

Multi-Agent Coalition Re-Formation and League Ranking

Onn Shehory
IBM Haifa Research Labs
Haifa University
Haifa 31905, ISRAEL
onn@il.ibm.com

Gal Kaminka
Dept. of Computer Science
Bar Ilan University
Ramat Gan 52900, ISRAEL
galk@cs.biu.ac.il

Eran Shoham
Technion, Israel Institute of
Technology
Haifa 32000, ISRAEL
erans@tx.technion.ac.il

1. Introduction

Feasible solutions of the coalition formation problem often result in sub-optimal coalitions [2][3]. The question we address is whether we can utilize knowledge of existing sub-optimal coalitions, their members and their capabilities, to form better coalitions. Given a set of coalitions (referred to as a league), we can form new coalitions by extracting agents from different coalitions within the league, and merging them into new (hopefully better) coalitions. We refer to this process as *coalition reformation*.

We focus in this paper on a particular variation of the coalition reformation problem—called the homogeneous coalition reformation problem—in which all coalitions have m (common) roles, and an agent has only one role (or one role slot) it can fulfill, e.g., goalie, leader, follower, etc. Thus, creating a new coalition involves selection of m agents (each specialized for its role) from the n original coalitions. For the homogeneous coalition reformation problem, the number of possible coalitions is thus $O(n^m)$. The set of all possible coalitions is called a *composed league*. The objective is to determine the best coalition in the composed league. We refer to leagues in which quality relations among coalitions are intransitive. In such leagues, the best coalition is only best in the sense that it is better than most other coalitions. Thus it may still be worse than some. This explicitly precludes using the (log complexity) single-elimination algorithm.

To determine this optimal coalition, the coalition reformation mechanism provided in this paper takes a composed league, and uses a *league-ranking* procedure to rank the coalitions with respect to each other. A naïve ranking scheme matches each coalition against all others— $O(n^{2m})$ matches—and the results of these matches are used for ranking. Given that each match is costly in time, this solution becomes infeasible in practice for even relatively small values of n and m . For instance, with four original coalitions ($n=4$), each of five members ($m=5$), and a match time of one minute, the naïve ranking would take close to two years to compute the optimal reformed coalition of TeamBots [1] robotic soccer.

To address this challenge, we introduce a novel *league-ranking* heuristic that allows a greedy search to approximate the optimal coalition in $O(n^m)$ number of matches (i.e., the size of the league). Although this is still a large number of matches, our technique makes it feasible in practice to determine the approximate optimal coalition, in leagues such as that described above (which our approach solves in a few days, compared to two years with the naïve solution).

2. League Ranking

The proposed league ranking scheme reduces the ranking complexity to linear in the size of the league. This scheme uses a pairwise team capability relation measure, denoted *Rel*, as the major parameter on which the ranking is based. *Rel* is defined below, and the details of the ranking scheme follow. For brevity, we call the scheme *Rel* too. Given a pair of teams A, B , we define $Score_{AB}$ as the cumulative score that team A gained during a match against team B . *Rel* is a measure that uses *Score* in order to quantify the relative competence of such a pair:

$$Rel_{AB} = \frac{Score_{AB} + 1}{Score_{BA} + 1} \quad \text{if } Score_{AB} > Score_{BA}$$
$$Rel_{AB} = -\frac{Score_{BA} + 1}{Score_{AB} + 1} \quad \text{otherwise}$$

The *Score* values are a good measure of the gap between the competences of two teams, because they accumulate the mutual scoring of the teams throughout a substantial number of games. *Rel*, which is the ratio between the mutual *Score* values, can be considered as the ratio between the teams' average mutual competence. The values of *Rel* can be any real, except zero. If the teams have equal competence, $Rel=1$. If the teams are not equally capable, $Rel_{BA} = -Rel_{AB}$, the positive value assigned to the superior team. We add one to the denominator in order to prevent division by zero.

The Rel algorithm:

1. initialize L with a copy of the league to be ranked.
2. choose a random team $A \in L$.
3. for each team $B \in L, B \neq A$:

- (a) set a match between A and B.
- (b) compute $Score_{AB}$ and $Score_{BA}$ and then Rel_{AB} .
4. set the value of Rel_{AA} to 1.
5. sort the teams in an ascending Rel values order.

The result of this algorithm will be a list of teams sorted by Rel values. For each pair of teams B, C in the sorted list, the order between them (from the viewpoint of A) is $B > C$ (B is better than C) if and only if $Rel_{AB} < Rel_{AC}$. Of course, the first team on the list is the best team.

The complexity of the Rel algorithm for a league of size k is as follows. Whereas the naïve scheme requires $O(k^2)$ matches, Rel requires only $O(k)$ matches. This is a significant reduction in complexity, given that k itself, in particular in the case of team re-formation, is a large, possibly exponential, number. The sort requires $O(k \log k)$ operations, seemingly greater than the $O(k)$ complexity of matches, however whereas each match is a costly operation, a sort operation is merely a comparison among two numbers. Denote the two complexities by $C_1 k \log k$ and $C_2 k$, and since $C_1 \ll C_2$, we have $C_1 k \log k + C_2 k \sim C_2 k$, hence the overall complexity is dominantly linear.

3. Experimental Evaluation

We have examined the quality of the Rel via a set of experiments in various league domains. Space limitations allow us to briefly present here only a sample.

We first show that Rel and (and its iterative version, $steps$) indeed find the optimal team. For this, we need to know what the optimal team is. We run these experiments in the TeamBots soccer domain. TeamBots makes available several pre-built robotic soccer teams, which we use as the original coalitions for our experiments. For practical reasons, we are able to compute (brute-force) the optimal team only for small composed leagues. In particular, we refer to two basic leagues with teams of size 5. For each basic league, a composed league is computed. The first composed league contains all 2^5 composed teams, re-formed from the agents of the teams CDTeamHetero and JunTeamHeteroG. The second league consists of all 2^5 composed teams re-formed from the agents of the teams CDTeamHetero and BrianTeam. We denote the basic teams CDTeamHetero, JunTeamHeteroG and BrianTeam by C, J and B respectively. We denote the composed leagues by CJ and CB, corresponding to the basic teams from which their agents are drawn. Table 1 presents the details of the TeamBots composed leagues.

League name	League size	Coalition size	Games in match	Max. possible matches
CJ	32	5	8	496
CB	32	5	8	496

Table 1: The TeamBots composed leagues

To rank the composed leagues via the naïve ranking algorithm, we ran in both leagues all the possible $(31 \times 32 = 992)$ directional matches, thus for any pair of

teams, both sides of the match were considered. Since each such match consisted of 8 games in each direction, a total some of 7936 games are required. Running all of the matches and sorting the results, we found that the best coalition in the CJ league was JJCCJ and the best coalition in the CB league was BBCCB. When running Rel , we wanted to find out how applicable Rel is. This can be learned from the portion of teams that, if selected as a basis, will lead Rel to a correct indication of the best coalition. The success portion column in Table 2 shows these results. We further wanted to find out how well Rel ranks the best coalition overall. This can be learned from the average position in which Rel ranks the true top coalition (average top rank column of Table 2).

League name	Success portion	Average rank of true top coalition
CJ	22/32	1.594
CB	8/32	3.594

Table 2. Rel results for the TeamBots leagues

As seen in Table 2, Rel does not always rank the top coalition first, and it may have a high rate of misses (as seen for CB). However, even in misses, the rank given to the true top team is among the top teams (even for CB as, given 32 coalitions, the average rank was 3.594).

We examined $steps$, measuring the average, minimal and maximal number of iterations the algorithm required to find the top coalition. Table 3 presents the results. Given that the size of a league is 32, and that $steps$ may thus iterate up to 32 times, the resultant average number of iterations is a good result, showing the success of $steps$ in finding the top coalition at a relatively low complexity.

League name	Ave. no. iterations	Min. no. iterations	Max. no. iterations
CJ	2.469	1	5
CB	2.781	1	4

Table 3. Steps results for the TeamBots leagues

For large leagues, computing the naïve ranking is infeasible. We have nevertheless computed Rel and $steps$ for both large robotic and human leagues (with 2^{10} possible re-formed teams and 2^{20} possible matches). $Steps$ found the top coalition in 4.3 days (naïve would take 2 years). Rel and $steps$ performed very well in ranking the human leagues too. We thus conclude that Rel is scalable, and applicable across domains. It further overcomes the computational limitation of naïve ranking.

7. References

- [1] Balch, T. TeamBots. www.teambots.org
- [2] Kraus, S., Shehory, O., and Taase, G. Coalition formation with uncertain heterogeneous information. Proc. AAMAS'03 (Melbourne Australia, 2003), 1-8.
- [3] Sandholm, T., and Lesser, V. Coalition formation among bounded rational agents, Artificial Intelligence Journal 94(1-2, (1997), 99-137.