Easy and Hard Coalition Resource Game Formation Problems - A Parameterized Complexity Analysis

Tammar Shrot, Yonatan Aumann, and Sarit Kraus
Department of Computer Science
Bar Ilan University
Ramat–Gan, Israel 52900
{machnet, aumann, sarit}@cs.biu.ac.il

ABSTRACT

Coalition formation is a key topic in multi–agent systems (MAS). Coalitions enable agents to achieve goals that they may not have been able to achieve independently, and encourages resource sharing among agents with different goals.

A range of previous studies have found that problems in coalitional games tend to be computationally complex. However, such hardness results consider the entire input as one, ignoring any structural information on the instances. In the case of coalition formation problems, this bundles together several distinct elements of the input, e.g. the agent set, the goal set, the resources, etc. In this paper we reexamine the complexity of coalition formation problems in the coalition resources game model, *as a function of their distinct input elements*, using the theory of *parameterized complexity*. The analysis shows that not all parts of the input are created equal, and that many instances of the problem are actually tractable. We show that the problems are $\mathcal{FPT}$ in the number of goals, implying that if the number of goals is bounded then an efficient algorithm is available. Similarly, the problems are $\mathcal{FPT}$ in the combination of the number of agents and resources, again implying that if these parameters are bounded, then an efficient algorithm is available. On the other hand, the problems are *para-$\mathcal{NP}$ hard* in the number of resources, implying that even if we bound the number of resources the problems (probably) remain hard. Additionally, we show that most problems are $\mathcal{W}[1]$-hard in the size of the coalition of interest, indicating that there is (probably) no algorithm polynomial in all but the coalition size. The exact definitions of the *parameterized complexity* notions $\mathcal{FPT}$, Para-$\mathcal{NP}$ and $\mathcal{W}[1]$ are provided herein.

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity;
I.2.11[**Artificial Intelligence**]: Distributed Artificial Intelligence — *Multiagent Systems*

General Terms

Algorithms, Theory

Cite as: Easy and Hard Coalition Resource Game Formation Problems - A Parameterized Complexity Analysis, Tammar Shrot, Yonatan Aumann, Sarit Kraus, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 433–440

Keywords

Coalition problems, Parameterized complexity

1. INTRODUCTION

In multi–agent systems (MAS), where each agent has limited resources, the formation of coalitions of agents is a very powerful tool [1, 14, 15]. Coalitions enable agents to accomplish goals they may not have been able to accomplish independently. As such, understanding and predicting the dynamics of coalitions formation, e.g. which coalitions are more beneficial and/or more likely to emerge, is a question of considerable interest in multi-agent settings. Unfortunately, a range of previous studies have shown that many of these problems are computationally complex ([12, 16, 17]). Nonetheless, as noted by Garey and Johnson [8], hardness results, such as $\mathcal{NP}$–completeness, should merely constitute the beginning of the research. $\mathcal{NP}$-hardness indicates that a general solution for all instances of the problem most probably does not exist. Still, efficient solutions for important sub-classes may well exist.

$\mathcal{NP}$-hardness results consider the entire input as one, ignoring any structural information on the problem. In the case of coalition formation problems, this bundles together several distinct elements of the input, e.g. the agent set, the goal set, the resources, etc. The $\mathcal{NP}$-hardness of coalition formation tells us that the complexity of problem, as a function of all its components *combined* is (probably) high. However, it is important to understand what is the source of this complexity; do all input parts contribute equally? What if we bound one of them, say the goal set, can the problem then be solved efficiently in terms of the other elements? Such questions are of considerable practical importance, as in many real-world applications some parts of the input may be known to be small. For example, we are witnessing the advent of small transaction commerce on the Internet for purchasing goods, information, and communication bandwidth [9]. In such domains there are many different agents but the number of different goals is limited. Therefore, a coalition formation algorithm that is very efficient in terms of the number of agents, yet not efficient in the number of goals, may not only be acceptable, but also preferable in practical terms.

The formal model for coalition formation we use in this work is the *Coalition Resource Game* (CRG) framework introduced in [17]. In this framework each agent has a set of *goals* and a fixed endowment of *resources.* For each goal, there is a specific profile of resources (i.e. types and amounts) required to fulfil this goal. Agents can form coalitions, whereby they combine their endowed resources to fulfil their respective goals. A coalition is *successful* if its combined resources are sufficient to fulfil its members' respective goals. With this formulation, given a specific instances of the problem, many questions can be posed: is a particular coalition successful? is it maximally so? and more. As stated earlier, most these problems were found to be $\mathcal{NP}$-hard.

In this paper we re-examine the complexity of these problem as a function of their distinct input elements: the number of agents (Ag), the number of goals (G), the number of resources (R), and the size of the specific coalition of interest ($|C|$). We do so using the theory of *parameterized complexity* [4, 7]. Parameterized complexity provides us with the necessary tools to investigate the relative contributions of the different input elements to the complexity of the problem. (A brief introduction to the basic concepts of parameterized complexity is provided in Section 2.2.) We show that the problems are $\mathcal{FPT}$ in the number of goals, $|G|$, roughly meaning that if $|G|$ is bounded, then an efficient algorithm is available, as a function of the rest of the input (the number of agents and resources). Similarly, the problems are $\mathcal{FPT}$ in $|Ag| + |R|$, again roughly meaning that if this parameter is bounded, then an efficient algorithm is available (as a function of the number of goals). On the other hand, the problems are para-$\mathcal{NP}$ hard in $|R|$, roughly meaning that if we bound the number of resources alone, then the problem remains hard. Additionally, in most cases, the problems are $\mathcal{W}[1]$-hard in the size of the coalition of interest, roughly meaning the that if we consider a small specific coalition C, then still no efficient algorithm is available, even if we allow inefficiencies in $|C|$. The exact definitions of the notions $\mathcal{FPT}$, Para-$\mathcal{NP}$ and $\mathcal{W}[1]$ are provided in Section 2.2.

The rest of this paper is organized as follows. In the next section we review the formal model of the *Coalition Resource Game* (CRG) framework, and the theoretical fundamentals of the *parameterized complexity* paradigm. In Section 3 we provide the parameterized analysis of several key coalition formation problems, and prove our main results – both positive and negative. Finally, Section 4 provides a discussion of these results followed by open problems and future directions.

2. PROBLEM DEFINITION

2.1 The CRG Model for Multi–Agent Systems

The framework we use to model coalitions is the CRG model introduced in [17], defined as follows. The model contains a non–empty, finite set $Ag = \{a_1, \ldots, a_n\}$ of *agents*. A *coalition*, typically denoted by C, is simply a set of agents, i.e. a subset of Ag. The *grand coalition* is the set of all agents, Ag. There is also a finite set of *goals*. Each agent $i \in Ag$ is associated with a subset G_i of the goals. Agent i is *satisfied* if *at least one* member of G_i is achieved, and *unsatisfied* otherwise.

Achieving the goals requires the expenditure of *resources*, drawn from the total set of resource types R. Achieving different goals may require different quantities of each resource type. The quantity $\mathbf{req}(g, r)$ denotes the amount of resource r required to achieve goal g. It is assumed that $\mathbf{req}(g, r)$ is a natural number. Each agent is *endowed* certain amounts of some or all of the resource types. The quantity $\mathbf{en}(i, r)$ denotes the amount of resource r endowed to agent i. Again, it is assumed that $\mathbf{en}(i, r)$ is a natural number.

Combining these components, we attain that a *coalitional resource game* Γ is an $(n + 5)$–tuple:

$$\Gamma = \langle Ag, G, R, G_1, \ldots, G_n, \mathbf{en}, \mathbf{req} \rangle$$

where:

- $Ag = \{a_1, \ldots, a_n\}$ is a set of *agents*;
- $G = \{g_1, \ldots, g_m\}$ is a set of *possible goals*;
- $R = \{r_1, \ldots, r_t\}$ is a set of *resources*;
- for each $i \in Ag$, $G_i \subseteq G$ is a set of goals, such that any of the goals in G_i would satisfy i – but i is indifferent between the members of G_i;
- $\mathbf{en} : Ag \times R \to \mathbb{N}$ is an *endowment function*; and
- $\mathbf{req} : G \times R \to \mathbb{N}$ is a *requirement function*.

The endowment function $\mathbf{en}$ is extended to coalitions, by summing up the endowment of its members:

$$\mathbf{en}(C, r) = \sum_{i \in C} \mathbf{en}(i, r)$$

Similarly, the requirements function $\mathbf{req}$ is extended to sets of goals, by summing up the requirements of its members:

$$\mathbf{req}(G', r) = \sum_{g \in G'} \mathbf{req}(g, r)$$

A set of goals G' *satisfies* agent i if $G' \cap G_i \neq \emptyset$, and satisfies coalition C if it satisfies every member of C. A set of goals G' is *feasible* for coalition C if that coalition is endowed with sufficient resources to achieve all the goals in G', i.e. for all $r \in R$, $req(G', r) \leq en(C, r)$. Finally, we say that a coalition C is *successful* if there exists a set of goals G' that satisfies it and is feasible for it.

For the sake of convenience, for a set of goals G', we denote $succ(G')$ to be the set of agents that would be satisfied if all goals in G' where achieved:

$$succ(G') = \{i \in Ag \ : G \cap G_i \neq \emptyset\}.$$

The CRG model is a very straightforward model, yet it can model many real-world situations. For example, the virtual organizations problem (see [2]). A virtual organization is a temporary alliance of organizations that come together to share skills and resources in order to better respond to business opportunities. Each organization has its own resources and needs to accomplish a certain target for its clients; the target can be achieved in several ways (goals). The organizations share their resources in order to accomplish their targets.

Another example is voting. Consider a voting domain where decisions are made based on the choices made by the voters, and where certain agents may affect how these voters vote. Each agent affects a certain subset of the voters (resources), and a coalition of agents may affect all the voters that can be affected by the members of the coalition. Different goals may require different number of voters.

2.1.1 Coalition Formation Problems

We consider four specific problems regarding coalition formations:

SUCCESSFUL COALITION (SC)
Instance: A CRG Γ, and a coalition C.
Question: Is C successful?

This problem was introduced in [16] as the most fundamental question that could be asked on coalitions. It was proven to be $\mathcal{NP}$–complete in [17].

EXISTENCE OF SUCCESSFUL COALITION OF SIZE k (ESCK)
Instance: A CRG Γ, and a number k.
Question: Does there exists a successful coalition of size (exactly) k?

This problem was not considered in [17], but we believe that it is a very interesting question in the CRG model. We found this problem to be $\mathcal{NP}$–hard by reduction from CLIQUE (Theorem 3.5).

MAXIMAL COALITION (MAXC)
Instance: A CRG Γ, and a coalition C
Question: Is every (proper) superset of C not satisfiable?

This problem was found to be co–$\mathcal{NP}$–complete in [17].

Note that MAXIMAL COALITION does not require that C, the coalition in question, be satisfiable. The following problem does:

MAXIMAL SUCCESSFUL COALITION (MAXSC)
Instance: A CRG Γ, and a coalition C
Question: Is C maximally successful (i.e. C is successful and every proper superset thereof not successful)?

This problem was found to be $\mathcal{D}^p$–complete in [17].

2.2 Parameterized Complexity

We now provide a brief introduction to the key relevant concepts from the theory of parameterized complexity. The definitions in this section are taken from [7] and [3].

The core idea of parameterized complexity is to single out a specific part of the input as the *parameter* and ask whether the problem admits an algorithm that is efficient in all but the parameter. In most cases the parameter is simply one of the elements of the input (e.g. the size of the goal set), but it can actually be any computable function of the input:

DEFINITION 2.1. *Let Σ be a finite alphabet.*

1. *A* **parametrization** *of Σ^* is a mapping $\kappa\colon \Sigma^* \to \mathbb{N}$ that is polynomial time computable.*
2. *A* **parameterized problem** *(over Σ) is a pair (Q, κ) consisting of a set $Q \subseteq \Sigma^*$ of strings over Σ and a parametrization κ of Σ^**

As stated, given a parameterized problem we seek an algorithm that is efficient in all but the parameter. This is captured by the notion of *fixed parameter tractability*, as follows:

DEFINITION 2.2. *A parameterized problem (Q, κ) is* fixed–parameter tractable *($\mathcal{FPT}$) if there exist an algorithm $\mathbb{A}$, a constant α, and a computable function f, such that $\mathbb{A}$ decides Q in time $f(\kappa(x))|x|^{\alpha}$.*

Thus, while the fixed-parameter notion allows inefficiency in the parameter $\kappa(x)$, by mean of the function f, it requires polynomial complexity in all the rest of the input. In particular, a problem that is $\mathcal{FPT}$ is tractable for any bounded parameter value.

While the core aim of parameterized complexity is to identify problems that are fixed-parameter tractable, it has also developed an extensive complexity theory, allowing to prove hardness results, e.g. that certain problems are (most probably) *not* $\mathcal{FPT}$. To this end, several parameterized complexity classes have been defined. Two of these classes are the class $\mathcal{W}[1]$ and the class *para-*$\mathcal{NP}$. We will formally define these classes shortly, but the important point to know is that there is strong evidence to believe that both classes are not contained in $\mathcal{FPT}$ (much like $\mathcal{NP}$ is probably not contained in $\mathcal{P}$). Thus, $\mathcal{W}[1]$-hard and para-$\mathcal{NP}$-hard problems are most probably not fixed-parameter tractable.

The class $\mathcal{W}[1]$ can be defined by its core complete problem, defined as follows.

SHORT NONDETERMINISTIC TURING MACHINE COMPUTATION
Instance: A single-tape, single-head nondeterministic Turing machine M, a word x, and a positive integer k.
Question: Is there a computation of M on input x that reaches the accepting state in at most k steps?
Parameter: k.

Note that this definition is analogous to that of $\mathcal{NP}$, with the addition of the parameter k.

DEFINITION 2.3. *The class $\mathcal{W}[1]$ contains all parameterized problems $\mathcal{FPT}$-reducible (defined hereunder) to Short-Nondeterministic-Turing-Machine-Computation.*

The class para-$\mathcal{NP}$ is defined as follows.

DEFINITION 2.4. *A parameterized problem (Q, κ) is in para-$\mathcal{NP}$ if there exists a* **non***-deterministic Turing machine M, constant α and an arbitrary computable function f, such that for any input x, M decides if $x \in Q$ in time $\leq |x|^{\alpha} f(\kappa(x))$.*

Establishing hardness results most frequently requires *reductions.* In parameterized complexity, we use $\mathcal{FPT}$-reduction, defined as follows:

DEFINITION 2.5. *Let (Q, κ) and (Q', κ') be parameterized problems over the alphabets Σ and Σ', respectively. An $\mathcal{FPT}$–reduction ($\mathcal{FPT}$ many–one reduction) from (Q, κ) to (Q', κ') is a mapping $R\colon \Sigma^* \to (\Sigma')^*$ such that:*

1. *For all $x \in \Sigma^*$ we have $(x \in Q \Leftrightarrow R(x) \in Q')$.*
2. *R is computable in time $f(\kappa(x))|x|^{\alpha}$ for some constant α and an arbitrary function f.*
3. *There is a computable function $g\colon \mathbb{N} \to \mathbb{N}$ such that $\kappa'(R(x)) \leq g(\kappa(x))$ for all $x \in \Sigma^*$.*

Point (1) simply states that R is indeed a reduction. Point (2) says that it can be computed in the right amount of time - efficient in all but the parameter. Point (3) states that the parameter of the image is bounded by (a function of) that of the source. This is necessary in order to guarantee that $\mathcal{FPT}$ reductions preserve $\mathcal{FPT}$-ness. I.e. with this definition we obtain that if (Q, κ) reduces to (Q', κ') and $(Q', \kappa') \in \mathcal{FPT}$ then (Q, κ) is also $\in \mathcal{FPT}$.

For detailed studies and explanations on the theory of *Parameterized Complexity* we refer the readers to [3, 6, 7].

	SC	ESCK	MAXC	MAXSC
|G|	FPT	FPT	FPT	FPT
|C|		W[1]-hard	W[1]-hard	W[1]-hard
|R|	Para-NP-hard		Para-NP-hard	Para-NP-hard
|Ag|+|R|	FPT		FPT	FPT

Table 1: Main results regarding the CRG model, for the Successful Coalition (SC), Existence of Successful Coalition of Size k (ESCK), Maximal Coalition (MAXC) and Maximal Successful Coalition (MAXCS) problems.

3. PARAMETERIZED COMPLEXITY ANALYSIS OF COALITION PROBLEMS

In this section, we consider the CRG problems presented in Section 2.1.1), and present the analysis of their complexity when parameterizing by the different input components. We consider four different parameterizations: $|G|$ - the total number of distinct goals, $|R|$ the number of distinct resource types, $|C|$ - the size of the specific coalition in question, and $|Ag| + |R|$ - the sum of the number of resources and agents. This is due to the fact that domains where one of these parameters is bounded are common and can be found in many every day scenarios. Therefore, exposing algorithms that work in pseudo polynomial time given that one of these parameters is bounded, is of interest from the practical point of view. The results are summarized in Table 1.

3.1 The $|G|$ Parameter

We first consider the complexity of the problem when parameterizing by $|G|$ - the number of distinct goals. Determining the complexity of a problem, given that $|G|$ is bounded is very important due to the fact that there are many domains with a large environment but with a small number of goals. For example, the *Carpool* schemes [10] of a large organization can be viewed as a coalition game with a large number of agents (employees) and resources, but a small number of goals (days and hours). Other interesting and well studied examples of cases with a bounded $|G|$ parameter include some biding settings, most negotiation problems and some sensor network problems.

Our analysis shows that all of the problems we consider are $\mathcal{FPT}$ when parameterizing by $|G|$. Intuitively, when the number of goals is bounded coalitions can be formed around the subset of goals, instead of vise versa. Determining whether a certain goal set is both feasible and satisfying can be done in polynomial time. Therefore, when the number of goals is bounded the number of calculations needed to investigate the environment reduces dramatically.

The formal proofs, regarding the $|G|$ parameter, for all the coalition formation problems are established as follows.

THEOREM 3.1. *Checking whether a coalition C is a Successful Coalition is $\mathcal{FPT}$ when parameterizing by $|G|$ or by $|C| + |R|$.*

PROOF. We will prove this by constructing an INTEGER LINEAR PROGRAMMING representation of the problem with $|G|$ variables and $|C| + |R|$ constraints. The problem we define shall be a satisfiability problem (rather than an optimization problem). That is, it consists of only of a set of constraints, and the question is whether there exists an integer solution to this set.

The variables of the programs shall be x_g, for each $g \in G$, where $x_g = 1$ represent the situation that goal g is achieved, and $x_g = 0$ the situation that the goal is not achieved. The INTEGER PROGRAMMING is the following:

$$\forall i \in C \qquad \textstyle\sum_{g \in G_i} x_g \geqslant 1 \tag{1}$$

$$\forall r \in R \qquad \textstyle\sum_{g \in G} x_g \mathbf{req}(g, r) \leqslant \mathbf{en}(C, r) \tag{2}$$

$$\forall g \in G \qquad x_g \in \{0, 1\} \tag{3}$$

The first set of constraints (Equation 1) ensures that each agent has at least one of its goals achieved. While the second set of constraints (Equation 2) ensures that the coalition has enough endowment to achieve all the goals.

It is apparent that any solution for this INTEGER PROGRAMMING problem is a set of goals for which coalition C has enough endowment, and which will satisfy coalition C.

INTEGER PROGRAMMING is $\mathcal{FPT}$ in the number of variables ($|G|$) and in the number of constraints ($|C| + |R|$) ([7], page 222). □

THEOREM 3.2. *Checking whether there is a Successful Coalition of size k is $\mathcal{FPT}$ when parameterizing by $|G|$.*

PROOF. The algorithm of Figure 1 solves the problem and runs in $O(|\Gamma| \cdot 2^{|G|})$ time, which is $\mathcal{FPT}$ in $|G|$.

1. For each $G' \subseteq G$:
   (a) Set $C' = succ(G')$ (the set of agents satisfied by G')
   (b) If $|C'| \neq k$ go to 1.
   (c) If G' is feasible for C' return "$True$".
2. return "$False$".

Figure 1: Algorithm for determining whether there is a successful coalition of size k in the given CRG.

The algorithm's main loop is of complexity $O(2^{|G|})$. Inside the loop, the algorithm creates a subset of agents that are satisfied by G' with a complexity of $O(|Ag|)$ and compares the subset size in a complexity of $O(1)$. In the last step the algorithm checks if the given subset of goals (G') is feasible for C'. This verified in $O(|\Gamma|)$ steps. Hence, the algorithm's complexity is $O(2^{|G|} \cdot |\Gamma|)$. □

THEOREM 3.3. *Checking whether C is a Maximal Coalition is $\mathcal{FPT}$ when parameterizing by $|G|$.*

PROOF. The algorithm of Figure 2 solves the problem and runs in $O(|\Gamma| \cdot 2^{|G|})$ steps.

The algorithm's main loop is of a complexity of $O(2^{|G|})$. Inside the loop, the algorithm creates a subset of agents that

1. For each $G' \subseteq G$:
   (a) Set $C' = succ(G')$
   (b) If G' not feasible for C' go to 1.
   (c) If $C \subset C'$ return "*False*".
2. return *True*.

Figure 2: Algorithm for determining whether the coalition is maximal.

are satisfied by G' in a complexity of $O(|Ag|)$ and ensures that subset C' has enough endowment to accomplish G' (a complexity of $O(|\Gamma|)$). The third step in the loop is to ensure that C' is not a superset of C, again with complexity $\leq O(|\Gamma|)$. Hence, the algorithm's complexity is $O(2^{|G|} \cdot |\Gamma|)$. □

THEOREM 3.4. *Checking whether C is a Maximal Successful Coalition is $\mathcal{FPT}$ when parameterizing by $|G|$.*

PROOF. The algorithm of Figure 3 solves the problem and runs in $O(|\Gamma| \cdot 2^{|G|})$ steps.

1. If C is not a Successful Coalition return "*False*".
2. Else if C is not a Maximal Coalition return "*False*".
3. return "*True*".

Figure 3: Algorithm for discovering whether the coalition is successful and maximal.

The algorithm's first step can be done in $O(\Gamma \cdot 2^{|G|})$ (Theorem 3.1). The algorithm's second step can also be completed in $O(\Gamma \cdot 2^{|G|})$ time (Theorem 3.3). Hence, the algorithm's complexity is $O(2^{|G|} \cdot |\Gamma|)$. □

3.2 The $|C|$ Parameter

The $|C|$ parameter is perhaps the most interesting parameter. There are many applications and domains with large environments but either we can bound the size of the coalition, or it is already known that the size of the coalition is going to be small, e.g. teamwork frameworks. For example the *multi–agent vehicle routing* problem that was described by Sandholm and Lesser [11], contains a bounded and relatively small number of agents in each coalition, while the number of goals and different resources seems to be large.

Some previous works (e.g. [13]) have shown cases where bounding the size of the coalition reduces their problems' complexity. Hence, we initially expected a similar behavior here. Interestingly, as the following proofs demonstrate, this is probably not the case. We prove that when parameterizing by $|C|$ the problems are $W[1]$-hard, which, as explained in Section 2.2, is a strong indication that they are most probably not $\mathcal{FPT}$. Thus, the $|C|$ parameter does not seem to be the source of the problems' hardness. Some intuition to why this is the case can be gained by observing that even if the size of the coalition is bounded, we still remain with the hard KNAPSACK–like problem of choosing the appropriate goals and dividing the resources among them.

The formal claims and proofs follow.

THEOREM 3.5. *Checking whether there is a Successful Coalition of size k is $\mathcal{W}[1]$–hard when parameterized by k.*

PROOF. We prove this by reduction from CLIQUE (parameterized by the size of the clique) a known $\mathcal{W}[1]$–complete problem [5].

Let $G = (V, E)$ a graph with $V = \{v_1, ..., v_n\}$ and k an integer. We construct a set of agents $Ag = \{a_1, ..., a_n\}$ such that each agent represents a single vertex, and a set of resources $R = \{r_1, ..., a_n\}$.

For each agent a_i, we set $G_i = g_i$. Each goal g_i demands resources, such that:

$$\mathbf{req}(g_i, r_j) = \begin{cases} 0, & \text{if } i \neq j; \\ k-1, & \text{otherwise.} \end{cases}$$

For each edge $(v_i, v_j) \in E$: $\mathbf{en}(v_i, r_j) = 1$ and $\mathbf{en}(v_j, r_i) = 1$

The construction of the CRG Γ can be completed in $O(|V|^2 \cdot |E|)$ steps.

The parameter of the new problem (the size of the coalition) is identical to the parameter of the Clique's problem.

We now prove that there is a successful coalition of size k in the CRG Γ_G iff there is a clique of size k in the graph G:

($\Rightarrow$) Assume that there is a coalition of size k in the CRG Γ_G. Namely, each agent a_i, belonging to the coalition, has exactly $k-1$ agents in the coalition, that has its kind of resource (r_i). An agent can have a resource of another agent only if the corresponding vertexes are neighbors. Therefore, the coalition can be successful only if the corresponding vertex for each agent in the coalition is a neighbor of all the corresponding vertexes of all other agents in the coalition. Thus there must be k vertexes that are all neighbors of each other.

($\Leftarrow$) Let us assume that there is a clique of size k in graph G. The clique $Clique = \{v_{l1}, ..., v_{lk}\}$ is composed of k vertexes. Choosing the corresponding agents a_{li} in the CRG will provide a coalition, of size k, where each agent a_i has exactly enough resources ($k-1$ of resource r_i) to complete its goal g_i. □

Notice that this reduction is both a classical many–one reduction, and a $\mathcal{FPT}$–reduction. Therefore, since CLIQUE is a known $\mathcal{W}[1]$–complete problem [5], this reduction can be use to prove that this problem is not only $\mathcal{NP}$–hard problem, but also $\mathcal{W}[1]$–hard problem.

THEOREM 3.6. *Checking whether C is a Maximal Coalition (*MAXC*) is $\mathcal{W}[1]$–hard when parameterizing by $|C|$.*

PROOF. We prove this by reduction from EXISTENCE OF SUCCESSFUL COALITION OF SIZE k (ESCK) (we already proved that this problem is $\mathcal{W}[1]$–hard in Theorem 3.5).

Let Γ be a CRG and k be a number. We construct Γ' such that:

$$Ag' = Ag \cup \{a_z\}\ ,\ G' = G \cup \{g_z\}\ ,\ R' = R \cup \{r_z\}$$

a_z, r_z and g_z are a new agent, a new resource and a new goal (respectively) that did not exist in Γ, such that:

$$\mathbf{req}(g_z, r_i) = \begin{cases} 0, & \text{if } i \neq z; \\ k, & \text{otherwise.} \end{cases}$$

$$\forall r_i \in R \quad \mathbf{en}(a_z, r_i) = 0$$

$$\forall a_i \in Ag,\ i \neq z \quad \mathbf{en}(a_i, r_z) = 1$$

$$G_z = \{g_z\}\ ,\ C = \{a_z\}$$

The construction of the CRG Γ' can be completed in $O(|\Gamma|)$ time.

The parameter of the MAXIMAL COALITION problem ($k' = |C|$) is a function of the parameter of the ESCK problem, and depends only on it ($k' = 1$).

We now prove that there is a successful coalition of size k in the CRG Γ iff C is not a maximal coalition in the CRG Γ':
($\Rightarrow$) Let us assume that there is a successful coalition C' of size k in CRG Γ. Thus the coalition $C'' = C' \cup C$ is a successful coalition in CRG Γ', since C' is successful and the k agents will enable the coalition to accomplish g_z for a_z. In addition, $C'' \supset C$, which means that C is not a maximal coalition.
($\Leftarrow$) Let us assume that C is not a maximal coalition in CRG Γ'. Thus there is a successful coalition C' s.t. $C' \supset C$. A coalition that contains a_z will be satisfied only if it contains at least k other agents that are satisfied. Therefore, if we remove a_z we will have a successful coalition of size k. Consequently, we know we have a successful coalition of size k in CRG Γ. □

THEOREM 3.7. *Checking whether C is a Maximal Successful Coalition (*MAXSC*) is $\mathcal{W}[1]$–hard when parameterizing by $|C|$.*

PROOF. We prove this by reduction from the problem MAXIMAL COALITION (MAXC) (we have already proven that this problem is $\mathcal{W}[1]$–hard in Theorem 3.6).

Let Γ be a CRG and $C \subset Ag$ be a coalition. We construct Γ' such that:

$$G' = G \cup \{g_z\}$$
$$\forall i \in C : \; G'_i = G_i \cup \{g_z\}$$
$$\forall r \in R : \; \mathbf{req}(g_z, r) = 0$$

The construction of the CRG Γ' can be completed in $O(|\Gamma|)$ time.

The parameter of the MAXIMAL SUCCESSFUL COALITION problem (k') is a function of the parameter of the MAXIMAL COALITION problem, and depends only on it ($k' = k$).

We now prove that C is a maximal coalition in the CRG Γ iff C is a maximal successful coalition in the CRG Γ':
($\Rightarrow$) Let us assume that C is a maximal coalition in CRG Γ. Thus C is a maximal coalition in CRG Γ' as well. In addition, we know that all the agents that belong to C are satisfied in Γ'. Therefore, C is a maximal successful coalition in CRG Γ'.
($\Leftarrow$) Let us assume that C is a maximal successful coalition in CRG Γ'. Consequently, C must be a maximal coalition in CRG Γ, since our structure did not prevent other coalitions from being formed. □

3.3 The $|R|$ Parameter

In many applications and domains the number of different resources is known to be very small. Scheduling missions [11] and virtual organizations [2] are well studied examples to such domains.

Investigating the problems using the *parameterized complexity paradigm* shows that limiting the $|R|$ parameter does not relax the problem to an intractable level. For an intuition observe that even with a small number of different resources we remain with a hard problem of finding a "Hitting Set" of goals for the coalitions.

The formal proofs, regarding the $|R|$ parameter, for SUCCESSFUL COALITION, MAXIMAL COALITION and MAXIMAL SUCCESSFUL COALITION problems are established as follows.

THEOREM 3.8. *Checking whether C is a Successful Coalition (*SC*) is $para-\mathcal{NP}$–hard when parameterizing by $|R|$.*

PROOF. The proof is by reduction from the HITTING SET problem. We show that even for the single slice of $|R| = 1$ the problem remains $\mathcal{NP}$-hard. By [7] (page 38) this suffices to prove $para-\mathcal{NP}$–hardness.

Let $U = \{e_1, \ldots, e_m\}$ a finite set of elements, T a collection of sets $T = \{S_1, \ldots, S_n\}$ s.t. $S_i \subseteq U \; \forall i \in \{1, \ldots, n\}$, and k an integer. We construct a set of goals $G = \{g_1, \ldots, g_m\}$ such that each goal represents a single element, and a set of agents $Ag = \{a_i, \ldots, a_n\}$ such that each agent represents a single set (S_i).

For each agent a_i, we set G_i to be the set of goals corresponding to the elements in set S_i.

$$\forall g_j \in G \; \mathbf{req}(g_j, r) = 1$$
$$\mathbf{en}(a_1, r) = k$$
$$\forall i \neq 1 \; \mathbf{en}(a_i, r) = 0$$
$$R = \{r\} \; , \; C = \{Ag\}$$

The construction of the CRG Γ can be completed in $O(|U| + |T|)$ time.

The $|R|$ parameter is bounded to $|R| = 1$.

We now prove that C is a successful coalition in the CRG Γ iff there is a Hitting Set of maximal size of k:
($\Rightarrow$) Let us assume that C is successful. Namely, each agent has at least one of its goals satisfied. Therefore, choosing the corresponding elements will results with a hitting set. In addition, since the coalition has only k endowment of the resource, and each goal demands one resource, the hitting set is of maximal size of k.
($\Leftarrow$) Let us assume we have a hitting set H of maximal size k. Therefore, choosing the corresponding goals in Γ will results with a set of goals that satisfied the coalition. In addition, since the hitting set is of size k (or smaller), it is obvious that the coalition has enough resources to accomplish the goals. □

THEOREM 3.9. *Checking whether C is a Maximal Coalition (*MAXC*) is $Para-\mathcal{NP}$–hard when parameterizing by $|R|$.*

PROOF. The proof is by reduction from GRAND COALITION SUCCESSFUL (GCS). It is easy to see that the GCS is $para-\mathcal{NP}$–hard since the proof used for SUCCESSFUL COALITION (Theorem 3.8) stand for the Grand coalition as well.

Given a CRG Γ, we construct a CRG Γ' and a coalition C such that: $\Gamma' = \Gamma$ and $C = Ag \setminus \{a_n\}$.

The construction of the CRG Γ' and coalition C can be completed in $O(|\Gamma|)$ time.

The $|R|$ parameter of the MAXIMAL COALITION problem is the same as the $|R|$ parameter in the GCS problem.

We now prove that the grand coalition is successful in CRG Γ iff C is not a maximal coalition in the CRG Γ':
($\Rightarrow$) Let us assume that the grand coalition is successful in Γ. Namely, $\{Ag\}$ is a successful coalition and $C \subset Ag$, i.e., C is not a maximal coalition.
($\Leftarrow$) Let us assume that C is not a maximal coalition. Then, since the only strict superset of C is Ag, this indicates that the $\{Ag\}$ coalition is successful, hence the grand coalition is successful. □

THEOREM 3.10. *Checking whether C is a Maximal Successful Coalition (*MAXSC*) is $Para-\mathcal{NP}$–hard when parameterizing by $|R|$.*

PROOF. The proof is by reduction from MAXIMAL COALITION (MAXC) (We already proved it to be $para-\mathcal{NP}$–hard is Theorem 3.9).

Notice that the reduction used to prove that the MAXIMAL SUCCESSFUL COALITION is $\mathcal{W}[1]$-hard in $|C|$ can be used here as well since the $|R|$ parameter is untouched in this reduction. □

3.4 The $|Ag|$ and $|R|$ Parameters

The $|Ag|$ and $|R|$ parameters are the total number of agents and the number of different types of resources in the environment. There are many applications where the number of agents available in the environment is small, and the number of different resources (skills) they have is bounded as well. An example for an application with a bounded number of agents and different resources can be find in Conitzer and Sandholm [2]. Conitzer and Sandholm describe coalition formation between medical companies (agents), each holds some medical patents (resources), and try to create useful drugs (goals). Other examples are scheduling missions, virtual organizations and "just-in-time" incorporation (agents grouping to handle workflows), to name just a few.

The *parameterized complexity* results show that in most considered coalition formation problems, limiting the $|Ag|$ and $|R|$ parameters relaxes the problems, thus they can be solved in a pseudo polynomial time. These results are very interesting since bounding only the $|R|$ parameter does not relax the problems. Bounding the $|Ag|$ parameter decreases the number of possible coalitions. However, determining whether a coalition is successful demands choosing the appropriate goals and dividing the resources correctly among them, which seems much like the KNAPSACK problem. Bounding the number of different resources ($|R|$) enables the algorithms to reduce the number of goals considered and to complete the resource division in pseudo polynomial time.

The formal proofs, regarding the $|Ag|$ and $|R|$ parameters, for SUCCESSFUL COALITION, MAXIMAL COALITION and the MAXIMAL SUCCESSFUL COALITION problems are established as follows.

The formal proof regarding the SUCCESSFUL COALITION problem can be found in Theorem 3.1. Notice, that in this problem the actual parameter is $|C|+|R|$, which means that even if the number of total agents is not bounded but we work with small coalitions (and the number of different resources is bounded) this problem can be solved in pseudo polynomial time.

THEOREM 3.11. *Checking whether C is a Maximal Coalition is $\mathcal{FPT}$ when parameterizing by $|Ag|+|R|$.*

PROOF. The algorithm of Figure 4 solves the problem in $O(|\Gamma| \cdot 2^{2|Ag|+|R|})$ steps.

1. For each $C' \subseteq Ag$:
   (a) If $C \not\subset C'$ go to 1.
   (b) If C' is successful return "$False$".
2. return "$True$".

Figure 4: Algorithm for determining whether the coalition is maximal.

The algorithm's main loop is of a complexity of $O(2^{|Ag|})$. Inside the loop the algorithm checks if the subset of agents C' is a superset of coalition C (a complexity of $O(|C'|)$), and checks whether subset C' is a successful coalition (i.e. can be done in a complexity $O(|\Gamma| \cdot 2^{|C'|+|R|})$ according to Theorem 3.1). Hence, the algorithm's complexity is $O(2^{|Ag|} \cdot 2^{|C'|+|R|} \cdot |\Gamma|) < O(|\Gamma| \cdot 2^{2|Ag|+|R|})$. □

THEOREM 3.12. *Checking whether C is a Maximal Successful Coalition is $\mathcal{FPT}$ when parameterizing by $|Ag|+|R|$.*

PROOF. The algorithm of Figure 5 solves the problem in $O(|\Gamma| \cdot 2^{2|Ag|+|R|})$ steps.

1. If C is not a Successful Coalition return "$False$".
2. Else, if C is not a Maximal Coalition return "$False$".
3. return "$True$".

Figure 5: Algorithm for determining whether the coalition is successful and maximal.

The algorithm's first step can be done in $O(\Gamma \cdot 2^{|C|+|R|})$ time (since the SUCCESSFUL COALITION problem is $\mathcal{FPT}$ in $|C|+|R|$). The algorithm's second step can be completed in $O(\Gamma \cdot 2^{2|Ag|+|R|})$ time (since the MAXIMAL COALITION problem is $\mathcal{FPT}$ in $|Ag|+|R|$). Hence, the algorithm's complexity is $O(2^{2|AG|+|R|} \cdot |\Gamma|)$. □

4. CONCLUSIONS AND FUTURE WORK

In this paper we re-examined several coalition formation problems defined in the CRG model using a parameterized complexity approach. Our analysis showed that even though all these problems have been previously proven to be hard, bounding some parts of the input can result in important tractable cases. Thus, many coalition formations problems may be easy in practice in many real-world applications. On the other hand, we have shown that in some other cases, bounding a parameter does not ease the problem. Throughout, parameterized complexity theory provided us with the framework, terminology and tools to establish the results.

All the problems we examined were found to be $\mathcal{FPT}$ in the $|G|$ parameter. In particular, if in a given domain the number of goals is small - a rather common situation - almost all problems regarding coalition formation, modeled by CRG, can be solved with in pseudo-polynomial time. Thus, limiting the number of goals dramatically reduces the time complexity of what seems to be a hard problem in general.

Similarly, in almost all problems we considered we found that the problems are $\mathcal{FPT}$ in $|Ag|+|R|$. Thus, if the number of agents in the domain and the number of different resources are both small, almost all the coalition formation problems modeled by CRG can be solved with a pseudo-polynomial algorithm (at least those we considered). Interestingly, this does not hold for the $|R|$ parameter alone, for which we proved that bounding this parameter does not relax the problems at all.

Similarly, it seems that limiting $|C|$ - size of the coalition - does not have a dramatic affect on the time complexity of the problems. Namely, limiting the size of the coalitions in a domain with a large number of goals and a large number of agents (or different resources) will not relax the intractability of the problems. Thus, the hardness of the problems does not seem to depend on the size of the coalition, nor on the number of different resources. The practical meaning is that in large domains, even if we work with a very small coalition (such as in teamwork) or with a small number of resources, we may still not be able to compute the answer to many coalition formation problems in a reasonable pseudo polynomial time.

There are many avenues for future work. First and foremost, the analysis can and should be extended to other coalition formation problems. Here, we considered few of the key ones, but there are many other ones. In particular, examining other problem types relating to coalition formation, such as "Efficient Coalition Formation" problems will be interesting. Efficient coalition formation problems focus not only on whether the agent is satisfied by the coalition, but also check the profile of resources the agent is asked to investigate and search for the successful coalition that asks for as little resources as possible. These problems invokes many computational problems, most of which where found to be hard [12, 16, 17]. However, as stated earlier, many of the input parameters tend to be small (such as the number of goals the agents must accomplish). Analyzing these problems using the *parameterized complexity paradigm* will reveal whether an efficient formation of coalitions can be made with a reasonable (pseudo-polynomial) complexity.

This research is based on the CRG model for coalition structures. As the research continues, other coalition formation models should be examined as well (such as the QCG [16]). Each model has a different input structure, and therefore raises different problems and requires different methods. It would be interesting to examine whether the *parameterized complexity paradigm* increases or decreases the similarity between the models.

In this work we discussed several key parameterizing options, but other parameterizations may also be of interest. In particular, the complexity of the problems when parameterizing by $|Ag|$ is an open question. Also, the complexity of SUCCESSFUL COALITION when parameterized by $|C|$ is an open question.

To the best of our knowledge, this is the first research that uses the *parameterized complexity* paradigm to analyze problems raised in coalitions of agents in multi–agent systems. Therefore, it would be wise to broaden this analysis to other coalition's problems (other than formation problems). Interesting problems could include: decision–making in coalitions, elections in coalitions and evaluating coalitions/coalitions decisions.

5. ACKNOWLEDGMENTS

This research is based upon work supported in part under NSF grant 0705587 and ISF grant #1357/07. Sarit Kraus is also affiliated with UMIACS.

6. REFERENCES

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

[2] V. Conitzer and T. Sandholm. Complexity of constructing solutions in the core based on synergies among coalitions. *Artificial Intelligence*, 170:607–619, 2006.

[3] R. Downey. Parameterized complexity for the skeptic. In *Proc. 18th IEEE Annual Conference on Computational Complexity*, pages 147–169, 2003.

[4] R. G. Downey and M. R. Fellows. Fixed parameter tractability and completeness I: Basic theory. *SIAM Journal of Computing*, (24):873–921, 1995.

[5] R. G. Downey and M. R. Fellows. Fixed parameter tractability and completeness II: On completeness for W[1]. *Theoretical Computer Science A*, 141(1–2):109–131, 1995.

[6] R. G. Downey and M. R. Fellows. Parameterized complexity after almost ten years: review and open questions. In *Combinatorics, Computation and Logic, DMTCS'99 and CATS'99. Australian Computer Science Communications, Springer-Verlag, Singapore*, vol. 21, pages 1–33, 1999.

[7] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[8] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H. Freeman, San Francisco, 1979.

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

[10] M. Naor. On fairness in the carpool problem. *Journal of Algorithms*, 55:93–98, 2005.

[11] T. W. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94:99–137, 1997.

[12] O. Shehory and S. Kraus. Coalition formation among autonomous agents: Strategies and complexity. In C. Castelfranchi and J.-P. Müller, editors, *From Reaction to Cognition — Fifth European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW-93 (LNAI Volume 957)*, pages 56–72. Springer-Verlag: Berlin, Germany, 1995.

[13] O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. *Artificial Intelligence*, 101(1–2):165–200, 1998.

[14] G. Weiß, editor. *Multi-Agent Systems*. The MIT Press: Cambridge, MA, 1999.

[15] M. Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons, 2002.

[16] M. Wooldridge and P. E. Dunne. On the computational complexity of qualitative coalitional games. *Artificial Intelligence*, 158(1):27–73, 2004.

[17] M. Wooldridge and P. E. Dunne. On the computational complexity of coalitional resource games. *Artificial Intelligence*, 170(10):853–871, 2006.