A Network Flow Approach to Coalitional Games

Rahwan Talal, Michalak Tomasz, Madalina Croitoru, Sroka Jacek, Jennings Nicholas

A Network Flow Approach to Coalitional Games

Talal Rahwan[1], Tomasz Michalak[1], Madalina Croitoru[2], Jacek Sroka[3], and Nicholas R. Jennings[1]

[1]**School of Electronics and Computer Science, University of Southampton, UK**
[2]**LIRMM, University of Montpellier, France**
[3]**Institute of Informatics, University of Warsaw, Poland**

Abstract. In this paper we propose a novel approach to represent coalitional games, called a Coalition-Flow Network (CF-NET), that builds upon a generalization of the *network flow* literature. Specifically, this representation is based on our observation that the coalition formation process can be viewed as the problem of directing the flow through a network where every edge has certain capacity constraints.

1 Introduction

One of the key issues in multi-agent coalition formation [2] is the *Coalition Structure Generation (CSG)* problem, which involves dividing the set of agents into subsets (i.e., coalitions) so that the overall efficiency of the system is maximized. Such a division is referred to as a *coalition structure*. This problem has recently attracted considerable attention in the multi-agent system literature [6, 5, 4]. The input is typically represented using a *characteristic function* which assigns a value for every possible coalition representing its optimal performance. Such representation of coalitional games is widely known as the *Characteristic Function Game (CFG)* representation.

A key issue when developing efficient solutions to the CSG problem is how the game is represented. This is important since a straightforward listing of the values of all the possible coalitions, as is the case with the conventional CFG representation, requires a space of exponential size (e.g., the fact that two agents are identical is completely disregarded when using this representation). In contrast, a well-crafted representation may be able to exploit the structure of the system and, therefore, model it in a much more concise manner. This can also facilitate significantly more effective solutions to the computational problem at hand [1, 2, 3]. In this paper we address the above issue by proposing a novel approach to model coalitional games based on a generalization of the *network flow* literature[7].

2 Preliminaries

A *characteristic function game* (CFG) representation of a coalitional game is a tuple (A, v), where $A = \{a_1, \ldots, a_n\}$ is a set of agents and $v : 2^A \rightarrow \mathbb{R}$ is a *characteristic function* that assigns to every coalition its value expressed as a real number. Intuitively, $v(C)$ is the total payoff that the members of coalition C can attain when working together and coordinating their activities. A usually implicit game rule is that every agent can belong to exactly one coalition. Given this, let us consider the following example:

Example 1 (CFG**)** *There are four unmanned aerial vehicles (UAVs) patrolling an area in order to locate potential enemies. The cooperation between aircrafts is desirable, as the quality of the data transmitted to the base is substantially enhanced if more UAVs monitor the same object. However, suppose that the radio equipment fitted to each aircraft allows for effective communication, i.e., to create a profitable coalition, with only one other aircraft. The* CFG *representation of this game is* (A, v)*, where* $A = \{a_1, a_2, a_3, a_4\}$ *and a possible characteristic function* v *is (assuming symmetry):*

$v(\{a_1\}) = 1$	$v(\{a_1, a_3\}) = 2.5$	$v(\{a_1, a_2, a_3\}) = 3.5$
$v(\{a_2\}) = 1$	$v(\{a_1, a_4\}) = 2.5$	$v(\{a_1, a_2, a_4\}) = 3.5$
$v(\{a_3\}) = 1$	$v(\{a_2, a_3\}) = 2.5$	$v(\{a_1, a_3, a_4\}) = 3.5$
$v(\{a_4\}) = 1$	$v(\{a_2, a_4\}) = 2.5$	$v(\{a_2, a_3, a_4\}) = 3.5$
$v(\{a_1, a_2\}) = 2.5$	$v(\{a_3, a_4\}) = 2.5$	$v(\{a_1, a_2, a_3, a_4\}) = 4.8$

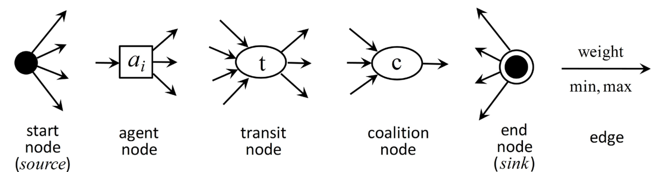

Figure 1*:* Components of the CF-NET network

3 The CF-NET Representation

In this section we propose our representation for coalitional games, which is based on a generalization of the *network flow* literature. Specifically, a network is a directed graph with a starting node (a *source*) and an end node (a *sink*). Moreover, each edge in the network has a certain capacity associated with it. Now, let us assume that there is a flow in the network which must satisfy the following two conditions: (1) the flow that goes through an edge must not exceed the capacity of that edge and (2) for every node other than the source and the sink, the total flow that enters that node must be equal to the total flow that leaves it. The second condition is called a *flow conservation rule*. Subject to the above two conditions, the problem of directing the flow through the network so that it is maximized is widely known as the *maximum flow problem*.

Our representation, which we call Coalition-Flow Network (CF-NET), is a generalization of flow networks, defined as follows:

Definition 2 *A Coalition-Flow Network (CF-NET) is a tuple* $(\mathcal{N}, \mathcal{E}, max, min)$*, where* $(\mathcal{N}, \mathcal{E})$ *is an acyclic digraph with a set of nodes* $\mathcal{N}$ *and a multiset of directed edges* $\mathcal{E}$*. Specifically:*

1. *The set of nodes* $\mathcal{N}$ *is the union of the following disjoint sets:* N^s *contains the source node from which the flow is pushed into the network;* N^k *contains the sink node to which the flow needs to be maximized;* N^a *is the set of agent nodes which represent unique agents;* N^c *is the set of coalition nodes which all are connected directly to sink and represent subsets of agents;* N^t *is the set of transit nodes which also represent subsets of agents, but are not directly connected to the sink.*

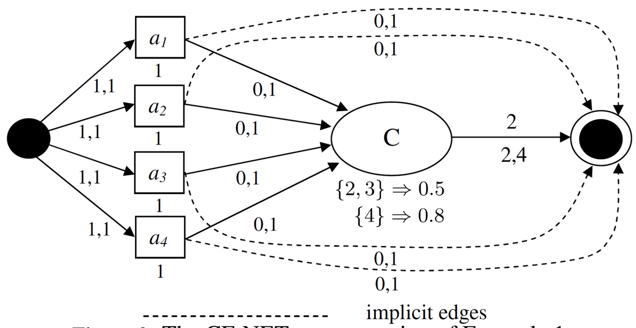

Figure 2: The CF-NET representation of Example 1.

2. *The underlying set of edges is* $E = (N^s \times N^a) \cup (N^c \times N^k) \cup E'$, *where* $E' \subseteq \{(n_i, n_j) : n_i \in N^a \cup N^t \wedge n_j \in N^c \cup N^t \wedge n_i \neq n_j\}$. *The weight of an edge* $(n_i, n_j) \in E$ *is the multiplicity of that edge in* $\mathcal{E}$, *which is equal to 1 unless* $n_i \in N^c \cup N^t$, *in which case it is greater than or equal to 1.*
3. *Functions* $min : E \to \mathbb{N}_+ \cup \{0\}$, $max : E \to \mathbb{N}_+ \cup \{0\}$ *denote the minimum and maximum capacity of the edges respectively.*

The nodes in N^s and N^k will simply be denoted $source$ and $sink$ respectively. To visually represent our network we use the graphical primitives shown in Figure 1. The flow that goes from the source, through the network, to the sink will be called a *CF-flow*. Formally:

Definition 3 (CF-flow) *Let us consider a CF-NET* $(\mathcal{N}, \mathcal{E}, max, min)$. *A coalition formation flow in CF-NET, or CF-flow, is a function* $f : E \to \mathbb{N}_+ \cup \{0\}$ *satisfying the following properties:*[1]

- $\forall (n_i, n_j) \in E : min(n_i, n_j) \leq f(n_i, n_j) \leq max(n_i, n_j) \vee f(n_i, n_j) = 0$;
- $\forall n_j \in N^a \cup N^c \cup N^t$, $\sum_{(n_i,n_j)\in E} f(n_i, n_j) = \sum_{(n_j,n_k)\in E} f(n_j, n_k)$.[2]

Intuitively, a CF-flow can be interpreted as agents (or types of agents) going through the network, and the process of directing that CF-flow can be interpreted as the process of determining which coalitions these agents (or types) should form. Basically, a CF-flow is pushed from the source to the agent nodes (hence $E \supset N^s \times N^a$), and then to coalition nodes that connect different agent nodes, and then finally to the sink (hence $E \supset N^c \times N^k$).

Example 4 (CF-NET example) *The game from Example 1 can be represented with the CF-NET shown in Figure 2. In particular, this CF-NET contains four agent nodes, each assigned a value of 1 to represent the values of singleton coalitions. The edges going from the source to these nodes have a capacity of* 1 *each to reflect the fact that every agent in Example 1 can only join one coalition. The CF-NET also has one coalition node to which all the agents are connected. The value assigned to this node is* 0.5 *when the total flow going through it is 2 or 3, and* 0.8 *when that flow equals 4.*

The values assigned to the coalition node in Figure 2 can be interpreted as follows. The synergy that results from the cooperation of any two or three agents is 0.5. Thus, the value of coalition $\{a_1, a_3\}$ equals $v(\{a_1\}) + v(\{a_3\}) + 0.5 = 2.5$. Similarly, the synergy from the cooperation of all four agents is 0.8. Therefore, $v(\{a_1, a_2, a_4, a_4\}) = 1 + 1 + 1 + 1 + 0.8 = 4.8$. As can be seen in Figure 2, this — as well as any other — CF-NET contains implicit edges connecting every agent node to the sink, to ensure that agents can choose to play the game as singletons. It can also be seen in Figure 2 how the possible CF-flows represent the possible ways of partitioning the agents. For example, a CF-flow going from the agent nodes representing a_1 and a_2 to the coalition node, and from those representing a_3 and a_4 to the sink (through the implicit edges), would represent the coalition structure $\{\{a_1, a_2\}, \{a_3\}, \{a_4\}\}$. Also note that the weight of the edge leaving the coalition node is 2. This means there are actually two such edges in the network. Now, consider the case where the flow goes from all the agent nodes to the coalition node. In this case, controlling the flow such that all 4 units of the flow leave the coalition node through one edge means that the agents will form the grand coalition (and thus get a synergy of 0.8). On the other hand, by controlling the flow such that every two units leave from a different edges, the agents will form two coalitions of size 2 (thus getting, at every edge, a synergy of 0.5).

4 Conclusions

In this paper we proposed CF-NET, a representation for coalitional games inspired by network flows. We showed that the coalition structure generation problem, given our representation, becomes similar to the *maximum flow problem*.

REFERENCES

[1] X. Deng and C. Papadimitriou, ‘On the complexity of cooperative solution concepts.’, *Mathematical Oprerational Research*, (19), 257–266, (1994).

[2] S. Ieong and Y. Shoham, ‘Marginal contribution nets: A complact representation scheme for coalitional games’, *ACM EC-06*, 170–179, (2006).

[3] T. Michalak, T. Rahwan andn J. Sroka, A. Dowell, M. Wooldridge, P. PcBurney, and N. R. Jennings, ‘On representing coalitional games with externalities’, in *In Proceedings of the Tenth ACM conference on Electronic Commerce (ACM-EC 09)*, pp. 11–20, (2009).

[4] T. Rahwan and N. R. Jennings, ‘Coalition structure generation: Dynamic programming meets anytime optimisation’, in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI 2008)*, pp. 156–161. AAAI Press / The MIT Press, (2008).

[5] T. Rahwan and N. R. Jennings, ‘An improved dynamic programming algorithm for coalition structure generation’, in *In Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, pp. 1417–1420, (2008).

[6] T. Rahwan, S. D. Ramchurn, V. D. Dang, and N. R. Jennings, ‘Near-optimal anytime coalition structure generation’, in *In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, ed., M. Veloso, pp. 2365–2371, (2007).

[7] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, Springer, 2003.

---

[1] For convenience, for any edge $(n_i, n_j) \in E$, we write $f(n_i, n_j), min(n_i, n_j), max(n_i, n_j)$ as a shorthand for $f((n_i, n_j)), min((n_i, n_j)), max((n_i, n_j))$ respectively.

[2] This means an agent node or transit node satisfies the flow conservation rule.