Optimisation in Multi-Agent Systems

I.Stengel[1,3], U.Bleimann[1,2] and J.Stynes[3]

[1]Center for Advanced Learning, Media and Simulation (igdv),
[2]Institute of Applied Informatics Darmstadt (aiDa),
University of Applied Sciences Darmstadt, Germany
[3]Department of Computing, Cork Institute of Technology, Ireland
e-mail: ingo.stengel@h-da.de

Abstract

Starting from a group forming scenario with a limited number of agents, a procedure to optimize performance is described. The approach defines an architecture to support competitive negotiation by combining an agent approach with a centralized approach. The procedure is evaluated using a case study. As a result the number of participating agents could be increased and the execution time could be reduced in limits.

Keywords

Multi-Agent Systems, Optimisation of a group forming scenario

1. Introduction

The motivation for this work lies in the need of supporting the learning group formation of students within a virtual university. Since agents due to their definition as computational systems on their own, are predestined to represent the interests of persons.

On the other side, group forming is considered to be an NP-complete problem, that cannot be solved within a finite amount of time, except the cases when the number of participants is small enough. The desire to implement it with an agent based approach lead us to an optimisation process that will help us to improve the number of agents to an acceptable amount, decreasing the execution time as well, and thus making this approach possible.

Optimization of a system always means optimizing the values of certain attributes or properties. An optimization for one algorithm may result in the deterioration of the result achieved by another. Therefore the problem to be solved must be the focus of the optimization. This can have several implications on e.g. the architecture, the methods, the communication etc.

2. Problem definition

The goal of this paper is to optimize an agent group forming scenario where agents are self-interested and competitive. Every agent *Ag* has a fixed number of independent characteristics: attributes ($attr_i(Ag)$), search-attributes the agent is looking for ($search_i(Ag)$) and priorities ($prio_i(Ag)$) used to model the importance of the search-attributes. The goal is to find an “overall good solution” from an agent view and from a central view.

A “good solution” should be able to distribute all agents into groups up to a maximum predefined size and offer the best possible solution based on the results of the preselection of possible partners and the concurrent negotiations between agents.

3. Discussion

Existing relevant approaches include centralised approaches, which calculate the optimal distribution into groups. This approach can be used to implement a centralised view on the problem. It is able to deliver optimal solutions, when the number of agents is limited. Since the complexity of the approach is exponential, this approach is only computable for a restricted number of agents.

In our context an agent based solution is able to approach the problem using views of different agents. Here again finding solution is restricted to a certain number of agents involved. If this number (above 15) is exceeded the problem cannot be computed in finite time (Sandholm et al.1999).

Although both approaches have limitations in the number of agents used, the behaviour is different. While centralised approaches usually need much time to calculate and compare the different possible groups and only later deliver solutions, the agent based approaches are able - up to a certain number of agents - to deliver fast results (65-90% of agents in groups in the first 50 seconds). However the remaining rest may need much more time. Here again, if a certain number of agents is exceeded the problem cannot be computed in a finite time and a flattening of the graph can be observed.

In our opinion it makes sense to combine both methodologies by taking advantage of their positive characteristics with the limitations known and thus increasing the number of possible agents involved. A further improvement can be expected by using heuristic approaches as suggested by (Shehory et al. 1998).

To implement this a proper architecture is needed. This has been found in the two layer negotiation architecture by (Zhang et al. 2003). Here two layers exist: the central solver layer and the agent layer. A closer look reveals that in this approach the groups are calculated by the central solver component, which is ignoring in the group forming context the strength of the agent technology: negotiation.

Our approach (Stengel et al. 2007) corrects this by moving the group forming task to the agents involved. They will achieve this by negotiating with potential partners. The central component calculates the lists of potential partners and transfer’s them to the agents in charge. The complexity of the calculation of the sorted lists is polynomial.

Further, in the agent structure negotiation units have been incorporated. They are used to support the competitive negotiation process by allowing concurrent negotiations. To reduce the complexity in this context an adaptive desperation threshold has been introduced. This threshold adapts the number of active concurrent negotiations. The initial number of negotiations is one. When no partner is found during a certain amount of time, the number of concurrent negotiations is increased using a predefined step that can take values up to a predefined number. This number is depending on the number of increases expected.

During the negotiation process every agent must be able to evaluate the offers received from other agents. Therefore a utility function (here Euclidian distance) is used during the decision process. It helps to decide how good agents are fitting into groups. (Schaefer, 2001)

[1] $d(Ag_1, Ag_2) = \sqrt{\sum_{i=1}^{k} prio_i(Ag_1) \times (search_i(Ag_1) - attr_i(Ag_2))^2}$

with $\sum_{i=1}^{k} prio_i = 1$

Further a harmony value H(g) of a group g is introduced. This value is oriented on the calculation of standard deviations. It is a medium value of deviations between the goals and the attributes of the members of a group. This shows how good the agents in a group are fitting together. d_{ij} is the distance between two different agents within the group g.

$$H(g) = \frac{\sqrt{\sum_{i=1}^{m} (\sum_{j=1}^{m} d_{ij}^2)}}{m \times (m-1)} \text{ for } i \neq j$$

It will be used to identify changes in the quality of the groups.

4. Methodology

To be able to optimize the performance of a multi-agent system in a group forming scenario, the behaviour of the system must be known. Measurements of a multi-agent system with competitive agents (Sims et. al. 2003) show that the expected behaviour will be similar to the one outlined in Figure 1.

This means that in the first time period [0, t_1], over 50% of the agents are already in groups. From this moment on, the speed in which the agents are finding partners, willing to form a group or join a group is drastically reduced.

The function *f(t)* represents the evolution of a solution using multi-agent negotiation exclusively. Optimization of the run of a multi-agent system with this characteristic can be done by finding a way to reduce the period from t_1 to the end of the execution.

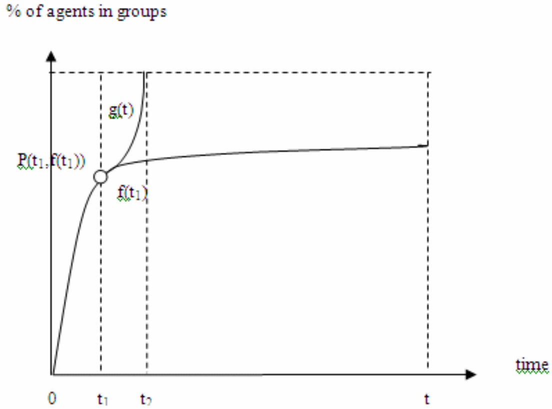

Figure 1: Introducing a second centralised approach

The goal is to reach a solution that is not worse than the one reached in normal time. Here, this was done by introducing a second centralised approach that redistributes the agents after the moment t_1.

The centralised approach calculates the distances between all remaining agents including representatives (Stengel et al. 2007) of a group. Then it redistributes the agents taking into account the harmony value of the group which is formed.

The second centralised procedure, shown in Figure 1, was characterised by the function *g(t),* which represents the progress of solutions over time by the central component (central solution). It needs less time than distributed procedures. In the period $(t_1, t_2]$, the agents that remained whithout a group at the moment t_1, are redistributed.

Essential to our optimization procedure is the determination **(1)** of the behaviour of the system over time, represented by the function *f(t).* In the next step the point $P(t_1, f(t_1))$ **(2)** for switching between the two procedures must be found. Finally the time in which the function *g(t)* reaches 100% **(3)** must be determined.

(1) Evaluating the Function *f(t)*

A feedback from all agents is used to evaluate the function *f(t)*. As soon as they join or form a group agents send a message to the central component (CC). This facility is for tracing only and allows the collection and evaluation of data by the CC with a short delay. Since the function *f(t)* differs from run to run the evaluation can be done using multiple runs with the same input data.

(2) Finding the Optimization Point P

The optimal moment to switch between the functions *f(t)* and *g(t)* is the moment when the gradient of the function *f(t)* has the fastest variation. Now, the speed with which agents are finding partners is decreasing as well. An approximation of the desired point P can be located by determining the distance between the point (0,100) and the points of the function *f(t)* (see Figure 2).

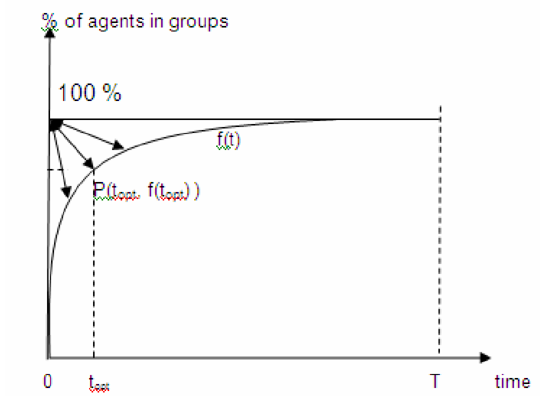

Figure 2: Finding the optimization point P

Wanted is the point $P(t_{opt}, f(t_{opt}))$ with
$dist\ (P(t_{opt}, f(t_{opt})), (0,100)) = min\ (dist(P(t, f(t)), (0,100)))$ for all t □ $[0,T]$.
This idea provides a good compromise between the number of agents in groups and the time.

The following approaches to evaluate the position of the point P are possible:

- **Multiple runs** can be evaluated as follows: For every run, the point P must be found. Using a histogram of all optimization points P of all runs over time, the time $t=t_{opt}$ with the most hits can be determined. When the optimization points are too distributed, a histogram of the optimization points in a moving time frame can be used to find t_{opt}.
- **The almost real-time approach** relies on the feedback that every agent that joins a coalition gives to the central component (CC). The CC can control the development of the characteristic over time. This allows the continuous calculation of the distance between P and the point (0,100). Once the shortest distance has been reached (in the next step, the distance is increasing), the conclusion can be drawn that at least a local minima has been reached. Now delayed, the multi-agent system can stop and switch to the second procedure. This approach is still faster than doing several runs and evaluating the optimization point.
- **The mixed approach with correction** uses both approaches described above. Details can be found in (Stengel,2008).

In this approach, the shortest distance was used to determine the optimized point P. This method is a heuristic approach because it delivers provable good runtimes but renounces to optimal solutions, delivering good solutions.

(3) Evaluation of the Function g(t)

The system switches from the distributed approach to the centralised approach at the point $P(t_{opt}, f(t_{opt}))$. Important is to determine the execution time of the centralised approach. This is done by measuring the run time.

5. Results

The behaviour of the system is not necessarily simply the sum of the agents' behaviours. As stated by (Wooldridge et. al. 1998), "the only way to find out what is likely to happen is to run the system - repeatedly" and analyze multiple runs with varying initial conditions.

The test runs are executed under the following conditions:

- Agent parameter values are generated using MatLab. They are normally distributed.
- The initial agent parameter values are identical in the multiple runs.
- Unless otherwise indicated, the agents have three attributes, search attributes and priorities.
- The multiple runs are executed using the same conditions.
- To evaluate the solution it is important to measure the performance in an objective way. We used the harmony value of a group (see (Stengel, 2008), section 4-3) which shows how large the average deviation between the goals of the group members is.

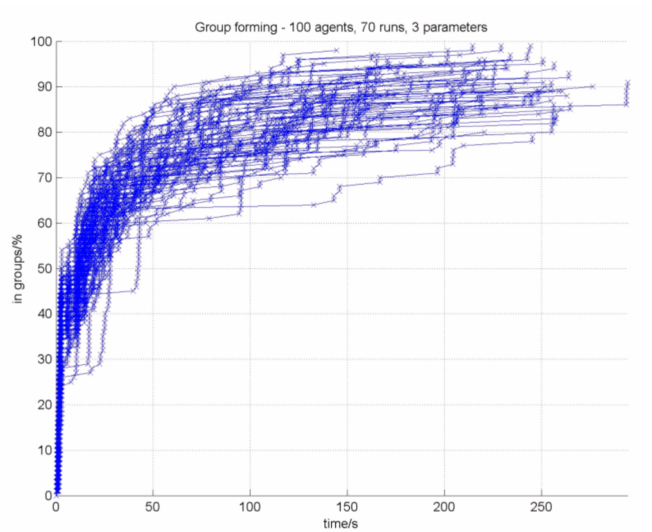

Figure 3: Multiple runs with 100 agents with 3 parameters in 250s

Figure 3 presents the results of 70 system runs using 100 competing agents with 3 parameters. The diagram shows the percentage of agents in groups vs. the time spent in group building. In the first period of about 50 seconds, a huge increase in the number of agents joining groups can be noticed. About 60-85% of the agents are in groups depending on the individual run.

In the first period, the number of concurrent agent negotiations is small due to the adapted initial threshold. Later, the threshold increases, leading to a bigger number of potential partners and thus, to a larger number of agents in groups. The theoretic range is of the threshold is (0,1]. The optimal value of the threshold step found in experiments is 0.025.

After the first period, e.g., in Figure 3 after 50 seconds, most of the agents are in groups and the remaining agents do not really fit in any groups.
A detailed discussion about the characteristics and the effect of factors like number of agents, threshold step etc. can be found in (Stengel, 2008).

Using the optimization method proposed, the distance from all points *P(t, f(t))* to the point (0,100) is taken into consideration. As can be seen in Figure 4 the time for switching between the distributed approach and the centralized approach for the example presented in Figure 3 has the value t_{opt} = 35 seconds. At this time, 65% of the agents are in groups. Here, delays generated when the system stops have not been taken into consideration.

The centralized approach was applied on the final data of the previous phases using the performance optimization method developed.

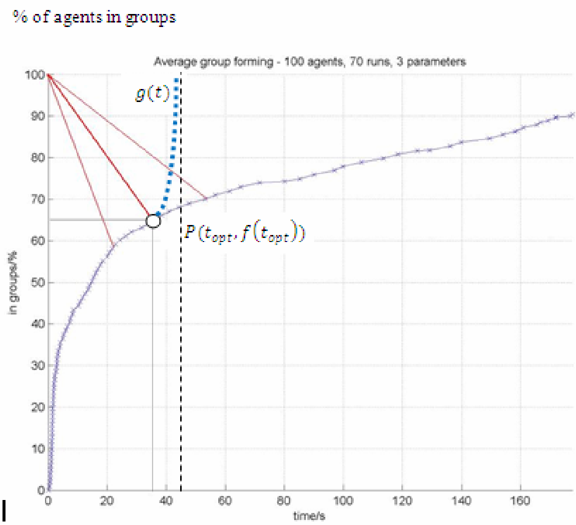

Figure 4: Estimation of the optimization point P using the distance method

The time needed for the execution of the centralized approach is estimated in the runs used for the calculation of the approximated function in Figure 4. The average value is 8,75 seconds per run, which is much shorter than the time needed using the agent-based approach. This demonstrates how good the method is starting from the time t_{opt}.

In Figures 5 a and b, the x-axis represents the distribution of agents in groups of different sizes (up to five members). The y-axis represents the number of groups added over multiple runs. The first histograms (red) show the distribution of the agents after the first approach at time 250 secs (Figure 5 a) and at t_{opt} = 35s. Applying the centralized approach, the agents that could not be distributed during the first approach,are now redistributed and represented in groups in the second histograms (green).

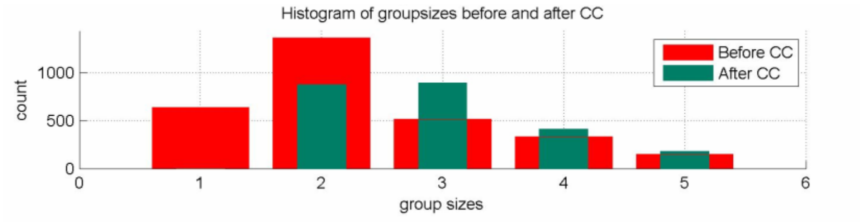

Figure 5 a: Histogram of the groupsize before and after the central approach (100agents, 250s, threshold step = 0.025)

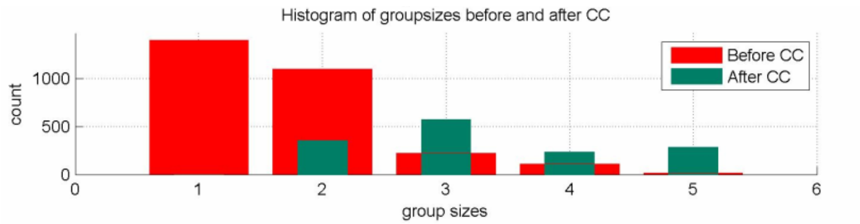

Figure 5 b: Histogram of the groupsize before and after the central approach (100agents, 35s, threshold step = 0.025)

In Figure 5 a, the CC will be used only after 250 seconds, i.e., the allocations in the first 250 secs are the result of agent negotiations. After the first phase, the majority of agents are already distributed into groups. When the CC is started, new groups of size three are formed by adding an agent to groups of size two. The number of groups of size four and five have increased only slightly. They can be formed by adding agents to groups from the next smaller size, or by merging groups.

When the system is optimized, the switch between the agent-based and centralized approach is done at time t_{opt} = 35s (see Figure 5 b). At this time, there are less agents in groups then after 250 secs (compare Figures 5 a and b.). CC is more effective since the number of agents to be redistributed is greater, and the number of groups of larger sizes increases faster, especially for groups of sizes four and five. Having more groups of larger size reduces the overall number of groups, implying less communication overhead since groups are represented by leaders. Thus, the system complexity could be reduced.

The early application of a centralized approach increases the number of groups with higher harmony values. The center of mass of the distributed groups moves to a better distribution to groups of larger size, reducing communication complexity. The number of groups with deteriorated harmony values increases only sparsely and can be ignored.

Applying this approach to a student learning group forming scenario, the following conclusions can be drawn:

All students could be placed in groups without a significant decrease in the harmony value, i.e., students are content within the group. In many of the groups these values actually improved.

The methodology presented improved the possible number of students significantly from theoretically 15 (Sandholm et al., 1999) to over 100. This allows now the use in a student learning scenario.

6. Related research

One of the most relevant papers regarding coalition structure was written by (Sandholm et al 1999). The paper dealt with worst-case guarantees and the authors confirmed that the optimal coalition structure is considered a NP-complete problem. Whenever the activity that generates coalition structures is resource-bound, it is too complex to find the optimal coalition structure. (Sandholm et al. 1999) stated that the number of possible coalition structures is so large that it cannot be enumerated until the total number of agents is below 15.

There are many differences between Sandholm's approach and ours: The goal in this work is to find a good overall solution, while the approach mentioned above seeks for the best solution. While the goal in this paper is to form groups, Sandholm uses groups to distribute tasks that are rewarded at the end. In our implementation the maximum size of a coalition is pre-defined and is much smaller than the total number of agents involved while Sandholm uses a number up to the "grand coalition".

Further approaches that try to reduce the complexity of the problem have been developed by Shehory and Kraus (Shehory et.al., 1998) and Remontino in (Remontino, 2004).

7. Conclusions

In this paper it has been shown that the heuristic approach proposed produces good results in the optimization of group forming using a multi-agent systems.

For a good optimization of our problem, it is essential that the centralized approach is applied only when the number of agents in groups reaches a certain percentage (usually over 50 %). Should the centralized approach be applied too early, e.g., when only 10 % of the agents are in groups, then the concurrent situation between agents is not considered and the costs of the centralized method will increase from seconds to a couple of minutes.

The systems used must be characterized by monotonically increasing (averaged) functions with faster group building processes at the beginning, similar to the expected function f(t) presented in Figure 1.

Using a heuristic approach combining two views – a centralised and an agent based view – the limit in the number of agent that can be used in a group forming scenario could be improved to a value over 100.

8. References

Lau, H.C. and Zhang, L., (2004), "A Two-Level Framework for Coalition Formation via Optimization and Agent Negotiation". *IAT 2004*: pages 441- 445.

Remondino, M. and Cappellini, A., (2004), "Minority game with communication: an agent based model", in *Proceedings 16th European Simulation Symposium, György Lipovszki, István Molnár* © SCS Press.

Sandholm, T., Larson, K., Andersson, M., Shehory, O. and Tohme, F. (1999), "Coalition structure generation with worst case guarantees", in *Artificial Intelligence No. 111*, Elsevier.

Schaefer, R. (2001), "Rules for Using Multi-Attribute Utility Theory for Estimating Users's Interests", *IST-1999-10688*, DFKI.

Shehory, O. and Kraus, S., (1998) "Methods for task allocation via agent coalition formation", Preprint Elsevier Science.

Sims, M., Goldman, C.V. and Lesser, V., (2003), "Self-Organization through Bottom-up Coalition Formation", *AAMAS'03,* July 14–18, 2003, Melbourne, Australia.

Stengel, I., (2008), "Optimisation in Multi-Agent System", PhD-Thesis submitted February 2008, Cork Institute of Technology, Ireland.

Stengel, I., Bleimann, U., Kreider, M., Kanold, M. and Stynes, J., (2007), "Multi-Agent Systems and Negotiation in a Group Forming Scenario", in *The proceedings of the 2007 European Simulation and Modelling Conference ESM'2007,* p452-457, Malta, October 22-24, 2007, ISBN: 978-90-77381-36-6.

Woolridge, M. and Jennings, N., (1998), "Pitfalls of Agent-Oriented Development, Dept. of Electrical Engineering", Queen Mary & Westfield College, University of London, London*, International Conference on Autonomous Agents, ACM*, p385-391.