Coalition Calculation in a Dynamic Agent Environment

Ted Scully EDWARD.SCULLY@NUIGALWAY.IE
Michael G. Madden MICHAEL.MADDEN@NUIGALWAY.IE
Gerard Lyons GERARD.LYONS@NUIGALWAY.IE
Department of Information Technology, National University of Ireland, Galway, Ireland.

Abstract

We consider a dynamic market-place of self-interested agents with differing capabilities. A task to be completed is proposed to the agent population. An agent attempts to form a coalition of agents to perform the task. Before proposing a coalition, the agent must determine the optimal set of agents with whom to enter into a coalition for this task; we refer to this activity as *coalition calculation*. To determine the optimal coalition, the agent must have a means of calculating the value of any given coalition. Multiple metrics (cost, time, quality etc.) determine the true value of a coalition. However, because of conflicting metrics, differing metric importance and the tendency of metric importance to vary over time, it is difficult to obtain a true valuation of a given coalition. Previous work has not addressed these issues. We present a solution based on the adaptation of a multi-objective optimization evolutionary algorithm. In order to obtain a true valuation of any coalition, we use the concept of Pareto dominance coupled with a distance weighting algorithm. We determine the Pareto optimal set of coalitions and then use an instance-based learning algorithm to select the optimal coalition. We show through empirical evaluation that the proposed technique is capable of eliciting metric importance and adapting to metric variation over time.

Appearing in *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, 2004.

1. INTRODUCTION

A coalition is a set of self-interested agents that agree to cooperate to execute a task or achieve a goal (Tsvetovat et al., 2000). Coalition formation is currently an active area of research in multiagent systems. This paper considers coalition formation in the context of a market-place where organizations (represented by agents) can cooperate to bid for and perform a task.

While the techniques proposed in this paper are generally applicable, we provide context for the work by considering a simplified model of a real world transport market-place. Each agent represents a transportation firm, and its abilities relate to the routes its firm can service. A task involves the transportation of an item from a collection point to a delivery point. The task is broken into subtasks, which are the routes that constitute a complete journey. Because different firms have different abilities, their representative agents may form coalitions to service a transportation task.

In this domain, a transportation task consisting of several subtasks is proposed in the market-place. Each agent attempts to form a coalition with other agents to bid for the task. Thus, multiple coalitions are proposed and these compete to be awarded the task. We examine the problem from the perspective of an individual agent; we are not concerned with overall system performance and we do not make any assumptions about other agents' strategies for coalition formation. We assume an agent can provide assessments of the abilities of all other agents in the market, but that these assessments are based on personal beliefs rather than objective information. Based on these assessments and the task proposed, it must determine the optimal coalition to propose. We define *coalition calculation* as the determination of the optimal set of agents with which to enter into coalition. Dutta and Sen (2002) observe that in real-life scenarios, multiple objectives like time, quality, dependability, etc., will be involved when an agent evaluates the benefit of interacting with another agent. This statement also holds true for coalition calculation: an agent will want to consider all available information in order to obtain an accurate valuation of interacting with a number of other agents in a coalition. This information is contained in the profile of other agents' abilities, as assessed by an agent. An agent's ability to perform a particular subtask is measured by multiple metrics. In the context of our transportation domain, for example, an agent's ability to deliver an item could be measured by the metrics of time, cost and reliability.

A number of complications arise in obtaining an accurate value of a coalition using multiple metrics. Firstly, metrics are rarely of equal importance. For example, cost may be significantly more important than reliability or time in one situation, but this may change depending on the type of market and specific market conditions. Secondly, metrics often conflict; for example, an agent may aim to minimize both cost and time, but deliveries at lower cost may take longer time. Another complication occurs in dynamic markets, where metric importance may vary over time. For example, the significance of cost might decrease over time while the importance of reliability increases. Hence, techniques must be sensitive to market changes that can affect metric importance.

To address these problems, we propose a new algorithm called *E-Pareto*, which is based on a multi-objective optimization evolutionary algorithm (MOEA). An MOEA combines multiple-objective decision making and evolutionary computation. The key concept in obtaining a true valuation of a coalition is Pareto dominance. The combination of an evolutionary algorithm and Pareto dominance allows for the approximation of the Pareto optimal set of coalitions (Zitzler et al., 2004). We incorporate into this a distance weighting algorithm to maintain diversity when searching for the Pareto optimal solution set, and to encourage search in areas of solution space that have been previously successful.

In order to select the optimal coalition from the approximated Pareto set, we use an instance-based learning (IBL) algorithm that rewards/punishes coalitions based on their proximity to previously successful/unsuccessful coalitions. We also incorporate exploration into this IBL algorithm so that it does not always pursue the greedy strategy of selecting the coalition that obtains the highest score.

This paper is organized as follows. We describe the problem in more detail in Section 2, followed by a detailed description of our *E-Pareto* solution in Section 3. Section 4 presents empirical results and analyzes the performance of the solution. Section 5 describes a collection of related work and details why the problem described above cannot be addressed by existing techniques. Finally, conclusions are drawn in Section 6.

2. Problem Description

We consider an environment consisting of a set of n self-interested agents $A = \{a_1, a_2, a_3, ..., a_n\}$. Tasks to be completed are proposed to all agents in A, where a task T consists of m subtasks $\{t_1, t_2, t_3, ..., t_m\}$.

Bearing in mind the individual-agent perspective taken in this work, as outlined in the Introduction, each agent maintains a private account of the abilities of other agents within the environment. For a subtask t_k, an agent a_i has an associated ability, B_{ik}. Since the ability represents p metrics such as speed, reliability, etc., B_{ik} is a vector: $\{b^1_{ik}, b^2_{ik}, b^3_{ik}, ..., b^p_{ik}\}$. We represent metric values in the range 0 to 1. No assumption is made about how this private data is derived; the objective of each agent is the determination of the optimal coalition given the data available to it. (In the absence of other information, it is rational for an agent to act on the basis that its beliefs are accurate.)

A coalition C consists of a set of agents such that $C \subseteq A$. A function *alloc* determines the agent within C that will perform each individual subtask: the agent within C that will perform subtask t_k is $alloc(k)$. An agent may perform more than one subtask in a coalition.

3. Solution Approach

An agent's objective is to determine the optimal coalition given its beliefs about the abilities of other agents, while remaining sensitive to market conditions. This section describes our *E-Pareto* algorithm, which is designed for this task. Since the ability of each agent in the coalition is represented by multiple metrics, the determination of the optimal coalition becomes a multi-objective optimization problem. We tackle this problem using an adapted MOEA. The MOEA we use is based on the Nondominated Sorting Genetic Algorithm (NSGA) (Srinivas & Deb, 1994). We use Pareto dominance and a distance weighted algorithm to determine the Pareto optimal set sensitive to market conditions (see Section 3.1). The algorithm terminates after a number of generations, returning the Pareto optimal set of coalitions. We detail how the optimal coalition is chosen from this Pareto optimal set using an instance-based learning algorithm in Section 3.2. We also describe how exploration of the solution space can be beneficial compared to a consistently greedy selection in Section 3.3.

3.1. Coalition Valuation

An agent has a different ability for each subtask it can perform. These abilities are represented by multiple metrics. A coalition's ability to perform a task can be determined by the abilities of its member agents to perform their respective subtasks. Hence, the ability of a coalition is also represented by multiple metrics. For example to calculate the quality metric of a coalition for a given task, we sum the quality metric of each member agent for the specific subtask they are performing. This value is then normalized by dividing it by the number of member agents in the coalition. Thus, for a coalition C_a, its ability β_a to perform the task is the vector $\{\beta_a^1, \beta_a^2, \beta_a^3, ..., \beta_a^p\}$ such that

$$\beta_a^r = \frac{1}{m} \sum_{i=1}^{m} b_{alloc(i)i}^r \tag{1}$$

(As before, m is the number of subtasks in T, which is also equal to the number of agents in a coalition to service T.) Thus, coalitions can be plotted in m-dimensional space based on their respective abilities.

In the MOEA, an individual represents a coalition consisting of several agents, configured so that each of these agents is assigned to 1 or more subtasks. We calculate fitness of an individual, $F(i)$, according to (2). The terms of this equation are explained in the following paragraphs. In our scheme, lower values are assigned to better individuals.

$$F(i) = L(i) + \alpha D(i) + (1 - \alpha) ND(i) \tag{2}$$

The first term, $L(i)$, relates to Pareto dominance. A vector x is said to dominate a vector y ($x \succ y$) if no component of x is smaller then the corresponding component of y and at least one component is greater (Zitzler et al., 2004). Each coalition i from the set that dominates all other coalitions in the population is given a fitness level of $L(i) = 1$ and is then removed from the population. Then, the next non-dominated set of coalitions are assigned a fitness level of $L(i) = 2$. This continues until the population is empty. At this stage, all coalitions have received an integer rating value based on Pareto dominance.

The term $D(i)$ in equation (2) seeks to promote diversity in the population, to counteract the tendency of evolutionary algorithms toward a single solution. We adopt an approach similar to that taken by the Strength Pareto Evolutionary Algorithm 2 (Zitzler et al., 2001). For an individual i, the average Euclidian distance, $AD(i)$, from i to all other individuals in the population is calculated. The density of individual i is then computed as $D(i) = \frac{1}{AD(i)+2}$. This term therefore penalizes individuals in areas of high density.

The purpose of the term $ND(i)$ is to ascribe better fitness scores to individuals that are more similar to previously successful coalitions. This encourages the MOEA to focus on areas of the solution space that have exhibited a high success rate, as such areas should correspond to accurate evaluations of metric importance. To achieve this, each time a coalition succeeds in obtaining a task, the evolutionary algorithm records its details in a set P. Then, for each individual i within the population, the average distance from i to all coalition points in P is calculated. $ND(i)$ is then computed as the average distance normalized by dividing it by the maximum possible distance D_{max} between two coalitions. Therefore, individuals that are further from previously successful coalitions are given larger penalties by this term.

Clearly, $L(i)$ is the most important term in (2), as it is an integer value whereas $D(i)$ and $ND(i)$ have values less than 1. The parameter, α, has a range of 0-1 and is used to control the relative importance of $D(i)$ and $ND(i)$. Since $\alpha D(i) + (1 - \alpha)ND(i)$ cannot exceed 1, these terms just affect the relative fitness's of individuals with the same level of Pareto optimality. When initialized, the algorithm has no available historical data. In this case the value of $ND(i)$ is set to 0, hence fitness calculation is purely dependent on the value of $L(i)$ and $D(i)$.

3.2. Coalition Selection using IBL

After evolving the population for a specified number of generations, the MOEA returns an approximate Pareto optimal set of coalitions. The preferred coalition must now be selected from this set. To distinguish between these coalitions, we propose a scoring mechanism that uses instance-based learning (IBL) to take into consideration previous successful and unsuccessful coalitions.

We assume that an agent is informed of the winning coalition after proposing a coalition. Each time the MOEA proposes a coalition, it records both this coalition and the coalition that is successful in obtaining the task in a set Q, such that $Q = \{q_1, q_2, q_3, ..., q_r\}$. Each coalition q_a in Q is assigned a value $V(q_a)$, where $V(q_a) = 2$ if the coalition's bid for the task was successful or $V(q_a) = -1$ if it was unsuccessful. Q has a maximum capacity so that older results are forgotten over time.

Based on this, the IBL algorithm assigns a score $S(i)$ to each coalition i that is a member of the Pareto optimal set based on weighted distance from members of Q. The score is based on the inverse distance squared from i to each member q_j of Q, weighted by $V(q_j)$:

$$S(i) = \sum_j \frac{V(q_j)}{|i - q_j|^2} \qquad (3)$$

Thus, the closer a coalition is to a negative result the more its score will be downgraded; the closer it is to a positive result the more its score will be upgraded. The weighting of $V(q_j)$ ensures that in a competitive area of the solution space, where there will be a high concentration of both positive and negative results in close proximity, positive results are given greater importance than negative results. The coalition with the highest $S(i)$ is chosen, with ties being broken at random. When the algorithm is initialized there is no available historical data, so all coalitions receive a score of 0. Initially therefore, the algorithm will select a coalition at random from the Pareto optimal set.

3.3. Adding Exploration to IBL

The IBL approach described above is purely greedy, as it focuses attention only on regions of the solution space that have exhibited a high success rate in the past. By doing this, it potentially misses regions that could possibly lead to higher success rates. To avoid this, we extend the technique to incorporate exploration into the selection process. This also allows the agent to adapt more quickly to changing metric values. Essentially, during an exploration phase an area of the Pareto curve is picked at random to explore. Future coalition selection will place emphasis on this area if, in the exploration phase, it is found to have a high associated success rate.

As in Section 3.2, the set Q records all coalitions proposed by the evolutionary algorithm as well as all winning coalitions. Exploration phases are initiated with a probability ϕ. To prevent coalition selection becoming too random a process, a small value of ϕ should be adopted. At the start of an exploration phase, a random coalition is chosen from the Pareto optimal set of coalitions. If this coalition is successful then the IBL algorithm begins to explore this area of the curve. This is done by creating a new set Q' that temporarily replaces Q. Q' initially contains just the randomly selected coalition and, like Q, records all coalitions proposed by the agent and all winning coalitions for the duration of the exploration phase. Since the IBL's scores, given by (3), are computed relative to Q', subsequent coalitions proposed will be in the same area of the Pareto curve. The exploration period ends after a pre-determined number of coalitions have been proposed. The success rates of Q and Q' are then compared. The success rate is calculated by dividing the number of successful coalitions in a set by the total number of coalitions in the set. If the success rate of Q is greater then the success rate of Q', then the exploration phase has proved unsuccessful and Q' is discarded. In that case, the IBL algorithm returns its focus to the area prior to exploration. However, if the success rate of Q' is greater then that of Q, exploration has proved successful. In that case, Q is set equal to Q' so that the IBL will assign higher scores to coalitions in this new area of the Pareto optimal curve.

4. Empirical Evaluation

The objective of the empirical evaluation is to demonstrate that our *E-Pareto* algorithm is capable of accurately evaluating coalitions by eliciting metric importance and adapting to metric variations over time for conflicting metrics. We also wish to establish that *E-Pareto* is an advancement when compared with previous approaches to coalition evaluation. We have developed a simulation testbed that contrasts the performance of *E-Pareto* against three other evolutionary based algorithms utilizing previous techniques of coalition evaluation. The following subsection presents the methodology and the results.

4.1. Experimental Methodology

As stated earlier, the coalition calculation operates from the perspective of one agent. This agent is a member of a market-place with a population of 250 agents, and maintains a personal profile of each agent's ability. Agents are capable of performing 15 possible subtasks; so 15 abilities are generated for each agent. These values were assigned to the number of agents and subtasks because they represent a significantly difficult problem to address. We found that algorithm performance was relatively independent of these values. For ease of visualization, in these experiments each agent's ability is measured by two metrics, m_1 and m_2, both in the range 0-1 inclusive. The agent wishes to maximize both metrics, but the metrics conflict with one another. The profile of agent abilities is generated randomly at the start of an experiment. To simulate conflicting metrics, they obey the rule ${m_1}^2 + {m_2}^2 \leq 1$.

To evaluate the performance of the *E-Pareto* strategy, we set up a competition between an agent using *E-Pareto* and three other agents that are also based on evolutionary algorithms. Therefore, four agents compete for all tasks, each using a different technique to compute fitness for their evolutionary algorithm:

1. *E-M1:* Calculates coalition fitness based solely on its m_1 value, where m_1 for a coalition is the sum of the coalition agents' respective m_1 abilities for the subtasks they perform

2. *E-M2:* Calculates coalition fitness based solely on its m_2 value

3. *E-M12:* Calculates coalition fitness based equally on its m_1 and m_2 values: m_1 and m_2 are simply summed without weighting

4. *E-Pareto:* As described in Section 3

Thus, *E-M1* ascribes importance to m_1 only, *E-M2* ascribes importance to m_2 only, and *E-M12* ascribes equal importance to both metrics. *E-Pareto* adapts to changes in the importance of metric values. To our knowledge this problem has not previously been addressed; therefore *E-Pareto* cannot be tested against other algorithms that generate good solutions over the entire range of importance values. Instead, it is tested against three algorithms that represent previous approaches to this problem. These algorithms *E-M1*, *E-M2*, *E-M12* represent the optimal strategies for specific metric values. By placing *E-Pareto* in competition with the other evolutionary algorithms, we can examine how *E-Pareto* performs both in the range of metric values for which the competing evolutionary algorithms represent the optimal strategy and outside of these metric values. Each algorithm is purposely based on an evolutionary search algorithm because the results obtained can be attributed entirely to the means of coalition evaluation used, independently of the underlying search algorithm.

In equation (2), α is used to control the relative importance of $D(i)$ and $ND(i)$ when assigning fitness to coalitions. It was found that a value of 0.25 for α produced the best results. Lower α values focused too heavily on previously successful areas while higher α values focused too heavily on maintaining diversity.

Each algorithm generates an initial random population of 250 coalitions. We used a crossover probability of 70% and a mutation probability of 5%, as these settings were found to be reasonable for all four algorithms. Moderate changes to these values does not significantly degrade performance. Although we assume throughout this work that an agent operates with personal beliefs about the abilities of all other agents that might not be accurate, for these experiments we provide the four competing agents with identical beliefs about the abilities of all agents so that we can compare the performance of the algorithms independent of this information.

The competition between these four algorithms allows us to determine which performs best when faced with the task of calculating the optimal coalition. The metrics that the agents use, m_1 and m_2, are conflicting, of differing importance and the importance rating of each metric varies over time. We refer to the importance of m_1 and m_2 as MI_1 and MI_2 respectively. Initially, MI_1 is set to 0% and MI_2 is set to 100%. With the metric importance values held constant, 60 identical tasks are proposed in succession to each of the agents, with the tasks consisting of 15 subtasks. The agents then compete to obtain the task.

For each task received, the evolutionary algorithms run for 35 generations. We limited the number of generations to 35 as we found that the algorithms plateaued after this value. They each then propose the optimal coalition they calculated. The current metric importance weighting is used to determine which of the four coalitions proposed has the highest value. The m_1 and m_2 values for each coalition are calculated based on the profiles of agent abilities. The overall value of a coalition is determined as $m_1MI_1 + m_2MI_2$. Whichever coalition obtains the highest values is awarded the task. After all 60 tasks have been proposed, the number of tasks won by each algorithm for the current combination of metric importance is recorded.

Next, the importance of each metric is altered, incrementing MI_1 by 1% and decrementing MI_2 by 1%, and the cycle of proposing 60 tasks begin again. The experiment terminates once MI_1 reaches a value of 100%. The steady increase/decrease in metric importance is not a realistic assumption. However, the results we obtain would equally hold for metric importance that changes gradually up or down from step to step in a random walk. The reason we use a steady unidirectional change in metrics in these experiments is to allow us to evaluate the performance over all possible metric values.

4.2. Results

When evaluating *E-Pareto* we are interested in how it performs against the other evolutionary algorithms described in Section 4.1. Also of interest is the difference in performance of *E-Pareto* when using a purely greedy coalition selection strategy (Section 4.2.1) and when using the exploration-enabled coalition selection strategy (Section 4.2.2).

4.2.1. Greedy Coalition Selection

Figure 1 depicts the number of tasks obtained by each evolutionary algorithm throughout the duration of this experiment. The x axis represents the current value of MI_1 in terms of percentages. The corresponding value of MI_2 can be calculated as (100% - MI_1). The y axis represents the number of tasks won. Initially MI_2 has a value of 100%, therefore *E-M2* is immediately competitive. The reason for this is that it bases its fitness function on obtaining the coalition with highest value of m_2. However, the *E-Pareto* algorithm also proves to be as successful as *E-M2*. When MI_1 reaches 30% the performance of *E-M2* drops sharply until it is no longer competitive, as would be expected. Simultaneously the performance of *E-M12* rises sharply from being uncompetitive to competitive. Throughout this transition, *E-Pareto* remains constantly competitive. *E-M12* remains competitive between the MI_1 values of 30% until 70%. This behavior is expected, because *E-M12* considers both m_1 and m_2 to be of equal importance. Finally, as the performance of *E-M12* drops sharply when MI_1 reaches 60%, *E-M1* becomes competitive. *E-Pareto* remains consistently competitive independent of the changing metric values.

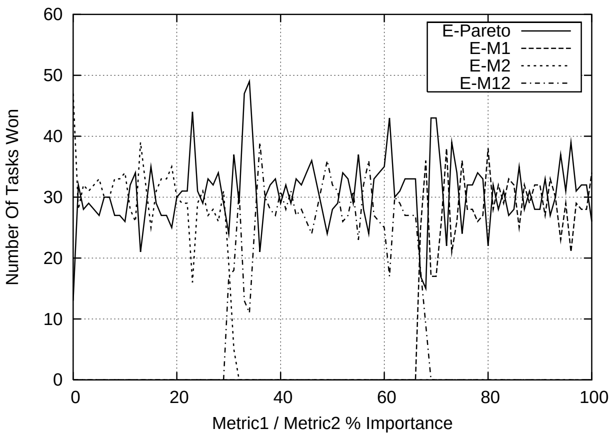

Figure 1. Number of tasks won by each evolutionary algorithm as the values of m_1 and m_2 values are varied. Exploration during coalition selection in *E-Pareto* is disabled.

Figure 2 represents a different perspective of the same experiment. We have mapped coalitions based on their percentage m_1 and m_2 values. Figure 2 maps every optimal coalition proposed by *E-Pareto* throughout the duration of the experiment. There are three areas of high concentration. The results of the experiment can clearly be divided into three sections, when *E-M2* is competitive with *E-Pareto*, when *E-M12* is competitive with *E-Pareto* and when *E-M1* is competitive with *E-Pareto*. These three areas of high coalition concentration reflect the three sections displayed in the results. Figure 1 shows that *E-M2* is initially competitive because MI_2 has a high value. Figure 2 shows that *E-Pareto* reacts to this, focusing on this area of competitive space for coalition selection. Hence, a high concentration of coalitions is evident in the top left corner of the graph. The two other areas of high concentration in Figure 2 can be explained in the same way.

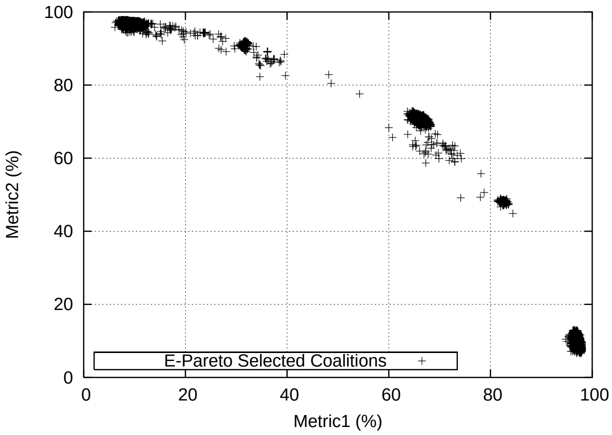

Figure 2. All coalitions proposed by the *E-Pareto* algorithm, without exploration during coalition selection, throughout the duration of the experiment.

4.2.2. Coalition Selection With Exploration

The same experiment run in Section 4.2.1 is run again with the difference that the *E-Pareto* algorithm uses the exploration-enabled strategy described in Section 3.3. From Figure 2 it can be seen that the *E-Pareto* algorithm focuses only on the currently competitive areas. Our hypothesis is that if the algorithm more fully explored the space available then it would identify areas with a high success rate. The results are presented in Figure 3. There are two main differences between these results and those obtained in Section 4.2.1.

The first is that the *E-Pareto* algorithm identifies two areas of space where it dominates its competition. As in the previous sub-section, *E-M2* is initially competitive with *E-Pareto*, however as MI_2 decreases the coalitions proposed by *E-M2* become weaker. The *E-Pareto* with a greedy strategy of coalition selection was unable to take advantage of this because it focused purely on the competitive area of space, where it was competing with *E-M2*. The exploration-enabled selection strategy allows the *E-Pareto* to identify areas of the solution space where it may be more competitive. This leads it to dominate all other evolutionary algorithms for certain values of metric importance in this experiment.

The second difference between this result and that obtained in Section 4.2.1 is that there is a competitive cost incurred for exploration. While *E-Pareto* still remains competitive with the other three algorithms, its level of performance has degraded slightly when compared with the performance it exhibited in Figure 1.

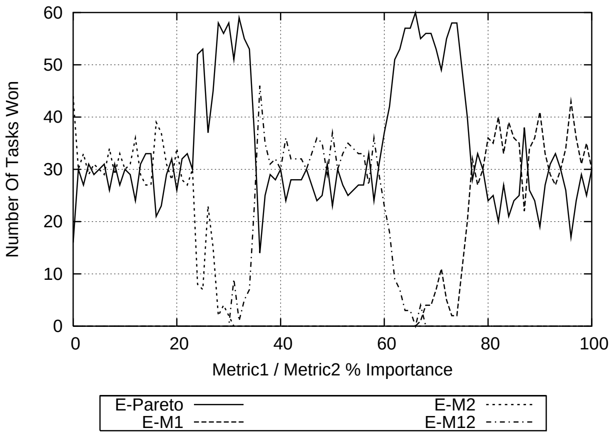

Figure 3. Number of tasks won by each evolutionary algorithm as the values of m_1 and m_2 are varied. Exploration during coalition selection is enabled in *E-Pareto*.

Figure 4 represents all optimal coalitions proposed by *E-Pareto* throughout the duration of the experiment. When contrasted with Figure 2, the exploration strategy demonstrates a thorough coverage of the solution space. Hence through exploration of the Pareto set, it is able to quickly identify and take advantage of shifts in metric importance.

5. Related Research

This paper addresses the topic of coalition formation. He et al. (2003) observes that it is a necessary technique for agents in e-commerce. However Shehory (2003a) comments that further improvements are required before coalition formation can become applicable for practical use.

Previous work in coalition formation has assumed that agents share an identical valuation for any given coalition (Shehory & Kraus, 1999), (Sandholm, 1999). Recently, coalition formation has been applied to the area of e-commerce. The assumption that agents share a common value for a given coalition is unrealistic in such an environment. Hence, each agent must calculate their own private value. Shehory (2003b) presents two alternative heuristics for evaluating a coalition, both of which are based on a single metric. In contrast with that, our work takes account of multiple conflicting metrics, differing metric importance and varying metric importance over time. Saha et al. (2003) assess the cost of asking another agent for help based on two metrics, time and quality. However, they treat all metrics as equally important, which is similar the *E-M12* strategy used in Section 4.

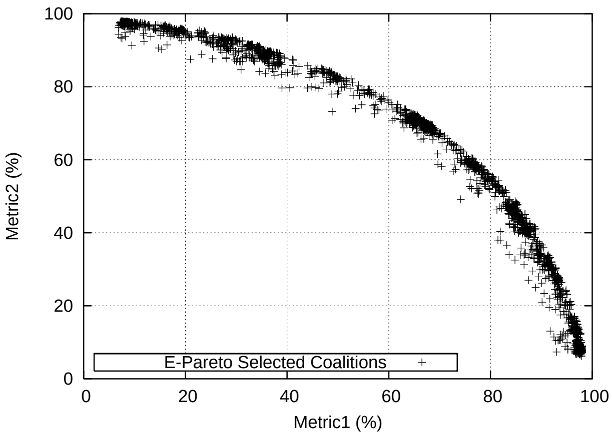

Figure 4. All coalitions proposed by the *E-Pareto* algorithm, using coalition selection with exploration, throughout the duration of the experiment.

Evolutionary algorithms have not often been applied in this problem domain. Sen and Dutta (2000) use a genetic algorithm to calculate the optimal coalition structure, but that is quite a different problem to the one we address, as it deals with the division of all agents in the system into one or more disjoint coalitions. Again it is assumed that agents share a common value for any coalition. Caillou et al. (2002) present a method of coalition formation which finds a Pareto optimal solution among all agents, but again that addresses a different problem. In their work, each agent's utility is measured with a single metric, and Pareto optimality is used to jointly maximize all agents' utility values, rather than to calculate an optimal coalition relative to multiple conflicting ability metrics. Our work does not present a mechanism to enable agents to negotiate to form coalitions, as theirs does; rather, we present a mechanism which will enable an agent to determine the optimal coalition in which to participate relative to the private information it has available to it.

6. Conclusion

This paper has considered the problem of *coalition calculation*, that is, calculating from a single agent's perspective the optimal coalition to propose for a task based on the information available to it. We have identified the problems associated with coalition calculation: conflicting metrics, differing metric importance and the variation of metric importance over time. To address these problems, an adapted multi-objective optimization evolutionary algorithm has been proposed, which enables an agent to identify the optimal coalition relative to the information it has about the world. We have shown through empirical evaluation that this approach is an improvement over previous techniques, and that when given conflicting metrics, it is capable of identifying the importance of these metrics, remaining competitive as the relative importance of metrics varies.

In the future, we aim to apply this mechanism of coalition calculation to a simulated market environment, where all agents maintain their own private information about other agents. We also hope to investigate the sensitivity of our approach to inaccuracy in the information held about other agents in such an environment.

Acknowledgments

The support of the Informatics Research Initiative of Enterprise Ireland is gratefully acknowledged. The authors are grateful to the anonymous reviewers for their detailed and helpful comments.

References

Caillou, P., Aknine, S., & Pinson, S. (2002). A multi-agent method for forming and dynamic restructuring of pareto optimal coalitions. *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems* (pp. 1074–1081). Bologna, Italy: ACM Press.

Dutta, P., & Sen, S. (2002). Emergence of stable coalitions via task exchanges. *Proceedings of the First International Joint Conference of Autonomous Agents and Multi-Agent Systems* (pp. 312–313). Bologna, Italy: ACM Press.

He, M., Jennings, N., & Leung, H. (2003). On agent-mediated electronic commerce. *Knowledge and Data Engineering, IEEE Transactions on*, *15*, 985–1003.

Saha, S., Sen, S., & Dutta, P. (2003). Helping based on future expectatioins. *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems* (pp. 289–296). Melbourne, Australia: ACM Press.

Sandholm, T. W. (1999). Distributed rational decision making. In G. Weiss (Ed.), *Multiagent systems: A modern approach to distributed artificial intelligence*, 201–258. Cambridge, MA, USA: The MIT Press.

Sen, S., & Dutta, P. (2000). Searching for the optimal coalition structure. *Proceedings of the Fourth International Conference on Multiagent Systems* (pp. 286–292). Boston, Massachusetts: IEEE.

Shehory, O. (2003a). Coalition formation: Towards feasible solutions. *Proceedings of the 3rd International Central and Eastern European Conference on Multi-Agent Systems* (pp. 218–251). Prague, Czech Republic: Springer-Verlag.

Shehory, O. (2003b). Coalition formation with uncertain heterogeneous information. *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems* (pp. 1–8). Melbourne, Australia: ACM Press.

Shehory, O., & Kraus, S. (1999). Feasible formation of coalitions among autonomous agents in nonsuperadditve environments. *Computational Intelligence*, *15*, 218–251.

Srinivas, N., & Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, *2*, 221–248.

Tsvetovat, M., Sycara, K., Chen, Y., & Ying, J. (2000). Customer coalitions in the electronic marketplace. *Proceedings of the Fourth International Conference on Autonomous Agents* (pp. 263–264). Barcelona, Catalonia, Spain: ACM Press.

Zitzler, E., Laumanns, M., & Bleuler, S. (2004). A tutorial on evolutionary multiobjective optimization. *Proceedings of the Workshop on Multiple Objective Metaheuristics*. Paris, France: Springer-Verlag.

Zitzler, E., Laumanns, M., & Thiele, L. (2001). *Spea2: Improving the strength pareto evolutionary algorithm* (Technical Report 103). Swiss Federal Institute of Technology, Gloriastrasse 35, CH-8092 Zurich, Switzerland.

Zitzler, E., & Thiele, L. (1998). *An evolutionary algorithm for multiobjective optimization: The strength pareto approach* (Technical Report 43). Swiss Federal Institute of Technology, Gloriastrasse 35, CH-8092 Zurich, Switzerland.