circular list) is removed from the routing solution and inserted back to the solution, but into the least expensive place while not violating the constraints. The drop-off location of the delivery has to be inserted after the pickup location into the same vehicle's route, but not necessarily into the same leg. We ran the refinement algorithm until no remove-insert operation enhanced the solution: a local optimum was reached. In the performance profiles we ignored the time to construct the initial solution, and only viewed how the solution cost decreased with more CPU seconds of iterative refinement, Fig. 2 left. The refinement algorithm is an anytime algorithm, but because the performance profiles are exact (as explained, they are precomputed for experimental purposes by running the base algorithm itself), the agents do not gain information from execution on that instance so far. Therefore the algorithm is equivalent to a design-to-time algorithm for our purposes.

We analyzed all of the $\binom{5}{3} = 10$ 3-agent games that can be acquired by choosing 3 of the 5 dispatch centers. There are 7 subgroups of the 3 agents: $\{1\}$, $\{2\}$, $\{3\}$, $\{1,2\}$, $\{2,3\}$, $\{3,1\}$, $\{1,2,3\}$ and 5 coalition structures: $\{\{1\}, \{2\}, \{3\}\}$, $\{\{1\}, \{2,3\}\}$, $\{\{2\}, \{1,3\}\}$, $\{\{3\}, \{1,2\}\}$, $\{\{1,2,3\}\}$. Figure 2 shows the performance profiles with agents 1, 2 and 3.

Each of our games is superadditive for rational agents because at worst, a composite coalition can use the solutions of separate coalitions. In general, a game can be non-superadditive only if the collusion process itself involves some cost, e.g., anti-trust penalties. Thus rational agents would be best off by forming the grand coalition. Surprisingly, none of the games were BRSUP for any c_{comp}, Fig. 5. For c_{comp} in the mid-range, the 3-agent games were often BRSUB (point M in Fig. 4), while in the low and high ranges (point LH), they were often neither BRSUP nor BRSUB. In some of these mixed games, for low c_{comp}, the grand coalition was the best coalition structure (point Lg). Existence of the core for rational agents is unknown for our games: the points M, LH, and Lg might really be M', LH', and Lg' in Figure 3. The BRC was non-empty in all 3-agent games for all values of c_{comp}. To summarize, in our 3-agent games, rational agents would be best off forming the possibly unstable grand coalition, while bounded-rational agents should form varying coalition structures (the grand coalition for some low values of c_{comp}), which are always stable. We also reran the experiments without the maximum route length restriction, and these results prevailed, Fig. 5.

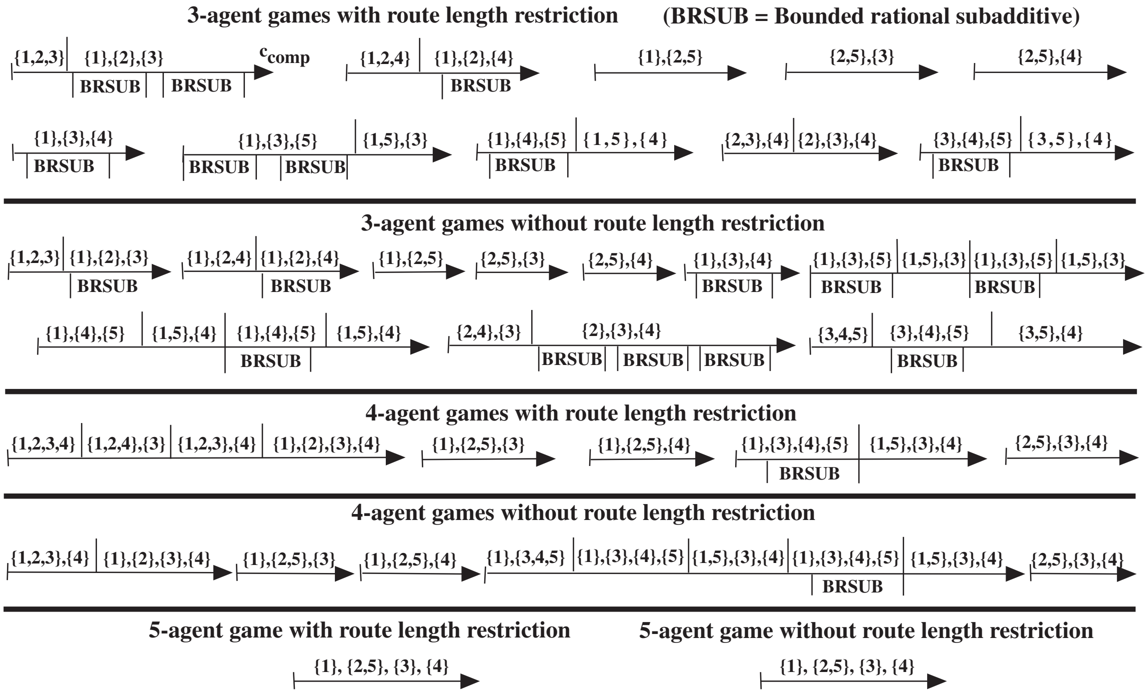

Figure 5: *Optimal coalition structure for bounded-rational agents and bounded-rational subadditivity as a function of* c_{comp}. *Tested by evaluating all possible coalition structures and bounded-rational super/subadditivity at varying points of* c_{comp} *chosen from a grid where* c_{comp} *is always incremented by 1%. Each arrow corresponds to a different subset of agents. The arrows are not drawn to scale.*

Centers 2, 3 and 5 were located near each other, while 1 and 4 were far from each other and the other centers. Centers 1, 3, 4 and 5 transported heavy low volume items, while 2 transported light voluminous items. Intuitively, adjacent centers have more potential savings from cooperation. Secondly, heavy and light items can often be beneficially combined to a load without violating the maximum load weight or maximum load volume constraint. Both with and without the route length restriction, 2 and 5 were best off by only mutually colluding for any c_{comp}. Their deliveries have considerable areal overlap due to adjacency, and the light voluminous items and heavy low volume items can be profitably joined into the weight and volume constrained vehicles. Centers 2 and 3 did not collude as much as 2 and 5 because 3's vehicles had tighter volume constraints than 5's—hindering the transport of 2's goods. No other two centers besides 2 and 5 were always best off in a 2-agent coalition independent of the third agent of the game. Relaxing the route length constraint increased collusion between the distant 2 and 4 while demoting collusion of the adjacent 2 and 3.

Next we analyzed the $\binom{5}{4} = 5$ 4-agent games and the 5-agent game with and without the route length restriction. In every game, the existence of $BRC(c_{comp})$ varied many times as a function of c_{comp} but it existed for the largest values of c_{comp}. No game was BRSUP for any c_{comp}, but some games were BRSUB for values in the medium range, Fig. 5. The grand coalition was the best coalition structure in only one of the twelve games with four and five agents. This happened for low c_{comp} in the game with agents 1, 2, 3, and 4, and the route length restriction. When this occurred, $BRC(c_{comp})$ happened to be non-empty (point Lg in Fig. 4). In all of the three, four, and five agent games, $BRC(c_{comp})$ was always nonempty when the best coalition structure was the grand coalition. To summarize, depending on c_{comp}, the games were at the points M, LH, Lg, or 45 (or M', LH', Lg', or 45'). The best coalition structure varied despite the fact that rational agents would be best off forming the grand coalition due to superadditivity. Again, whenever both agents 2 and 5 participated, they were best off by mutually colluding for all computation unit costs. In those games no other agents colluded.

Put together, the main surprising result is that although rational agents should always form the grand coalition, this is anything but obvious among bounded-rational agents. None of the vehicle routing games of our experiments—using real data and a sensible iterative refinement algorithm—exhibited bounded-rational superadditivity. The observed bounded-rational subadditivity of some of the games implies a non-empty BRC: the best coalition structure in those games is stable. Even when bounded-rational subadditivity did not hold, the BRC was often non-empty—especially for large c_{comp}.

Another interesting observation is that the presented normative theory prescribes the bounded-rational agents to choose coalition structures that agree closely with what human agents would select. The best bounded-rational coalition structures mostly agreed with our intuitions of what coalitions should form based on domain specific considerations such as adjacency of the dispatch centers and the combinability of their loads. On the other hand, these coalition structures differ significantly from those which rational agents would choose.

Finally, higher computation unit costs seem to often promote smaller coalitions than lower computation unit costs. This has a possible intuitive explanation. Each step of the refinement algorithm takes $\Theta(vd^2)$ time, where v is the number of vehicles and d is the number of deliveries. Because this is superlinear in deliveries, a larger coalition can make fewer refinement steps in a given time than the agents in partitions of that coalition can. To compensate, a refinement step of the larger coalition would need to reduce solution cost more than a refinement step of a smaller coalition. The size of the saving has to be averaged over all refinement steps in the optimal time allocation. If c_{comp} is low, more time is allocated, and small coalitions will often run out of profitable refinements. If c_{comp} is high, less time is allocated, and all coalitions will have profitable refinements, though the larger coalition will have time to make fewer of them. Thus it was not surprising that in games where the grand coalition was optimal, it was optimal for very small computation unit costs only.

Surprisingly, two agents colluding was often better than all agents working separately even for large computation unit costs. The result that higher computation unit costs promote smaller coalitions is somewhat deemphasized by our choice of not including the initial solution construction phase in the performance profiles. Shifting the performance profiles right to begin at the time when the initial solution was finished (instead of at zero) would shift the performance profiles of small coalitions less than the performance profiles of large coalitions because the initial solution construction is superlinear both in tasks and vehicles. Thus small coalitions would gain an advantage—that is most significant for large c_{comp}. If the time of initial solution generation is discarded, the best coalition structure for the highest computation unit costs depends only on the quality of the initial solutions of the different coalitions because no refinement steps are beneficial. In our experiments, coalitions $\{1,5\}$, $\{2,5\}$, and $\{3,5\}$ achieved a better initial solution cost than the sum of the initial solution costs of the two agents separately, Fig. 5 (this set of pairs prevailed even without the route length restriction).

6 Externalities among coalitions and different algorithms among agents

As is common practice [20, 52, 53, 59, 22], so far in this paper we have restricted our attention to studying coalition formation in characteristic function games (CFGs), Figure 3. In such games, the rational value of each coalition is independent of nonmembers' actions (it is given by the characteristic function v_S^R). However, in general the value of a coalition may depend on nonmembers' actions due to positive and negative externalities (interactions of the agents' solutions). Such settings can be modeled as *normal form games* (NFGs). CFGs are a strict subset of NFGs, Fig. 3.[14]

Negative externalities between a coalition and nonmembers are often caused by shared resources. Once nonmembers are using the resource to a certain extent, not enough of that resource is available to agents in the coalition to carry out the planned solution at the minimum cost. Negative externalities can also be caused by conflicting goals. In satisfying their goals, nonmembers may actually move the world further from the coalition's goal state(s) [34]. Positive externalities are often caused by partially overlapping goals. In satisfying their goals, nonmembers may actually move the world closer to the coalition's goal state(s). From there the coalition can reach its goals less expensively than it could have without the actions of nonmembers.

Let us now introduce a new domain class: *bounded-rational characteristic function game* (BRCFG), Fig. 4. In BRCFGs, the value of each coalition S is defined by the bounded-rational value $v_S(c_{comp})$. Thus, so far in this paper we have studied BRCFGs.

Externalities between domain solutions of different coalitions may exclude some problems from the class BRCFG also. In general, the rational value of a coalition may depend on the actions of nonmember agents due to positive and negative interactions of the agents' domain solutions as discussed above. Such games are NFGs, but not CFGs. For the same reason, the value of some bounded-rational coalition's domain solution—computed by a bounded-rational agent—may depend on the actions of nonmembers, and is therefore not characterized by any $v_S(c_{comp})$. Such games are not in the class BRCFG.

There is also another reason why a game may not be a BRCFG. If the agents have *different performance profiles* for a given coalition—due to different algorithms—the game may not be a BRCFG because the value of a coalition can depend on whether a nonmember agent is willing to compute the solution for the coalition (for a payment). This becomes an issue if the nonmember agent has a better algorithm than any of the coalition members. In other words, the value of a coalition may depend on the *computational actions* of nonmembers.

Games where the agents have different unit costs (c_{comp}) for computation—e.g., due to different execution platforms—are also in general not BRCFGs. Such games are analogous to games with a global c_{comp} but agents with different performance profiles. A game where agents have different computation unit costs can be modeled as a game with a uniform computation unit cost after the c_{comp}-axis of each $v_S(c_{comp})$

[14]The two are equivalent in constant-sum games with unrestricted side payments and perfect communication. In such games, the characteristic function value of a coalition is its minimax value from the normal form game [56]. A coalition's minimax value is the maximum payoff that the coalition can bring about for itself given that nonmembers choose strategies that are worst for the coalition.

function is appropriately rescaled based on the real c_{comp} of the corresponding coalition S.

There exist BRCFGs that are not CFGs. This is due to the fact that one can construct games where the domain cost of the *actual solution* (for any coalition) attained by the algorithm of a bounded-rational agent may be independent of the actions of nonmembers even though the domain cost of the *best solution* attained by a rational agent depends on the actions of nonmembers. For example, in some domains it is possible to restrict oneself to using algorithms that only consider solutions whose values are not affected by nonmembers. There also exist CFGs that are not BRCFGs. For example, the agents may have different performance profiles and therefore the bounded-rational value of a coalition may depend on whether nonmembers are willing to carry out the computation for the coalition. There is also another reason why some CFGs are not BRCFGs. The algorithms that the agents use may produce solutions whose values depend on the actions of nonmembers although the values of the optimal solutions would not.

In the distributed vehicle routing problem discussed earlier, there are no positive or negative domain solution interactions between coalitions. There are no shared resources because all of the resources—vehicles and depots—are exclusively and exhaustively distributed among the agents (and thus among coalitions). Secondly, each agent (and thus each coalition) has its own goal: delivering all of the parcels at the lowest possible cost. A coalition's handling of its deliveries is unaffected by how nonmember agents handle their deliveries. Therefore this vehicle routing problem is a CFG. For the same reason, domain solution interactions do not preclude the problem from belonging to the class BRCFG. So, if all agents have the same performance profiles—as was assumed earlier—the distributed vehicle routing games are BRCFGs.[15] Yet if the agents had different performance profiles or computation unit costs, the games would not necessarily be BRCFGs.

The core solution concept (as well as superadditivity and subadditivity) is defined via the coalition values v_S^R which are well-defined only in games where nonmembers' physical and computational actions do not affect the coalition value (CFGs). Therefore, that solution concept is not applicable for the more general games (non-CFGs). Similarly, in non-BRCFGs, the bounded-rational core (as well as bounded-rational superadditivity and bounded-rational subadditivity) is undefined. Thus other solution concepts are needed. The rest of this section discusses appropriate alternative solution concepts which are applicable to general NFGs. They also have a *strategic foundation* which guarantees that agents are motivated to adhere to their strategies: they analyze stable points in the space of agents' strategies. This differs from the core

[15]Also in many distributed scheduling domains, domain interactions do not occur unless agents share resources. On the other hand, even in toy problems such as the blocks world, positive and negative interactions often occur.

solution concept presented earlier which has a *purely axiomatic foundation*: it postulates desirable stability properties of the outcomes, but does not guarantee stability of the strategies that lead to those outcomes.[16]

One alternative solution concept is the *Nash equilibrium* [29, 24]. It guarantees stability in the sense that no agent *alone* is motivated to deviate by changing its strategy given that others do not deviate. Often the Nash equilibrium is too weak because *subgroups of agents* can deviate in a coordinated manner.

The *Strong Nash equilibrium* [1] is a solution concept for NFGs that guarantees more stability. It requires that there is no subgroup that can deviate by changing their strategies jointly in a manner that increases the payoff of all of its members given that nonmembers do not deviate from the original solution. The Strong Nash equilibrium is often too strong a solution concept: in many games no such equilibria exist.

Recently, the *Coalition-Proof Nash equilibrium* [2, 3] for NFGs has been suggested as a partial remedy to the nonexistence problem of the Strong Nash equilibrium. This solution concept requires that there is no subgroup that can make a beneficial deviation (keeping the strategies of nonmembers fixed) *in a way that the deviation itself is stable according to the same criterion.* A conceptual problem with this solution concept is that the deviation may be stable within the deviating group, but the solution concept ignores the possibility that some of the agents that deviated may prefer to deviate again with agents that did not originally deviate. Furthermore, even Coalition-Proof Nash equilibria do not exist in all NFGs. Clearly, there is room for further research on coalition formation solution concepts—even among fully rational agents.

The solution concepts presented above guarantee forms of stability for the beginning of the game. To ensure stability throughout the game, the equilibria should prevail in subtrees of the game tree also. To guarantee this, *sequential* [24] or *subgame perfect* [50, 24] refinements of the above solution concepts should be used.

The strategic solution concepts presented above provide a rigorous tool for analyzing general games (NFGs), and they also extend the strategic approach to settings where axiomatic solution concepts like the core are well-defined. New analyses of coalition formation among bounded-rational agents should extend each agent's strategy to include its computational actions and communication actions. This type of modeling of strategies, and the use of the strategic solution concepts in the space of those strategies would allow one to formulate theories of the coalition formation process—as opposed to just the outcomes—which normatively incorporate computation and communication. This is part of our current research.

[16] On the other hand, results with strategic solution concepts are specific to a given interaction protocol while core-based analyses are not (unless the coalition formation process itself affects the payoffs).

7 Related research on computational coalition formation

Coalition formation has been widely studied in game theory [20, 2, 3, 1, 56, 32], and only the most relevant concepts were presented. Many of the solution concepts for coalition formation are static. They address the question of how to divide the payoffs among agents. Some of them also address the question of which coalition structure should form. But being static in nature, they do not usually address the dynamics of the coalition formation process. This section reviews work that has addressed the dynamics.

Friend [10, 20] has developed a program that simulates a 3-agent coalition formation situation where agents can make offers, acceptances, and rejections to each other regarding coalitions and payoffs. In the model, at most one offer regarding each agent can be active at a time. A new offer makes old offers regarding that agent void. Players consider only current proposals: there is no lookahead or memory. The negotiations terminate when two agents have reach a dyad and the third one has given up. Specifically, the termination criterion is based on a local threat-counterthreat examination: an agent does not necessarily accept a new better offer if that introduces a risk of being totally excluded in the next step. The model is purely descriptive. There is no guarantee that a self-interested agent would be best off by using the specified local strategy.

Transfer schemes are another dynamic approach to coalition formation [20]. The agents stay within a given coalition structure and iteratively exchange payments in a prespecified manner. Again, there is no guarantee that a self-interested agent would be best off by using the specified local strategy: by using some other strategy, an agent may be able to drive the negotiation to a final solution that is better for the agent. Transfer schemes address the payoff distribution problem but not the optimization problem or coalition structure generation. For example, a transfer scheme for the core solution concept has been developed. This scheme alternates between two operators. In the first, an agent's payoff is incremented by its coalition's excess (value of the coalition minus the sum of the members' current payoffs) divided by the number of agents in the coalition. In the second, every agent's payoff is decremented equally, and just enough to keep the total payoff vector feasible. The method can be implemented in a largest-excess-first manner, or in a round-robin manner among agents. Both schemes converge to a payoff vector in the core if the core is nonempty, i.e., if such a stable payoff vector exists. Transfer schemes reduce the cognitive demands placed on the agents. For example in the case of the core, the agents do not need to search for a payoff vector that satisfies the numerous constraints in the definition of the core. Instead they can simply follow the transfer scheme until a payoff division in the core has been reached. Transfer schemes assume that the agents know the values of the characteristic function. On the other hand, our work addresses precisely the problem of agents not being able to solve these values due to combinatorial complexity. Transfer schemes can trivially be used in conjunction with our work. In this hybrid method, our approach would be used to determine the bounded-rational coalition values, and the transfer scheme would be used to find an appropriate payoff division given those values.

Zlotkin and Rosenschein [59] analyze coalitions among rational agents that cannot make side payments, while our agents do. Their analysis is limited to "Subadditive Task Oriented Domains" (STODs), which are a strict subset of CFGs, Fig. 3. In their work, the coalition structure generation is trivial since the agents always form the grand coalition. More specifically, one agent handles the tasks of all agents. In STODs this is optimal because STODs never exhibit diseconomies of scale. We do not assume that one agent can take care of all the agents' tasks. Unlike our work, they also assume that all agents have the same capabilities (symmetric cost functions for task sets). Their method guarantees each agent an expected value that equals its Shapley value [20, 32]. The Shapley value is a specific payoff division among agents that motivates individual agents to stay with the coalition structure and the group of all agents to stay. Unlike the core, the Shapley value does not in general motivate every subgroup of agents to stay. In a subset of STODs, "Concave Task Oriented Domains" (Fig. 3), the Shapley value also motivates every subgroup to stay, i.e., that payoff configuration is in the core.

A naive method that guarantees each agent an expected payoff equal to the Shapley value has exponential complexity in the number of agents, but Zlotkin and Rosenschein present a novel cryptographic method for achieving this with linear complexity in the number of agents. Yet each one of these linearly many problems involving the agents' tasks needs to be solved optimally. In combinatorial problems such as the vehicle routing problem presented in this paper—and the Postmen Domain of Zlotkin and Rosenschein for that matter—this is clearly intractable if the problem instances are large.

In this paper we assumed that the problem instances (tasks and resources) of all agents are common knowledge. This is somewhat unrealistic in open environments with a large number of agents. In practice it is often necessary to learn the other agents' characteristics from previous encounters. Alternatively, the agents can be made to explicitly declare their tasks and resources, but they may lie in order to gain monetarily. Rosenschein and Zlotkin have analyzed when rational agents are motivated to declare truthfully [34]. Unfortunately that work assumes only two agents and that they can optimally solve exponentially many $\mathcal{NP}$-complete problems without computation costs. Even under these assumptions, in most cases, truth-telling is not achieved. To our knowledge, the effect of bounded rationality on truthful revelation has not been studied.

Our problem is outside the domain classification of Rosenschein and Zlotkin [34], Fig. 3, because agents do not have symmetric capabilities due to heterogeneous fleets. If Rosenschein and Zlotkin's definition were extended to allow asymmetric capabilities, our domain would be in the class SOD but outside the subclass TOD. Our domain would not be a TOD because any one agent is not necessarily able to individually handle all tasks of all agents. If we further dropped the maximum route length constraint (this experiment was also presented), and restricted ourselves to domains where each center has at least one sufficient vehicle to satisfy the weight/volume constraints of any order of any center (not true in our data), then the domain would be a TOD. The following simple example shows that it would still not be a "Subadditive TOD" because the depots are geographically distributed. Let us look at a game with just two agents (A1 and A2), two delivery tasks (T1 and T2), and two identical vehicles—one for each agent. Say that the pickup site and the drop-off site of T1 are close to A1's depot, and T2's pickup and drop-off are close to A2's depot. Now say that the depots are far from each other. Thus the sum of the route lengths when A1 manages T1 and A2 manages T2 is lower than when either agent individually manages both tasks.

Ketchpel [22] presents a non-normative coalition formation method for rational agents which have different expectations of coalition values. The (computational) origin of these expectations is not addressed. His assumption of imperfect information differs from our setting where the agents have perfect information but cannot perfectly deduce. The method addresses coalition structure generation as well as payoff distribution. These two activities are handled simultaneously. Ketchpel's coalition formation algorithm runs in cubic time in the number of agents, but does not guarantee stability. His protocol is based on mutual offers. In practice it may be hard to prevent out-of-protocol offers such as multiagent offers. In our approach, if the agents' payoff vector is chosen from within the bounded-rational core, the coalition structure is stable against *all* offers. Finally, his 2-agent auction is manipulable and computationally inefficient.

Ketchpel's method is related to a contracting protocol of Sandholm [40, 46] (TRACONET) where agents construct the global solution by contracting a small number of tasks at a time, and payments are made regarding each contract before new contracts take place. An agent updates its approximate local solution after each task transfer.

In general equilibrium market mechanisms such as WALRAS [57], non-manipulative agents iterate over the allocation of resources and tasks, and payments are usually made only after a final solution has been reached. Unlike our work, general equilibrium methods are only guaranteed to work in very restricted settings. For example, an equilibrium may not exist if the domain exhibits economies of scale. General equilibrium mechanisms assume that agents view the prices as fixed—although in reality the agents affect the prices. If all agents act as price takers, and an equilibrium is reached for the market, that equilibrium is guaranteed to be in the core: no subgroup of agents is motivated to leave the market and form their own market [28]. This is not the case if agents speculate how their demands and supplies affect the market prices. Recently, Sandholm and Ygge have devised general insincere strategies that allow an agent to drive the market to an equilibrium where the agent's maximal gain from speculation materializes [48]. Speculative behavior in general equilibrium markets has recently been studied in the context of learning by Hu and Wellman [19].

Shehory and Kraus [53] analyze coalition formation among rational agents with perfect information in CFGs that are not necessarily superadditive. Their protocol guarantees that if agents follow it (this assumption makes their approach non-normative), a certain stability criterion (K-stability) is met. This requires the solution of an exponential number of optimization problems. Their other protocol guarantees a weaker form of stability (polynomial K-stability), but only requires the solution of a polynomial number of optimization problems. Unfortunately, each one of these may be intractable. Their algorithm switches from one coalition structure to another guaranteeing improvements at each step: coalition structure generation is an anytime algorithm although each domain problem is solved optimally. On the other hand, in our work, each domain problem is solved using an approximation (design-to-time) algorithm.

Shehory and Kraus [52] also present an algorithm for coalition structure generation among cooperative—social welfare maximizing, i.e., not self-interested—agents. Among such agents the payoff distribution is a non-issue and is thus not addressed. The distributed algorithm forms disjoint coalitions—which by their definition can only handle one task each—and allocates tasks to the coalitions. The complexity of the problem is reduced by limiting (possibly compromising optimality) the number of agents per coalition. The greedy algorithm guarantees that the solution is within a loose ratio bound from the best solution that is possible given the limit on the number of agents. The work assumes that the domain problem of each coalition can be solved optimally and without cost, which is not the case in the combinatorial problems of this paper. Also, in our work a coalition can handle more than one task.

Put together, most prior work on coalition formation has addressed the payoff division activity and sometimes the coalition structure generation activity, but not the optimization activity. That work has been targeted to reduce the computational complexity *in the number of agents* while assuming that the optimization problems can be solved exactly and costlessly. On the other hand, the work in this paper reduces the computational complexity *of the optimization problem of each coalition* while considering all possible coalition structures—the number of which grows exponentially in the number of agents. Future work should focus on simultaneously reducing complexity off all three activities of coalition formation—along both of these complexity generating dimensions.

8 Conclusions and future research

This paper studied settings where agents coordinate their computational actions and real world actions within each coalition but not across coalitions. A normative, application- and protocol-independent theory of coalitions in combinatorial domains was presented where the rationality of self-interested agents is bounded by computational complexity. This work is an extension of game theory which classically assumes perfect rationality: algorithms that find the optimal solution, and zero computation unit cost. On the other hand, in this paper, computational limitations were quantitatively modeled by a unit cost of computation and performance profiles of the agents' problem solving algorithms.

The algorithms used by the agents significantly impact the coalition structure that should form. From the model of bounded rationality of this paper, the social welfare maximizing coalition structure can always be determined when the performance profiles and computation unit cost are known. However, for example when agents are sent to execute at a remote host, the computation unit cost is not necessarily known in advance. To attack this problem, a sufficient condition (Theorem 3.1) on the performance profiles was presented that guarantees that any two coalitions are best off merging *for any computation unit cost*, i.e., for any execution platform. It follows that the best coalition structure would be the grand coalition. Next it was shown that the presented condition is not a necessary condition in general (Theorem 3.2) but is one if the performance profiles exhibit diminishing returns to added computation (Theorem 3.3). This is usually the case with design-to-time algorithms, and often anytime algorithms exhibit this general character also. Finally, a sufficient condition on the performance profiles was presented that guarantees that, for any computation unit cost, any possible coalition is best off by splitting up (Theorem 3.4). It follows that all agents are best off alone. This condition does not turn into a necessary condition even if the performance profiles exhibit diminishing returns to added computation (Theorem 3.5).

Stability of the payoff configuration was analyzed in terms of the *core* solution concept: the configuration is considered stable if no subgroup of agents can increase their payoff by breaking off and forming a new coalition. From the formal model of computational limitations, the stability can always be determined. There are games that have stable coalition structures for both rational and bounded-rational agents, for one but not the other, and for neither. Theorems relating the shapes of the performance profiles and the computation unit cost to stability were also presented. First, if these computation limitations are such that the agents are best off operating individually, then that coalition structure is stable (Theorem 4.1). Second, necessary and sufficient conditions on the coalition's bounded-rational values were presented that guarantee stability if the mentioned beneficial merging property (bounded-rational superadditivity) holds (Theorem 4.3), and more generally, if the best coalition structure is the grand coalition (Theorem 4.2). Finally, sufficient conditions were presented on the performance profiles that guarantee stability when the beneficial merging property holds (Theorem 4.5), or more generally, if the best coalition structure is the grand coalition (Theorem 4.4).

An application-independent domain classification was also presented for bounded-rational agents (Figure 4). Its relationship to two existing domain classifications for fully rational agents (one from game theory, and one by Rosenschein and Zlotkin, Fig. 3) was detailed. The domain classification carries with it information about the optimal coalition structure and its stability. It also incorporates domain classes where the value of a coalition is affected by the actions of nonmembers. Such games occur if agents have different optimization algorithms or if there are domain solution interactions—unlike in the vehicle routing problem and many other real world problems. Such games require different solution concept as was discussed in Section 6.

Coalitions were experimentally analyzed using real world data from a distributed vehicle routing problem. A local routing algorithm that was based on iterative refinement was used. The experiments show that the computational limitations of the agents significantly impact the coalition structure that should form as well as its stability. Although the beneficial merging property (superadditivity) holds for rational agents in almost all domains, it was surprisingly anything but obvious in practice among bounded-rational agents. None of the vehicle routing games of our experiments exhibited this property for bounded-rational agents. The optimal coalition structure for bounded-rational agents varied although rational agents should always form the grand coalition. Section 3 developed conditions on the performance profiles that guarantee that the beneficial merging property holds for bounded-rational agents. It also discussed a separate solving approach—based on a problem decomposition step—that guarantees that the base algorithm fulfills those conditions. With our sensible deterministic iterative refinement algorithm, these conditions were—somewhat surprisingly—never met. The real desideratum is not necessarily to generate algorithms that guarantee beneficial merging (and thus the superiority of the grand coalition over other coalition structures), but algorithms that provide the highest social welfare (for the best coalition structure, which need not be the grand coalition). Sometimes these goals conflict.

In the experimental games where the agents were best off separately, the coalition structures were stable as our theory predicts. Even in games were subgroups were not necessarily best off by splitting up, the coalition structures were often stable—especially for large computation unit costs.

The experiments suggest that often with superlinear iterative refinement steps, low computation unit costs promote large coalitions while high computation unit costs promote smaller ones. A plausible explanation for this phenomenon was presented.

Another interesting observation is that the presented normative theory prescribes the bounded-rational agents to choose coalition structures that agree closely with what human agents would select. The best coalition structures among bounded-rational agents mostly agreed with our intuitions of what coalitions should form based on domain specific considerations such as adjacency of the dispatch centers and the combinability of their loads. On the other hand, these coalition structures differ significantly from those which rational agents would choose.

Our model of bounded rationality is based on costly computation resources. Future work includes analyzing another model where each agent has a fixed free CPU and no more CPU time can be bought. If the domain cost increases with real time due to a dynamic environment, such agents with *bounded computational capabilities* are often best off by distributing the computation. In such settings, the problem of coordinating the computations themselves arises. On the other hand, in the *costly computation model* of this paper, it is best to allocate each coalition's computation to a single agent. The models are equivalent if the domain cost increases linearly with real time and distribution does not speed up computation. Certainly other models of bounded rationality besides these two also deserve attention.

Our current work includes analyzing the interplay of dynamic coalition formation and belief revision among bounded-rational agents [55]. Extensions of our work include generalizing the methods of this paper to agents with different and probabilistic performance profiles, as well as anytime algorithms where the performance profiles are conditioned on execution so far [43, 58, 16]. Agents with probabilistic performance profiles may want to reselect a coalition if the value of their original coalition is lower than expected—but sunk computation cost has already been incurred. Future research should also address agents that can refine solutions generated by others. Finally, we are in the process of developing interaction protocols that efficiently guide self-interested agents towards the optimal and stable (whenever possible) coalition structures—as determined by the theory developed in this paper. The goal is to construct normative methods that reduce the complexity—in the number of agents and in the size of each coalition's optimization problem—of coalition structure generation, optimization, and payoff division.

A Appendix: proofs

Proof. (Theorem 3.1). Let us analyze two arbitrary potential coalitions S and T, where $S, T \subseteq A$ and $S \cap T = \emptyset$. The conditions in the theorem state

$$\forall r_S \geq 0, \forall r_T \geq 0, c_{S \cup T}(r_S + r_T) \leq c_S(r_S) + c_T(r_T)$$

and obviously

$$\begin{aligned} &\exists r'_S, r'_T \geq 0 \text{ s.t. } c_S(r'_S) + c_{comp} \cdot r'_S + c_T(r'_T) + c_{comp} \cdot r'_T \\ &= \min_r [c_S(r) + c_{comp} \cdot r] + \min_r [c_T(r) + c_{comp} \cdot r] \end{aligned}$$

It follows that

$$\begin{aligned} &\exists r'_S, r'_T \geq 0 \text{ s.t. } c_{S \cup T}(r'_S + r'_T) + c_{comp} \cdot (r'_S + r'_T) \\ &\leq \min_r [c_S(r) + c_{comp} \cdot r] + \min_r [c_T(r) + c_{comp} \cdot r] \\ \Leftrightarrow \quad &\exists r' \geq 0 \text{ s.t. } c_{S \cup T}(r') + c_{comp} \cdot r' \leq \min_r [c_S(r) + c_{comp} \cdot r] + \min_r [c_T(r) + c_{comp} \cdot r] \\ \Leftrightarrow \quad &\min_r [c_{S \cup T}(r) + c_{comp} \cdot r] \leq \min_r [c_S(r) + c_{comp} \cdot r] + \min_r [c_T(r) + c_{comp} \cdot r] \\ \Leftrightarrow \quad &v_{S \cup T}(c_{comp}) \geq v_S(c_{comp}) + v_T(c_{comp}) \end{aligned}$$

which completes the proof. □

Proof. (Theorem 3.2). **Counterexample.** Let us analyze a 2-agent game where $A = \{1, 2\}$. Let the performance profiles of the algorithms be

$$c_{\{1\}}(r) = c_{\{2\}}(r) = \begin{cases} \frac{1}{2} - \frac{1}{2}r & \text{if } 0 \leq r \leq 1 \\ 0 & \text{if } r > 1 \end{cases} \quad \text{and } c_{\{1,2\}}(r) = \begin{cases} 1 & \text{if } 0 \leq r \leq 1 \\ 2 - r & \text{if } 1 < r \leq 2 \\ 0 & \text{if } r > 2 \end{cases}$$

Thus (see also Figure 6),

$$v_{\{1\}}(c_{comp}) = v_{\{2\}}(c_{comp}) = -\min_r [c_{\{2\}}(r) + c_{comp} \cdot r] = \begin{cases} -c_{comp} & \text{if } c_{comp} \leq \frac{1}{2} \\ -\frac{1}{2} & \text{if } c_{comp} > \frac{1}{2} \end{cases}$$

$$\text{and } v_{\{1,2\}}(c_{comp}) = -\min_r [c_{\{1,2\}}(r) + c_{comp} \cdot r] = \begin{cases} -2c_{comp} & \text{if } c_{comp} \leq \frac{1}{2} \\ -1 & \text{if } c_{comp} > \frac{1}{2} \end{cases}$$

So when $c_{comp} \leq \frac{1}{2}$,

$$v_{\{1,2\}}(c_{comp}) = -2c_{comp} = -c_{comp} + -c_{comp} = v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp})$$

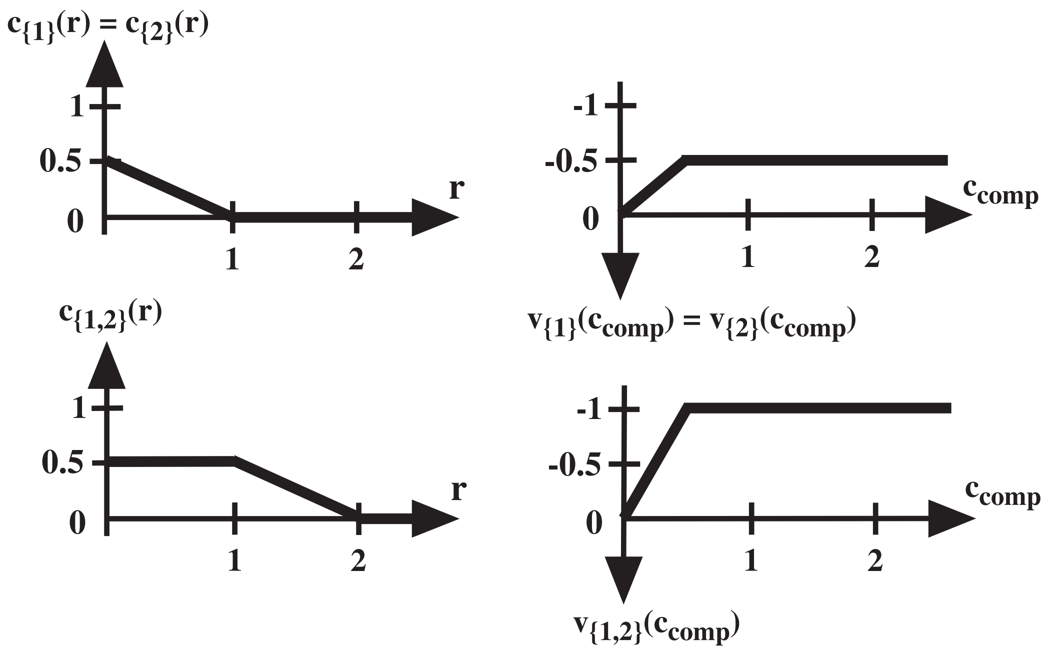

Figure 6: Performance profiles and value functions of the counterexample.

and when $c_{comp} > \frac{1}{2}$,

$$v_{\{1,2\}}(c_{comp}) = -1 = -\frac{1}{2} + -\frac{1}{2} = v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp})$$

Thus, $(\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset), v_{S \cup T}(c_{comp}) \geq v_S(c_{comp}) + v_T(c_{comp})$, i.e., the game is BRSUP for all c_{comp}. But $c_{\{1,2\}}(\frac{1}{2} + \frac{1}{2}) = 1 > \frac{1}{4} + \frac{1}{4} = c_{\{1\}}(\frac{1}{2}) + c_{\{2\}}(\frac{1}{2})$. □

The proof of Theorem 3.3 relies on the following Lemma:

Lemma A.1 *Let $f(x)$ be a decreasing, convex function. For any given x^*, $\exists c \geq 0$ s.t.*

$$\min_x [f(x) + cx] = f(x^*) + cx^*$$

Proof. (Lemma A.1). Let us define $x' = \text{argmin}_x [f(x) + cx]$. Assume—for contradiction—that $\exists x^*$ s.t. $\forall c \geq 0$,

$$\begin{aligned} \min_x [f(x) + cx] &\neq f(x^*) + cx^* \\ \Leftrightarrow f(x') + cx' &\neq f(x^*) + cx^* \end{aligned}$$

Because $f(x)$ is convex,

$$\begin{aligned} f(x^*) &\leq \frac{f(x^* - \delta) + f(x^* + \delta)}{2} \\ \Rightarrow \lim_{\delta \to 0} \frac{f(x^*) - f(x^* - \delta)}{\delta} &\leq \lim_{\delta \to 0} \frac{f(x^* + \delta) - f(x^*)}{\delta} \end{aligned}$$

Thus $c \geq 0$ is well-defined when chosen as follows:

$$\lim_{\delta \to 0} \frac{f(x^*) - f(x^* - \delta)}{\delta} \leq -c \leq \lim_{\delta \to 0} \frac{f(x^* + \delta) - f(x^*)}{\delta}$$

Now there are two cases:

Case 1: $x' < x^*$:

$$\begin{aligned} x' &< x^* \\ \Leftrightarrow \operatorname*{argmin}_x [f(x) + cx] &< x^* \\ \Leftrightarrow f(\operatorname*{argmin}_x [f(x) + cx]) + c \cdot \operatorname*{argmin}_x [f(x) + cx] &< f(x^*) + cx^* \\ \Leftrightarrow f(x') + cx' &< f(x^*) + cx^* \\ \Leftrightarrow f(x^* - \epsilon) + c \cdot (x^* - \epsilon) &< f(x^*) + cx^* \\ \Leftrightarrow f(x^*) - f(x^* - \epsilon) &> -c\epsilon \\ \Leftrightarrow \frac{f(x^*) - f(x^* - \epsilon)}{\epsilon} &> -c \\ \Rightarrow \frac{f(x^*) - f(x^* - \epsilon)}{\epsilon} &> \lim_{\delta \to 0} \frac{f(x^*) - f(x^* - \delta)}{\delta} \end{aligned}$$

This violates convexity. Contradiction.

Case 2: $x' > x^*$:

$$\begin{aligned} x' &> x^* \\ \Leftrightarrow \operatorname*{argmin}_x [f(x) + cx] &> x^* \\ \Leftrightarrow f(\operatorname*{argmin}_x [f(x) + cx]) + c \cdot \operatorname*{argmin}_x [f(x) + cx] &< f(x^*) + cx^* \\ \Leftrightarrow f(x') + cx' &< f(x^*) + cx^* \\ \Leftrightarrow f(x^* + \epsilon) + c \cdot (x^* + \epsilon) &< f(x^*) + cx^* \\ \Leftrightarrow \frac{f(x^* + \epsilon) - f(x^*)}{\epsilon} &< -c \\ \Rightarrow \frac{f(x^* + \epsilon) - f(x^*)}{\epsilon} &< \lim_{\delta \to 0} \frac{f(x^* + \delta) - f(x^*)}{\delta} \end{aligned}$$

This also violates convexity. Contradiction. Because both cases lead to a contradiction, the original assumption is false. $\square$

Proof. (Theorem 3.3). The *if*-part was proven in Theorem 3.1. Now the *only if*-part is proven.

$$
\begin{aligned}
& \text{Game is BRSUP } \forall c_{comp} \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset), \\
& v_{S \cup T}(c_{comp}) \geq v_S(c_{comp}) + v_T(c_{comp}) \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset), \\
& \min_r [c_{S \cup T}(r) + c_{comp} \cdot r] \leq \min_r [c_S(r) + c_{comp} \cdot r] + \min_r [c_T(r) + c_{comp} \cdot r] \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_T \geq 0), \\
& \min_r [c_{S \cup T}(r) + c_{comp} \cdot r] \leq c_S(r_S) + c_{comp} \cdot r_S + c_T(r_T) + c_{comp} \cdot r_T
\end{aligned}
$$

Now, by Lemma A.1, for any $r_S + r_T \geq 0$, $\exists c_{comp} \geq 0$ s.t. $\min_r [c_{S \cup T}(r) + c_{comp} \cdot r] = c_{S \cup T}(r_S + r_T) + c_{comp} \cdot (r_S + r_T)$. Thus,

$$
\begin{aligned}
& (\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_T \geq 0, \exists c_{comp} \geq 0), \\
& c_{S \cup T}(r_S + r_T) + c_{comp} \cdot (r_S + r_T) \leq c_S(r_S) + c_{comp} \cdot r_S + c_T(r_T) + c_{comp} \cdot r_T \\
\Leftrightarrow \quad & (\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_T \geq 0, \exists c_{comp} \geq 0), \\
& c_{S \cup T}(r_S + r_T) \leq c_S(r_S) + c_T(r_T) \\
\Leftrightarrow \quad & (\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_T \geq 0), \\
& c_{S \cup T}(r_S + r_T) \leq c_S(r_S) + c_T(r_T)
\end{aligned}
$$

This completes the proof. □

Proof. (Theorem 3.4).

$$
\begin{aligned}
& (\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_T \geq 0), c_{S \cup T}(r_S + r_T) > c_S(r_S) + c_T(r_T) \\
\Leftrightarrow \quad & (\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_{S \cup T} \geq 0), c_{S \cup T}(r_{S \cup T}) > c_S(r_S) + c_T(r_{S \cup T} - r_S) \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_{S \cup T} \geq 0), c_{S \cup T}(r_{S \cup T}) + c_{comp} \cdot r_{S \cup T} \\
& > c_S(r_S) + c_{comp} \cdot r_S + c_T(r_{S \cup T} - r_S) + c_{comp} \cdot (r_{S \cup T} - r_S) \\
\Rightarrow \quad & (\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S, r_{S \cup T} \geq 0), \\
& \min_r [c_{S \cup T}(r) + c_{comp} \cdot r] \\
& > c_S(r_S) + c_{comp} \cdot r_S + c_T(r_{S \cup T} - r_S) + c_{comp} \cdot (r_{S \cup T} - r_S) \\
& \geq \min_r [c_S(r) + c_{comp} \cdot r] + \min_r [c_T(r) + c_{comp} \cdot (r)] \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall S, T \subseteq A, S \cap T = \emptyset), v_{S \cup T}(c_{comp}) < v_S(c_{comp}) + v_T(c_{comp}) \\
\Leftrightarrow \quad & \text{Game is bounded-rational subadditive } \forall c_{comp}
\end{aligned}
$$

This completes the proof. □

Proof. (Theorem 3.5). It suffices to show an example where all performance profiles are decreasing and convex, the game is BRSUB for all c_{comp}, and the condition $[(\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S \geq 0, \forall r_T \geq 0), c_{S \cup T}(r_S + r_T) > c_S(r_S) + c_T(r_T)]$ fails to hold. Let us analyze a 2-agent game where $A = \{1, 2\}$. Let the decreasing and convex performance profiles of the algorithms be

$$c_{\{1\}}(r) = \begin{cases} 0.6 - 0.6r & \text{if } 0 \leq r \leq 1 \\ 0 & \text{if } r > 1 \end{cases}$$

$$c_{\{2\}}(r) = \begin{cases} 0.4 - 0.4r & \text{if } 0 \leq r \leq 1 \\ 0 & \text{if } r > 1 \end{cases}$$

$$c_{\{1,2\}}(r) = \begin{cases} 1.01 - 0.5r & \text{if } 0 \leq r \leq 2 \\ 0.01 & \text{if } r > 2. \end{cases}$$

Now, $c_{\{1,2\}}(0+1) = 0.51 < 0.6 + 0 = c_{\{1\}}(0) + c_{\{2\}}(1)$. This violates $[(\forall S, T \subseteq A, S \cap T = \emptyset, \forall r_S \geq 0, \forall r_T \geq 0), c_{S \cup T}(r_S + r_T) > c_S(r_S) + c_T(r_T)]$.

What remains to be shown is that the game is BRSUB for all c_{comp}.

Case 1: $0 \leq c_{comp} < 0.4$: $v_{\{1\}}(c_{comp}) = -c_{comp}, v_{\{2\}}(c_{comp}) = -c_{comp}, v_{\{1,2\}}(c_{comp}) = -2c_{comp} - 0.01$. So, $v_{\{1,2\}}(c_{comp}) < v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp})$.

Case 2: $0.4 \leq c_{comp} < 0.5$: $v_{\{1\}}(c_{comp}) = -c_{comp}, v_{\{2\}}(c_{comp}) = -0.4, v_{\{1,2\}}(c_{comp}) = -2c_{comp} - 0.01$. So, $v_{\{1,2\}}(c_{comp}) < v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp})$.

Case 3: $0.5 \leq c_{comp} < 0.6$: $v_{\{1\}}(c_{comp}) = -c_{comp}, v_{\{2\}}(c_{comp}) = -0.4, v_{\{1,2\}}(c_{comp}) = -1.01$. So, $v_{\{1,2\}}(c_{comp}) < v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp})$.

Case 4: $0.6 \leq c_{comp}$: $v_{\{1\}}(c_{comp}) = -0.6, v_{\{2\}}(c_{comp}) = -0.4, v_{\{1,2\}}(c_{comp}) = -1.01$. So, $v_{\{1,2\}}(c_{comp}) < v_{\{1\}}(c_{comp}) + v_{\{2\}}(c_{comp})$. □

Proof. (Theorem 4.1). Let a game be bounded-rational subadditive for c_{comp}, i.e., $(\forall S, T \subseteq A, S \cap T = \emptyset), v_{S \cup T}(c_{comp}) < v_S(c_{comp}) + v_T(c_{comp})$. Let us study the coalition structure $CS = \{\{1\}, \{2\}, ..., \{|A|\}\}$ which clearly maximizes social welfare. Let us choose $\vec{x}$ s.t. $\forall i \in A$, $x_i = v_{\{i\}}(c_{comp})$. Now,

$$\sum_{i \in A} x_i = \sum_{i \in A} v_{\{i\}}(c_{comp}) = \sum_{j \in CS} v_{S_j}(c_{comp})$$

and

$$\forall S \subset A, \sum_{i \in S} x_i = \sum_{i \in S} v_{\{i\}}(c_{comp}) \geq v_S(c_{comp})$$

Thus $\vec{x} \in BRC(c_{comp})$ which implies $BRC(c_{comp}) \neq \emptyset$. □

Proof. (Theorem 4.2). Shapley [51] proved the following fact (his Theorem 2) for rational agents. In games where $\{A\}$ is a social welfare maximizing coalition structure for rational agents, $C \neq \emptyset$ iff for every minimal balanced set $\mathcal{B} = \{B_1, ..., B_p\}, \sum_{j=1}^{p} \lambda_j v_{B_j}^R \leq v_A^R$. Theorem 4.2 follows by analogy. □

Proof. (Theorem 4.3). Shapley [51] proved the following fact (his Theorem 3) for rational agents. In a superadditive game, $C \neq \emptyset$ iff for every proper minimal balanced set $\mathcal{B} = \{B_1, ..., B_p\}, \sum_{j=1}^{p} \lambda_j v_{B_j}^R \leq v_A^R$. Charnes and Kortanek [6] proved that this set of inequalities is minimal. Theorem 4.3 follows by analogy. $\Box$

Proof. (Theorem 4.4). Let us analyze an arbitrary minimal balanced set $\mathcal{B} = \{B_1, ..., B_p\}$.

$$
\begin{aligned}
& (\forall B \in \mathcal{B}, \forall r_B \geq 0), \sum_{j=1}^{p} \lambda_j c_{B_j}(r_{B_j}) \geq c_A(\sum_{j=1}^{p} \lambda_j r_{B_j}) \\
\Rightarrow \quad & (\forall c_{comp}, \forall B \in \mathcal{B}, \forall r_B \geq 0, \exists r_A \geq 0), \\
& \sum_{j=1}^{p} \lambda_j c_{B_j}(r_{B_j}) + c_{comp} \cdot (-r_A + \sum_{j=1}^{p} \lambda_j r_{B_j}) \geq c_A(r_A) \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall B \in \mathcal{B}, \forall r_B \geq 0, \exists r_A \geq 0), \\
& \sum_{j=1}^{p} \lambda_j c_{B_j}(r_{B_j}) + c_{comp} \cdot \sum_{j=1}^{p} \lambda_j r_{B_j} \geq c_A(r_A) + c_{comp} \cdot r_A \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall B \in \mathcal{B}, \forall r_B \geq 0), \\
& \sum_{j=1}^{p} \lambda_j c_{B_j}(r_{B_j}) + c_{comp} \cdot \sum_{j=1}^{p} \lambda_j r_{B_j} \geq \min_r [c_A(r) + c_{comp} \cdot r] \\
\Leftrightarrow \quad & (\forall c_{comp}, \forall B \in \mathcal{B}, \forall r_B \geq 0), \\
& \sum_{j=1}^{p} \lambda_j [c_{B_j}(r_{B_j}) + c_{comp} \cdot r_{B_j}] \geq \min_r [c_A(r) + c_{comp} \cdot r] \\
\Leftrightarrow \quad & (\forall c_{comp}), \sum_{j=1}^{p} \lambda_j \min_{r'_{B_j}} [c_{B_j}(r'_{B_j}) + c_{comp} \cdot r'_{B_j}] \geq \min_r [c_A(r) + c_{comp} \cdot r] \\
\Leftrightarrow \quad & (\forall c_{comp}), \sum_{j=1}^{p} \lambda_j v_{B_j}(c_{comp}) \leq v_A(c_{comp})
\end{aligned}
$$

Since this holds for an arbitrary minimal balanced set, it has to hold for every minimal balanced set. Thus, by Theorem 4.2, $BRC(c_{comp}) \neq \emptyset$. $\Box$

Proof. (Theorem 4.5). Analogous to the proof of Theorem 4.4, except that now an arbitrary *proper* minimal balanced set is considered. Furthermore, the reference to Theorem 4.2 should be changed to a reference to Theorem 4.3. $\Box$

References

[1] R. Aumann. Acceptable points in general cooperative n-person games. volume IV

of *Contributions to the Theory of Games*. Princeton University Press, 1959.

[2] B. D. Bernheim, B. Peleg, and M. D. Whinston. Coalition-proof Nash equilibria: I concepts. *Journal of Economic Theory*, 42(1):1–12, June 1987.

[3] B. D. Bernheim and M. D. Whinston. Coalition-proof Nash equilibria: II applications. *Journal of Economic Theory*, 42(1):13–29, June 1987.

[4] M. Boddy and T. Dean. Deliberation scheduling for problem solving in time-constrained environments. *Artificial Intelligence*, 67:245–285, 1994.

[5] A. H. Bond and L. Gasser. *Readings in Distributed Artificial Intelligence*. Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[6] A. Charnes and K. O. Kortanek. On balanced sets, cores, and linear programming. Technical Report 12, Cornell Univ., Dept. of Industrial Eng. and Operations Res., Ithaca, NY, 1966.

[7] E. Durfee, V. Lesser, and D. Corkill. Cooperative distributed problem solving. In A. Barr, P. Cohen, and E. Feigenbaum, editors, *The Handbook of Artificial Intelligence*, volume IV. Addison Wesley, 1989.

[8] T. Finin, R. Fritzson, and D. McKay. A language and protocol to support intelligent agent interoperability. In *Proc. of the CE & CALS Washington '92 Conference*, June 1992.

[9] K. Fischer, J. Müller, M. Pischel, and D. Schier. A model for cooperative transportation scheduling. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 109–116, San Francisco, CA, June 1995.

[10] K. E. Friend. *An information processing approach to small group interaction in a coalition formation game*. PhD thesis, Carnegie-Mellon University, 1973.

[11] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

[12] A. Garvey and V. Lesser. Design-to-time real-time scheduling. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(6):1491–1502, 1993.

[13] A. Garvey and V. Lesser. A survey of research in deliberative real-time artificial intelligence. *Real-Time Systems*, 6:317–347, 1994.

[14] General Magic, Inc. Telescript technology: The foundation for the electronic marketplace, 1994. White paper.

[15] I. Good. Twenty-seven principles of rationality. In V. Godambe and D. Sprott, editors, *Foundations of Statistical Inference.* Toronto: Holt, Rinehart, Winston, 1971.

[16] E. A. Hansen and S. Zilberstein. Monitoring the progress of anytime problem-solving. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1229–1234, Portland, OR, Aug. 1996.

[17] E. Horvitz. Reasoning about beliefs and actions under computational resource constraints. In *Proceedings of Third Workshop on Uncertainty in Artificial Intelligence*, pages 429–444, Seattle, Washington, July 1987. American Association for Artificial Intelligence. Also in L. Kanal, T. Levitt, and J. Lemmer, ed., *Uncertainty in Artificial Intelligence 3,* Elsevier, 1989, pps. 301-324.

[18] N. Howard. *Paradoxes of Rationality: Theory of Metagames and Political Behavior.* The MIT Press, 1971.

[19] J. Hu and M. P. Wellman. Self-fulfilling bias in multiagent learning. In *Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS-96)*, pages 118–125, Keihanna Plaza, Kyoto, Japan, Dec. 1996.

[20] J. P. Kahan and A. Rapoport. *Theories of Coalition Formation.* Lawrence Erlbaum Associates Publishers, 1984.

[21] R. Kalakota and A. B. Whinston. *Frontiers of Electronic Commerce.* Addison-Wesley Publishing Company, Inc, 1996.

[22] S. Ketchpel. Forming coalitions in the face of uncertain rewards. In *Proceedings of the National Conference on Artificial Intelligence*, pages 414–419, Seattle, WA, July 1994.

[23] S. Kraus, J. Wilkenfeld, and G. Zlotkin. Multiagent negotiation under time constraints. *Artificial Intelligence*, 75:297–345, 1995.

[24] D. M. Kreps. *A Course in Microeconomic Theory.* Princeton University Press, 1990.

[25] S. Lin and B. W. Kernighan. An effective heuristic procedure for the traveling salesman problem. *Operations Research*, 21:498–516, 1971.

[26] S. Linnainmaa, O. Jokinen, T. Sandholm, and A. Vepsäläinen. Advanced computer supported vehicle routing for heavy transports. In *Finnish Artificial Intelligence Conference (STeP-92), New Directions in Artificial Intelligence*, volume 3, pages 163–172, Espoo, Finland, 1992.

[27] M. G. Lundgren, K. Jörnsten, and P. Värbrand. On the nucleolus of the basic vehicle routing game. Technical Report 1992-26, Linköping Univ., Dept. of Mathematics, Sweden, 1992.

[28] A. Mas-Colell, M. Whinston, and J. R. Green. *Microeconomic Theory.* Oxford University Press, 1995.

[29] J. Nash. Equilibrium points in n-person games. *Proc. of the National Academy of Sciences*, 36:48–49, 1950.

[30] Office of Technology Assesment. Electronic enterprises: Looking to the future, 1994.

[31] C. H. Papadimitriou and M. Yannakakis. On complexity as bounded rationality. In *STOC-94*, pages 726–733, 1994.

[32] H. Raiffa. *The Art and Science of Negotiation.* Harvard Univ. Press, Cambridge, Mass., 1982.

[33] E. Rasmusen. *Games and Information.* Basil Blackwell, 1989.

[34] J. S. Rosenschein and G. Zlotkin. *Rules of Encounter.* MIT Press, 1994.

[35] S. Russell and D. Subramanian. Provably bounded-optimal agents. *Journal of Artificial Intelligence Research*, 1:1–36, 1995.

[36] S. Russell and E. Wefald. *Do the right thing: Studies in Limited Rationality.* The MIT Press, 1991.

[37] T. W. Sandholm. A strategy for decreasing the total transportation costs among area-distributed transportation centers. In *Nordic Operations Analysis in Cooperation (NOAS-91): OR in Business*, Turku School of Economics, Finland, 1991.

[38] T. W. Sandholm. Automatic cooperation of area-distributed dispatch centers in vehicle routing. In *International Conference on Artificial Intelligence Applications in Transportation Engineering*, pages 449–467, San Buenaventura, CA, 1992.

[39] T. W. Sandholm. A bargaining network for intelligent agents. In *Finnish Artificial Intelligence Conference (STeP-92), New Directions in Artificial Intelligence*, volume 3, pages 173–181, Espoo, Finland, 1992.

[40] T. W. Sandholm. An implementation of the contract net protocol based on marginal cost calculations. In *Proceedings of the National Conference on Artificial Intelligence*, pages 256–262, Washington, D.C., July 1993.

[41] T. W. Sandholm. Limitations of the Vickrey auction in computational multiagent systems. In *Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS-96)*, pages 299–306, Keihanna Plaza, Kyoto, Japan, Dec. 1996.

[42] T. W. Sandholm. *Negotiation among Self-Interested Computationally Limited Agents*. PhD thesis, University of Massachusetts, Amherst, 1996.

[43] T. W. Sandholm and V. R. Lesser. Utility-based termination of anytime algorithms. In *ECAI Workshop on Decision Theory for DAI Applications*, pages 88–99, Amsterdam, The Netherlands, 1994. Extended version: Univ. of Mass. at Amherst, Comp. Sci. Tech. Report 94-54.

[44] T. W. Sandholm and V. R. Lesser. Coalition formation among bounded rational agents. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 662–669, Montreal, Canada, Aug. 1995.

[45] T. W. Sandholm and V. R. Lesser. Equilibrium analysis of the possibilities of unenforced exchange in multiagent systems. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 694–701, Montreal, Canada, Aug. 1995.

[46] T. W. Sandholm and V. R. Lesser. Issues in automated negotiation and electronic commerce: Extending the contract net framework. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 328–335, San Francisco, CA, June 1995.

[47] T. W. Sandholm and V. R. Lesser. Advantages of a leveled commitment contracting protocol. In *Proceedings of the National Conference on Artificial Intelligence*, pages 126–133, Portland, OR, Aug. 1996. Extended version appeared as University of Massachusetts at Amherst, Computer Science Department technical report 95-72.

[48] T. W. Sandholm and F. Ygge. On the gains and losses of speculation in equilibrium markets. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, Nagoya, Japan, Aug. 1997.

[49] K. Schweers Cook and M. Levi. *The Limits of Rationality*. University of Chicago Press, 1990.

[50] R. Selten. Spieltheoretische behandlung eines oligopolmodells mit nachfrageträgheit. *Zeitschrift für die gesamte Staatswissenschaft*, 12:301–324, 1965.

[51] L. S. Shapley. On balanced sets and cores. *Naval Research Logistics Quarterly*, 14:453–460, 1967.

[52] O. Shehory and S. Kraus. Task allocation via coalition formation among autonomous agents. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 655–661, Montreal, Canada, Aug. 1995.

[53] O. Shehory and S. Kraus. A kernel-oriented model for coalition-formation in general environments: Implemetation and results. In *Proceedings of the National Conference on Artificial Intelligence*, pages 134–140, Portland, OR, Aug. 1996.

[54] H. A. Simon. *Models of bounded rationality*, volume 2. MIT Press, 1982.

[55] F. Tohmé and T. W. Sandholm. Coalition formation processes with belief revision among bounded rational self-interested agents. In *IJCAI Workshop on Social Interaction and Communityware*, Nagoya, Japan, Aug. 1997.

[56] W. J. van der Linden and A. Verbeek. Coalition formation: A game-theoretic approach. In H. A. M. Wilke, editor, *Coalition Formation*, volume 24 of *Advances in Psychology*. North Holland, 1985.

[57] M. Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research*, 1:1–23, 1993.

[58] S. Zilberstein and S. Russell. Optimal composition of real-time systems. *Artificial Intelligence*, 82(1–2):181–213, 1996.

[59] G. Zlotkin and J. S. Rosenschein. Coalition, cryptography and stability: Mechanisms for coalition formation in task oriented domains. In *Proceedings of the National Conference on Artificial Intelligence*, pages 432–437, Seattle, WA, July 1994.