Agreements without disagreements

Samir Aknine[1] and Philippe Caillou[2]

Abstract. The first part of this paper presents a coalition formation method for multi-agent systems which finds a Pareto optimal solution without aggregating the preferences of the agents. This protocol is adapted to problems requiring coordination by coalition formation, where it is undesirable, or not possible, to aggregate the preferences of the agents. The second part proposes an extension of this method enabling dynamic restructuring of coalitions when changes occur in the system.

1. INTRODUCTION

The European project *Princip* was initiated as an attempt to provide protection against racist and hate speech on the Internet. The practical goal of this project is twofold: firstly to set up a web crawler that will repeatedly look for racist documents, and secondly provide the list of identified racist sites to self-protection programs, either individual or collective. This article focuses on the first issue and, in particular, shows how multi-agent systems solve this problem. This approach proves its usefulness in combining several computational linguistic techniques on documents provided by information or meta-information search engines which find them using keywords and expressions. This article presents the coalition formation method we adopted to combine linguistic rules. This method is based on the principle of all agents making gradual concessions in order to reach a consensus on the coalitions to be formed.

To enable the agents to form coalitions, most current protocols make the assumption that the utility functions of agents, which measure their degree of satisfaction for each suggested solution, must be comparable or identical. This means that agents must be able to agree on a common utility function, either of all the agents as in [7], or of their coalition. This assumption seems acceptable for most multi-agent systems. However, in many cases comparing the utilities of agents, and even more so their aggregation, is delicate. The numerical measurement of the utility of an agent is already a strong assumption compared with the simple classification of available choices. Comparing the utilities of two individuals is stronger. A second limitation of current models lies in assuming that all calculations are recomputed as and when a condition changes (an agent joins or leaves a coalition, a task is added or canceled, etc.). However, these protocols are very complex and these changes can happen very frequently. Using the information obtained in a previous execution of the protocol, i.e. a dynamic reorganization of the coalitions formed, could decrease computations. This is the second aim of our models. More recent are the methods presented in [1][3] which are based on uncertain reasoning mechanisms. The principal differences with our former models [2] is that our models do not suppose that the utilities of agents must be aggregated or compared.

[1] LIP6, Université Paris 6, Aknine@lip6.fr
[2] LAMSADE, Université Paris Dauphine, caillou@lamsade.dauphine.fr

In this new model, we wish to improve our methods by relaxing some of their constraints. Our objective is to propose a coalition formation model which respects all the assumptions of our former models, but which also overcomes their limitations: (1) no order is imposed for the agents while participating in the coalition formation process; (2) equity for all the agents is guaranteed; (3) the agents are not obliged to trust each other.

The paper is organized as follows. Section 2 describes briefly the problem of document filtering in this project. Section 3 presents our coalition formation model, first the principles of this model, then the behaviors of the agents in the model. Section 4 analyzes its properties and presents the results of the experimental evaluation. Section 5 draws a general conclusion from this work and proposes some perspectives.

2. PROBLEM DESCRIPTION

The tracking of racist documents on the Internet comes up against a number of obstacles, which make it impossible to relay only on the classical keyword-based approach, or on neural network techniques, etc. Racist discourse spans from hate speech to more subtle insinuations, with different themes: racist, revisionist, anti-semitic, etc. Different kinds of discourses: political, historical, religious, etc. Some are related to organizations or churches, to quote: “The National Alliance, World Church of the Creator, Eastern Hammer skins, etc.", Different genres: pseudo-scientific articles, pamphlets, etc.

From the analysis of large sets of racist, anti-racist and non-racist documents, a number of candidate criteria for identifying racist content have been identified by the teams of linguists working on the project: (1) *Unique racial expressions* used only by racist people (but it is not possible to distinguish them from anti-racist documents), for example “Rahowa” standing for “Racial Holy War; (2) *Average frequencies* of certain words are not the same in racist documents. These words are not necessarily racist ones but more common words (like “their” or “white”); thematic words (e.g. words that denote fear of the multiplicity of the ethnic out-group, e.g. “multiply”); truth claims (words like “certain” or “fact”); (3) *Combined frequencies* of certain word pairs are relevant, e.g. the combination of “our” with “civilization”, “race” or “religion”; (4) *Suffixes* like “al”, “ence”, “ism” are good indicators for separating racist and anti-racist documents; (5) *Fonts* such as gothic, or some images, though they are never proof of racism.

There are many other similar features and most of them have been discovered by a comparative statistical study. However there are no clear indicators of racism on which one might rely to build a detection system. This is because there are no words or any other linguistic features that only racist people use. Hence we have to fall back on statistical analysis but, although this brings out differences between racist documents and non-racist ones, the weakness of the statistical approach is that it does not enable assertions about one single document, only about groups of documents. Two factors explain the global complexity of the system. Firstly, only the convergence of several criteria may be a good indication of racism, provided that there are no concomitant indications of anti-racism. Hence the number of criteria (several hundreds), their individual complexity, their correlations and relative relevance have an influence on the overall complexity. Secondly, the empirical factor has an important role: some criteria that seem conceptually close may have very different results, discriminating power, efficiency or computation speed. Finally, the multiple possible combinations of criteria may be more or less precise and efficient. But we do not possess any reliable theory or model to determine in advance the precision or the efficiency of such or such combination with respect to a given information retrieval goal. The aim of the multi-agent approach is to give a solution to criteria combination using coalition models.

3. COALITION FORMATION MODEL FOR CRITERIA-AGENTS

Among the problems stated previously, we have the problem of the complexity of processing the documents using more than one criterion. We have assimilated the combination of the linguistic criteria to a cooperation process between agents, and for that a dynamic coordination is necessary. In this section we present the coalition formation model used by the *criteria-agents*, that we associate to the linguistic criteria, in order to allow a better choice of the documents to which these will be applied. The role of a *criterion-agent* is to produce a set of information on the characteristics of a document using a set of processing and based on the information produced by other *criteria-agents*. The set of resources of the agents is called *R*. A criterion can be initiated if its resources provided by other *criteria* are available.

3.1. Why coalitions?

Each *criterion-agent* has some particular capabilities. It is programmed to analyze the document from a single viewpoint. These agents are not able to provide individually a definitive judgment on a document. This is why they must be combined dynamically. A possible way to coordinate them is to look for the agents that form the appropriate groups to analyze a document. In other words, we have to form coalitions of agents. A coalition is defined as a temporary association between agents in order to carry out joint goals. The aim is a better distribution of competences in order to achieve a complex goal. Coalitions are well adapted when there are strong externalities between the sub-tasks and/or when interactions between agents are such that the contribution of an agent within a coalition depends on which agents a coalition contains.

The protocol suggested is particularly suitable for problems with complex tasks (where there is a need for several agents and for coalitions) and for dynamic problems (tasks may be added, others canceled or modified, constantly) where the agents have different utility functions. Agents are self-interested. However they respect all the commitments to which they agreed. We assume that the utility functions of the agents are not known by the other agents and do not need to be cardinal, an ordinal utility is enough. Agents just need to be able to choose between two situations (or to be indifferent). This mechanism proceeds by making each agent build partial solutions, solutions that take its preferences into account better. These solutions are then converged, i.e. agents try to reconcile their points of view, by favoring concessions in order to reach a consensus on the coalitions to be formed. In this model agents therefore prefer to seek gradual and reciprocal concessions rather than to camp on a position which would not allow them to reach their objectives. The concessions are made gradually. They are not imposed only on one or two agents but on all of them in order to guarantee equity. Step by step, each agent is obliged to give way a little more so as to be able to meet or cross the positions of the others. An agent cannot continue to have the same position unless the others agree in order to avoid the coalition formation process failing. Concessions are therefore necessary made by all the agents. In the European project, this means that the *criteria-agents* should make concessions on the documents to which they will be applied first. Having one result from the application of a *criterion-agent* is not sufficient to have an opinion on a document, as we showed in section 2, and applying all the criteria to a document is impossible, as we have identified hundreds of criteria.

3.2. Coalition formation mechanism

3.2.1. Definitions

Coalition: a coalition is formed for each task. It contains one or more agents which will achieve the task (for instance, analyzing the content of a document with a particular parameter on the maximal duration of the analysis).
Coalition set: a coalition set represents a solution to the problem of coalition formation. It contains as many coalitions as tasks to be performed at a given moment.
Group of coalition sets: a group of coalition sets corresponds to several sets of coalitions brought together in order to be computed and transmitted collectively (for instance, several forms of organizations of the *criteria-agents*). In the rest of this article, it will be referred to as a group of sets or simply a group. When an agent computes a group of equivalent sets, this means that it is indifferent regarding all the sets.
Context: a set of unspecified parameters which must be stable during a negotiation step. This context is important, particularly, because the location of the documents changes every day since their content is illegal.
Utility function: this function is used to represent the preferences of the agents. It may be ordinal or cardinal. In this case, measuring the utility of a set of coalitions means comparing it with a reference situation which will be the same one throughout the negotiation.
Reference situation: in order for the agents to know if they have to accept a set of coalitions as a solution, they need to be able to compare it with what they are sure to obtain during the negotiation. This minimum is the reference situation. If there are already coalitions, it is the current situation, with possibly some changes in order to take into account new information. To be sure to find a solution after a negotiation, the reference situation needs to be feasible and to be the same for all the agents.
Acceptable set: we consider that a set is acceptable for an agent if it is preferred or equivalent to the current reference situation.
Pareto optimum: a Pareto optimum is a situation where it is not possible to improve the situation of an agent without deteriorating that of at least one other [4]. Graphically, for two agents a situation is optimal if no other situation is at the top right position (cf. fig. 1).
Signature of a set E_i, denoted *$Sig(E_i)$,* defines the set of *criteria-agents* that have approved this set.

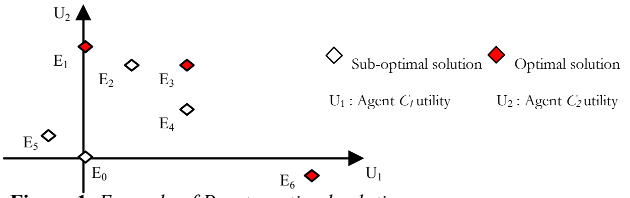

Figure 1. *Example of Pareto optimal solutions*

Knowledge $K(C_i)$ of an agent C_i on the sets defines the sets the *criteria-agent* knows and their signatories.

Unacceptable sets: the unacceptable sets $Out(C_i)$ for a *criterion-agent* C_i, are the sets $E_j \in K(C_i)$ that the *criterion-agent* knows and that it declares to the other agents as being unacceptable for it. If a set is declared by an agent as being unacceptable, it could not be proposed as a solution by any agent.

3.2.2. Principle

The agent which initiates a negotiation seeks one or more sets of coalitions it prefers and chooses an agent to which it sends them. The method for making the choice of the next agent to which to send them depends on the agent's strategies. The agent also declares the sets that are unacceptable to it. It may continue to seek the sets that it would choose as a second choice and send them, and so on, until there are no more sets at least equivalent to the current situation. Before sending its new chosen sets, the agent has to wait until it receives a message from another agent either about the former set it has proposed or about a new set proposed by this second agent. In the same way, each of the agents also starts by computing itself the sets that it prefers. It then computes the sets it would choose in second position and so on. When an agent receives a group of sets, it evaluates them. If there is at least one set which is preferable or equivalent to its best choice, it forwards this set to the next agent that it wishes to include in the negotiation. The agent signs this set and adds it to $Sig(E_i)$. This information indicates that this set has been approved by this agent but also by all the preceding senders. When the agent finds in the set it receives unacceptable sets, it has to declare them to the other agents in $Out(C_i)$. $Out(C_i)$ also contains the sets that itself identified and considered as unacceptable. This information is necessary, if the agents trust each other, because it avoids other agents computing sets that will be systematically refused. In the case of non-trust, agents do not transmit this information, so as to prevent the others using it in their strategies, because it would enable them to know which sets this agent could accept, and thus turn the negotiation to their advantage.

If a set is acceptable to the agent but its utility is inferior to its first preferred choice, the agent may nevertheless decide to forward it to the next agent. Consequently the number of agents having approved this set grows and the set is thus reinforced. It may also decide to temporarily block this set but indicates in $K(C_i)$ that it has received it, if it considers that there is still enough negotiating time to reach a consensus. In this case the agent sends to the next one the sets that it has preferred. When an agent receives a group of sets, if there is at least one set which is preferable or equivalent to the current situation and if all the agents have already approved this group, the set of this group that it prefers is a Pareto optimum and may be used as a solution set for the negotiation.

For instance, let us consider two agents and seven sets of coalitions. Let $E(U_1;U_2)$ be the relative utilities of agents C_1 and C_2 for the set E. Having E_0 as the initial situation, the seven possible sets are: $E_0(0;0)$; $E_1(0;10)$; $E_2(2;8)$; $E_3(4;8)$; $E_4(4;5)$; $E_5(-2;2)$; $E_6(10;-1)$. Of these seven sets, three are Pareto optima (E_1, E_3 and E_6). Figure 1 represents them according to their utility. Agent C_1 initiates the negotiation. It sorts all the sets it considers acceptable into equivalent groups of sets (Figure 2): $G_1(E_6)$; $G_2(E_4;E_3)$; $G_3(E_2)$; $G_4(E_0;E_1)$. E_5 is not sorted as the reference situation is better.

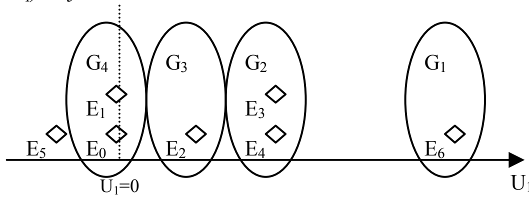

Figure 2. *Group of sets of agent C_1*

Groups G_1, G_2, G_3 and G_4 are acceptable to agent C_1 as they correspond to a situation which is as satisfactory as the initial reference situation, or better. In the same way, agent C_2 also searches for its preferred acceptable sets and sorts them into equivalent groups of sets (figure 3): $G'_1(E_1)$; $G'_2(E_2;E_3)$; $G'_3(E_4)$; $G'_4(E_5)$; $G'_5(E_0)$. E_6 is not sorted as the reference E_0 is better.

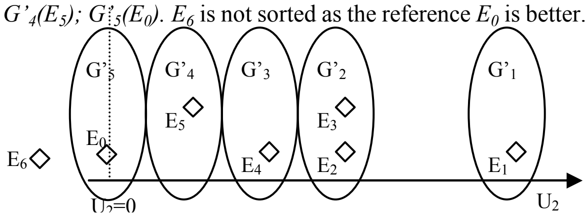

Figure 3. *Group of sets of agent C_2*

Groups G'_1, G'_2, G'_3, G'_4 and G'_5 are acceptable to agent C_2 as they correspond to a situation which is as satisfactory as the initial reference situation, or better. Thus agent C_1 starts by sending its first preferred group G_1. Agent C_2 starts by receiving G_1 and evaluates it (cf. figure 4). The unique set E_6 in G_1 is unacceptable to the agent because it would bring a less satisfactory situation than the initial situation. Agent C_2 does not send this set. It indicates to agent C_1 that it refuses the group $G_1(E_6)$.

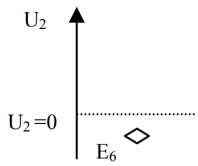

Figure 4. *Position of the first group received by agent C_2*

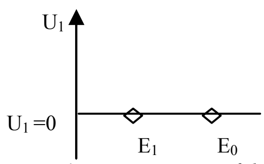

Figure 5. *Position of the first group received by agent C_1*

Agent C_1 has no other choice than to wait for a new proposal from agent C_2. Thus C_2 sends its preferred set, i.e. $G'_1(E_1)$. This group contains only set E_1 which is acceptable to agent C_1 but which corresponds at the same time to its final choice regarding its preferences. For E_1 the agent utility is the same as that of its reference situation E_0 (cf. figure 5). Agent C_1 does not refuse this set but it still has a possibility to propose a new group to agent C_2. C_1 thus sends its second preferred group $G_2(E_3; E_4)$. Agent C_2 receives G_2 and separates its two sets into two groups. E_3 of G_2 now belongs to its group G'_2 which corresponds to its second preferred group. As for set E_4, it belongs to group G'_3 that it orders as its third choice. E_3 is acceptable. As all the other agents have already participated in the negotiation, agent C_2 cannot send it to others. Set E_3 of G'_2 can thus be a solution. Agent C_2 has no other choice than E_3 which is Pareto optimal. Either it sends this set to agent C_1 in order to indicate to it the final result of the negotiation, or it waits until C_1 gives way on set E_1 which is also Pareto optimal considering the fact that agent C_2 has already refused one of the sets that agent C_1 proposed to it. The negotiation between these agents inevitably finishes on one of the two optima and before expiration of the duration of the negotiation decided initially.

3.2.3. Algorithm

The negotiation process is based on three phases: initialization of the negotiation and transfer of tasks, the negotiation proper, transmission of the solution. There is no pre-established sorting for the agents. They can intervene indifferently during the phases of the negotiation process and they can participate at any time. The only imposed condition is that an agent cannot communicate until it receives an answer to its previous ask.

- Phase 1: Initialization of the negotiation and transfer of tasks

Any agent can initiate the negotiation. This action may result when new documents to be analyzed are sent by the search engines or by a *criterion-agent,* for instance when it changes its preferences due to a negative evaluation by another *criterion-agent* on the documents that it is analyzing. In this project, *criteria-agents* are initially distributed into several levels according to their dependences, for instance, the domain of the documents, etc. When an agent initiates a negotiation it informs only the agents of its level. Any agent which wants to begin another negotiation must wait until the end of the negotiation in progress. To avoid conflicts between two simultaneous requests, each agent sends a confirmation. Each agent asks the other agents to send it their tasks. It therefore uses several operators in order to describe its choices in terms of both time and the resources that it needs. The receiver agent now has the set of tasks and can now define the coalitions. The initiator agent computes the first possible sets of coalitions, and gathers them in groups of sets in order to initiate the negotiation. Then it sends its first group of preferred sets that it signs in $Sig(E_i)$.

- Phase 2: Negotiation

Each of the other agents also computes its preferred sets. It then computes the sets that it will choose in second position and so on. When an agent receives a group of sets, it sorts the sets in order of preference into homogeneous new groups of sets. In these groups, all sets are equivalent in terms of agent utility. It updates its knowledge on the sets in $K(C_i)$. When the utility of a new group is equivalent to the utility of a group which is already known to the agent, these groups are not merged by the receiver agent since for the sender agent their utility is not the same. In these groups, all sets are equivalent in terms of agent utility. The agent sorts only those sets that are at least equivalent to the reference situation and the others are not considered. It declares them in $Out(C_i)$ at least to the next agent that it will contact, of course if the agents trust each other. In the contrary situation, it will not indicate them.

The agent chooses the next agent to which it sends a group of sets. It then sends to this agent its preferred group of sets, if it has not already sent it, otherwise it chooses the next group in its sorted preferences. This best group may possibly contain the sets that have been approved by the preceding agents that have participated in the negotiation as it can only contain new ones. The difference between the two is the number of agents that have approved these sets. If all the agents have participated in the negotiation, the agent sorts the sets into groups. If at least one of the sets is acceptable, it can consider it as its best group if it believes that the negotiation time has expired. All the sets of this group are Pareto optima, so it can arbitrarily choose one of them as a solution for the negotiation or it can continue the negotiation if it still has groups of sets to propose, and of course provided it has not previously declared them to others.

- Phase 3: Transmission of the solution

Once the last agent has identified a Pareto optimal solution which is approved by all, it sends this set to the other agents which accept it as the solution for the negotiation, as they have already confirmed it in $Sig(E_i)$.

3.2.4. Using heuristics to improve the complexity

In order to reduce the computation time, we have proposed several heuristics, for instance, to avoid that an agent starts by computing all the possible combinations for all the tasks, which can augment computational complexity of our model and the volume of data sent to the following agents. A way to improve the computation complexity without decreasing information quality, and thus the result and the properties of the algorithm, is to use and transmit undeveloped coalitions, i.e. the tasks for which all possible coalitions have not yet been computed. If an agent receives an undeveloped coalition in a set E_i and this coalition does not affect its utility (if it joins it or not), it leaves it aside and does not compute it. It just needs to approve it in $Sig(E_i)$. If it does, it computes the possible combinations for the corresponding task. Considering our assumption, the result of this computation is the same whatever the agent which does it. For instance, in our project a *criterion-agent* which initiates the negotiation will only develop the coalitions which concern the documents that it could analyze. They are the only ones that can improve its utility.

Other heuristics are also used, particularly doing tests on the acceptability of the sets, *search limited to the best group, limited search using intermediate evaluation, non-exhaustive methods, prospective search, i.e.* instead of starting from an empty set and developing it, an agent can use its knowledge of its utility function and the tasks to be achieved in order to deduce the best sets.

3.2.5. Importance of the agent execution order

The order in which agents negotiate does not have a real influence on the final result. The first agent can choose the solution that it wants to propose among all the possible sets, and to send it to the following agents. Off course the following agents are not obliged to agree with this solution. However the interest an agent has in making the right choice of next agent to be included in the negotiation process is that if this second agent accepts the same sets as those of its sender, the number of agents having approved the same sets would be reinforced more and more. It would be more difficult for another agent to propose a new set, in the following steps, if it estimates that the remaining negotiation time is short and that the probability of the other agents changing their positions is weak. It is still free to make this choice though. However, to improve the complexity of the protocol, it is preferable to take the agent which appears the most often in the computed sets. We assume that, since it takes part in many coalitions, this agent will be more interested in the alternatives which will be proposed than will be an agent which is less involved.

4. IMPLEMENTATION RESULTS

In order to evaluate this protocol, we cannot check if the utility function is maximal, as we assume that the multi-agent system has several utility functions that are incomparable. We have checked that during the tests we always obtain a result and that this result is a Pareto optimum. We have analyzed the performance of the protocol by observing several parameters: the number of messages exchanged, the size of these messages (the number of coalition sets they contain), number of coalition sets that have been evaluated and the runtime of the negotiation. The number of messages exchanged between the agents is independent of the search method strategy. However, their size depends on the use of undeveloped coalitions. As for the number of evaluated sets, it depends very much on the search method used. Four heuristics have been implemented. In our filtering system, *criteria-agents* are organized in several levels. In each of these levels agents form the coalitions. In the following, we will analyze these factors on simple examples.

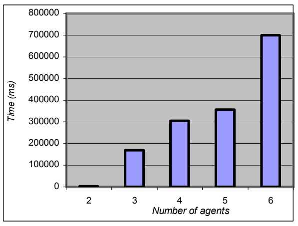

Figure 6: *Negotiation runtime*

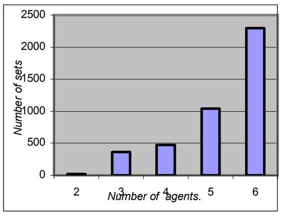

Figure 7: *Total size of messages sent*

Figure 6 shows negotiation runtimes obtained in milliseconds with the number of agents varying between 2 and 6 in one level of the *criteria-agents* structure. The negotiation runtime is increasing in this figure. This is due to the higher number of proposals that they would compute and exchange. However the search time is acceptable even when the number of agents is augmented.

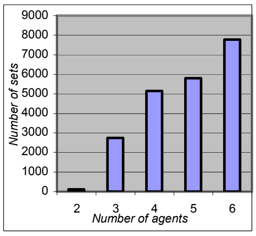

Figure 8: *Total size of evaluated messages*

Figure 7 shows the total size of the messages sent with the number of agents varying between 2 and 6. For instance, with 4 agents and 4 tasks, only 472 sets have been sent compared with the 6561 possible sets. The number of messages sent varies considerably according to the incompatibility of the preferences of the agents.

Figure 8 shows the number of evaluated sets and makes it possible to measure the effectiveness of the search heuristics of the best group. If the basic method is used, the first agent would simply evaluate 6561 possible sets and the number of evaluated sets will considerably increase and would exceed the 6561 sets. It is not the case in Figure 8 since only 5146 sets have been evaluated.

5. RELATED WORK

Current protocols in multi-agent systems are based on the following decomposition of the problem: generation of the coalitions, resolution of the optimization problem in each coalition and distribution of the created value between the agents. [7] proposes a simple and effective protocol, which can be applied in very general cases (recovery of coalitions, scheduling) and makes it possible to find the best solution. However, the protocol also implies that the value of the set of possible coalitions is calculated at least once. This gives a high complexity. [6] deals with this problem by proposing a method with a limited complexity while searching for a minimum result (with respect to the optimum result). [5] presents an analysis of the problems of having limits in computation capacity and proposes a terminology adapted to this type of problem. Zlotkin and Rosenschein have proposed a mechanism for coalition formation that uses cryptography techniques for sub-additive task-oriented domains. This mechanism is based on a Shapley value. The Shapley value is the expected utility that each agent will have from a random process [8]. However, this mechanism can only be applied to small-sized multi-agent systems because of its combinatorial complexity due to the computation of all possible coalitions. It is difficult to compare our protocol with current protocols since it does not have the same objectives. In current protocols, utility functions of the agents are systematically aggregated or adapted. On the contrary, the utilities here are neither aggregate nor transmitted. The results cannot thus be compared because they relate to different problems.

6. CONCLUSION

The protocol proposed is adapted to problems requiring coordination through the formation of coalitions and where it is not desirable, or possible, to aggregate the preferences of the agents. The protocol provides Pareto optimal solutions. In current protocols, utility functions of the agents are systematically aggregated or adapted. The method proposed extends our previous work [2]. Among other things, in the new model (1) the protocol does not require the agents to participate in the coalition formation mechanism in any particular order; (2) equity is respected, since the agents have to make gradual concessions so as to reach a solution that is acceptable to all. The model has been illustrated on a real application involving the search for and filtering of racist documents on the internet. It was then tested and evaluated, and has been shown to make a positive contribution.

We would like to thank ECCAI for his travel award, which allowed the presentation of this paper at the conference

7. REFERENCES

[1] B. Blankenburg, M. Klusch, and O. Shehory, "Fuzzy Kernel Based Coalitions Between Rational Agents", AAMAS, Melbourne, 2003.

[2] P. Caillou, S. Aknine, and S. Pinson, "Multi-Agent Models for Searching Pareto Optimal Solutions to the Problem of Forming and Dynamic Restructuring of Coalitions", ECAI, Lyon, 2002.

[3] S. Kraus, O. Shehory, and G. Tasse, "Coalition formation with Uncertain Heterogeneous Information", AAMAS, Melbourne, 2003.

[4] V. Pareto, *Cours d'économie politique*. Dioz, Switzrland, 1896.

[5] T. W. Sandholm and V. R. Lesser, "Coalitions among computationally bounded agents", *Artificial Intelligence*, vol. 94, pp. 99-137, 1997.

[6] T. W. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohmé, Coalition Structure Generation with Worst Case Guarantees", *Artificial Intelligence*, vol. 111, pp. 209-238, 1999.

[7] O. Shehory and S. Kraus, "Methods for task allocation via agent coalition formation", *Artificial Intelligence*, vol. 1998, pp. 165-200, 1998.

[8] G. Zlotkin and J. Rosenschein, "Coalition, Cryptography and Stability: Mechanisms for Coalition Formation Task Oriented Domains", AAAI, Seattle, 1994.