

Dynamic Coalition Formation among Rational Agents

Matthias Klusch and Andreas Gerber, *German Research Centre for Artificial Intelligence*

Designing self-interested autonomous software agents that can negotiate rationally in stable coalitions can dramatically benefit end users. Rational agents are usually required to form beneficial coalitions in open, distributed, and heterogeneous environments, including scenarios in which dynamically occurring events might interfere

with the coalition processes. Dynamic coalition formation (DCF) methods promise to be particularly well suited for applications of ubiquitous and mobile computing, including mobile commerce in wireless environments.

In mobile-commerce settings, for example, personalized information agents, each representing a potential business partner, might dynamically form temporary profit-oriented coalitions to enhance a customer's purchasing and negotiating strategies in multiple electronic marketplaces. This vision for the common Internet user isn't far from being realized, especially with recent advances in wireless computing and communication appliances.¹⁻³

Static formation of stable coalitions

We categorize coalition formation into two approaches: utility-based and complementary-based. These two models divide the actors into those that follow the principle of *bellum omnium contra omnes* as it is largely favored, for example, by game theory, and those that rely on the collaborative use of complementary individual skills to enhance the power of each agent to accomplish its goals.^{4,5} Until now, most classic methods for forming stable coalitions among rational agents follow the utility-based approach and rely on derived concepts from cooperative game theory, economics, and operations research. Utilitarian coalition formation covers two main activities: generating coalition structures and distributing gained benefit among the participants of each coalition.^{6,7}

We define a *cooperative game* as a set A of agents in which each subset of A is called a coalition and a

characteristic function v assigns each coalition C in A its maximum gain. (We offer an introduction to cooperative game theory here, but other sources provide a more in-depth explanation.^{8,9}) The value $v(C)$ does not depend on the actions of agents outside the coalition. Any coalition C forms by a binding agreement on the distribution of its coalition value $v(C)$ among its members.

The solution of a cooperative game with side payments is a *coalition configuration*, which consists of a partition S of A , the coalition structure, and an n -dimensional payoff distribution vector in which components are computed by a utility function u . The payoff distribution assigns each agent in A its utility $u(a)$ out of the value $v(C)$ of the coalition C in a given coalition structure S . The number or size of coalitions formed using a coalition formation method is often restricted to ensure, for example, polynomial complexity of the formation process.

In coalition configurations with so-called pareto-optimal payoff distributions, no agent is better off in any other valid payoff distribution for the given game and coalition structure. A coalition configuration (S, u) is stable if no agent has an incentive to leave its coalition in S due to its assigned payoff $u(a)$. Different characteristics and stability criterion define different solution spaces for cooperative games. Rational agents involved in a cooperative game (A, v) negotiate a stable payment configuration (S, u) as a solution that uses an appropriate coalition algorithm (CA). Each agent can execute the CA locally. Negotiation according to the CA is completely decentralized. The CA provides a stable coal-

Dynamic coalition formation (DCF) promises to be well suited for applications of ubiquitous and mobile computing. This article proposes a simulation-based DCF scheme designed to let rational agents form coalitions in dynamic environments.

tion configuration for any cooperative game at any time.

A coalition formation environment for a given set of agents A is the set of assumptions and constraints that are valid for any kind of coalition-forming activity between agents in A , including propositions on the task-related functionality of each individual agent in A , including its set of tasks, goals, actions, and methods to compute the individual utilities of task-related productions. The set of assumptions also includes valid methods for computing the values of coalitions and valid methods for determining coalition configurations, including methods for searching coalition structures.

In a given coalition formation environment, the agents agree on what kind of stable coalitions will be negotiated and what particular CA will be used for the negotiation. Different agents can compute their utilities of task execution and corresponding productions differently. However, most work on coalition formation relies on coalition formation environments in which all agents are homogeneous.

A coalition formation environment is super-additive or subadditive, depending on the type of all cooperative games it allows. In subadditive games, at least one pair of potential coalitions is not better off by merging into one. We define a coalition formation model by both the considered coalition formation environment and a given CA for this environment.

Stable and static coalitions

The meaning of coalition stability depends on the considered discipline and application domain. Many (if not most) of the coalition formation algorithms rely on chosen game theory concepts for stable payoff division within coalitions according to, for example, the Shapley value, the core, the bargaining set, or the kernel.⁸ All traditional approaches to coalition formation remain static in the sense that they do not allow for any type of interference with the running coalition formation process. In addition, known results for superadditive coalition formation environments must be transported into general or subadditive environments to gain practical relevance for the development and application of DCF algorithms to real-world open environments.

Core-stable coalitions

One approach¹⁰ to form stable coalition configurations consists of the following two

steps: searching for a coalition structure in a corresponding coalition structure graph for the given game (A, v) and then computing its payoff according to the stability concept of the core. We define the core of a game with respect to a given coalition structure as the set of coalition configurations that don't necessarily have unique payoff distributions. Only coalition structures that maximize the social welfare—the sum of all coalition values of coalitions in the considered structure—are core-stable.

However, searching for an optimal coalition structure (given a set A of agents) among the exponential number of $|A|^{A/2}$ possible coalition structures is computationally difficult, because we have to try at least $2^{|A|-1}$

In a given coalition formation environment, the agents agree on what kind of stable coalitions will be negotiated and what particular CA will be used for the negotiation.

coalition structures.^{10,11} Another well-known problem with core-stable configurations is that the core might be empty for certain cooperative games, and is exponentially difficult to compute. Because of these problems, using the core-stable coalition is quite unpopular. In fact, we aren't aware of any CA for computing core-stable coalition configurations in DCF environments.

Shapley-value-stable coalitions

Any payoff division scheme according to the Shapley value provides an agent with the added value that it brings to the given coalition structure, averaged over all possible joining orders. That makes the Shapley value exponentially hard to compute. Algorithms for forming stable coalitions that rely on the stability concept of the Shapley value, and a variation of it, the bilateral Shapley value¹² applied to arbitrary cooperative games, are proposed elsewhere.¹³ Computing a proposed payoff division according to the bilateral Shapley value with equal or proportional his-

tory-based share among coalition members is efficient and rational for superadditive games. Because this does not necessarily hold for subadditive games, these algorithms are not suitable for DCF.

Kernel-stable coalitions

The kernel of a cooperative game is the set of kernel-stable configurations (S, u) in which all coalitions in S are in equilibrium. Coalition C is in such an equilibrium if each pair of agents in C is in equilibrium—that is, if any pair of agents in C is balanced so that none of both agents can outweigh the other in (S, u) by having the option to get a better payoff. Each agent has to compare its surplus with those of other agents.

A game's kernel is exponentially hard to compute unless a constant limits the coalition's size. The kernel appears to be attractive because it is unique for any three-agent game, it assigns symmetric agents of some coalition in a given coalition structure for equal payoff, and it is locally Pareto-optimal. Polynomial CAs for polynomial kernel-stable coalition configurations have been developed and applied to the domain of cooperative information agents.^{14,15}

Fuzzy cooperative games

Negotiation during the coalition-forming process might be uncertain. Such uncertainties could be caused by the possibility of nondeterministic events that hamper the negotiation process and produce incomplete information. Agents might have uncertain knowledge about the share of coalition income in which they intend to participate or on the degree of their membership in one or multiple coalitions. An agent might determine the degree of its membership to potential coalitions by individually leveled commitments to other agents or bargains that indicate the degree of collaboration that the agents desire. The first case might imply the formation of fuzzy-valued coalitions, whereas the second case might induce the formation of fuzzy coalitions, which might partially overlap.

A fuzzy cooperative game¹⁶ consists of a set of agents, a fuzzy characteristic function v , and the membership function m of the fuzzy quantities $v(C)$ that might be interpreted as expectation of the common coalition profit that is to be distributed among its members. That is, the worth $v(C)$ of a fuzzy-valued coalition C is a fuzzy set of its possible real-valued coalitional profits. This set of fuzzy quantity

$v(C)$ has at least one modal value determined by the membership function m . If, for a given fuzzy cooperative game, the coalition value $v(C)$ is equal to one modal value of C for all possible coalitions C , it is equivalent to a (deterministic) cooperative game.

Stochastic cooperative games

Another class of cooperative games arises from cooperative decision-making problems in stochastic environments. A game with stochastic payoffs¹⁷ is defined by a set of agents, a set of possible actions coalitions might take, and a function assigning to each action of a coalition a real-valued stochastic variable with finite expectation, representing the payoff to a coalition when this particular action is taken. Thus, in contrast to a deterministic cooperative game, the payoffs can be random variables, and the actions a coalition can choose from are explicitly modeled, because the payoffs are not uniquely determined.

Developing DCF Schemes

We define the DCF research domain as the set of cooperation methods, schemes, and enabling technologies designed to cope with the problem of dynamically building beneficial coalitions among agents in open, distributed, and heterogeneous environments. The DCF problem must be solved in any collaboration environment and scenario in which agents enter or leave coalition formation processes and in which the set of tasks that individual agents must accomplish change dynamically. Cooperation scenarios inducing uncertain, time-limited, context-based utilities and coalition values can exacerbate the DCF problem.

One challenge is to get agents to react to different kinds of changes in real time without having to restart the complete negotiation process. Doing so requires agents to handle uncertain environment knowledge through appropriate adaptation mechanisms. Another research challenge concerns the transformation of the traditional game-theory criterion of coalition stability to these dynamic environments. It hardly makes sense for an agent to determine stable coalitions according to, for example, the Shapley value or the kernel in a frequently changing environment.

Basic research must clarify which kinds of dynamic settings, and to what extent available algorithms for the static formation of stable coalitions, should be adopted. In particular, the developed methods must let agents deliberately restart their coalition

negotiations at any time depending on the environmental changes. There is a trade-off between the efficiency of DCF and the quality of computed stable coalition configurations. Though this trade-off seems intuitively clear, it must be investigated further.

The development of DCF schemes might benefit from adopting appropriate methods for quantitative or qualitative decision-making that is based on partial, uncertain, and tentative information. Reasonable solutions for fuzzy and stochastic cooperative games might be adopted for cooperation schemes that let agents deal with different uncertainty types. Such uncertainties might be induced in DCF environments by, for example, network faults, changes of trust, or the receipt of vague or

Basic research must clarify which kinds of dynamic settings, and to what extent available algorithms for the static formation of stable coalitions, should be adopted.

even incomplete data during task execution.

There are no CAs for fuzzy or stochastic coalitions available to date. Developing such CAs for DCF environments appears to be even more challenging. To our knowledge, no such work is available to date. Other relevant work for developing cooperation schemes for dynamic environments include, for example, utility-based schemes for dynamically reorganizing organizational structures and exception-tolerant reasoning and multicriteria decision-making.

Social-reasoning mechanisms are essential building blocks suitable to situations in which agents might dynamically enter or leave the society without any global control. Advances in social reasoning have a clear impact on developing DCF schemes. Social-reasoning mechanisms are often based on the notion of social dependence or aim at reputation and trust management. To acquire and use dependence knowledge on the considered agent society, each agent must explicitly represent some properties of the other

agents, exploit this representation and optimize its behavior according to the evolution of the society, and monitor and revise its representation to avoid inconsistencies.

Reputation management aims at avoiding interaction with undesirable participants and might complement other security technologies for authentication and authorization. Mechanisms for building, propagating, measuring, and maintaining reputation and trust are useful to apply to settings for coalition formation among self-interested agents in e-commerce applications in which trusted third parties are required but not available. Merging several individual trust matrices, which are commonly used as a means to assess trust relationships among agents, requires further research. In general, mechanisms that let agents react on frequent changes of reputation ratings and assessment of trustworthiness of potential coalition partners are rare. Rational agents might face many potentially beneficial choices related to the timing of events that might occur during the individual decision process and the negotiation with other potential coalition partners. The preliminary results and experiences reported in relevant work might help design more complex methods for customer coalition formation in real time.

A simulation-based scheme

We designed the DCF-S scheme to help agents react to changes in their set of goals and in the agent society. We can instantiate the DCF-S scheme using different computational methods and negotiation protocols. Each instantiation yields a particular DCF-S-based CA. The development, implementation, and experimental evaluation of such DCF-S algorithms is part of our ongoing research efforts.

Environment

For our research, we assume a coalition formation environment in which agents continuously receive a set of goals from their users or other agents. Furthermore, any agent can freely enter or leave the society at any time. Each agent must use appropriate mechanisms to cope with the uncertainties and gradually adapt its decision-making techniques to changes in the environment. We assume that each agent is equipped with an appropriate learning component for this purpose.

We also assume an additional set of special agents, called *world-utility agents*. Any WUA might receive, compile, and maintain information about each of its registered agents.

DCF-S Scheme Functions, Numbers, and Sets

The local knowledge base of an agent a consists of $GS(a)$, the set of goals the agent must accomplish. Interleaved goals are aggregated by a into one goal. The list $CLIG$ (*BestCLIG*) contains the candidates with which agent a might coalesce to accomplish a goal G in $GS(a)$. The list $ACLIG$ of agent information records contains information about agent a on the capabilities of other agents a' with respect to G . Each record stores a finite-dimensional vector of real-valued attributes of an agent a' with respect to its estimated value of contribution to the accomplishment of goal G in $GS(a)$. Goal-related attributes of agent a' concern, for example, the estimated amount of its available resources, costs, quality, and efficiency with respect to goal G .

Other attributes of a' concern its reliability and trustworthiness in cooperation. The attributes include the following:

- The real value $crv(a', ACLIG, C)$ in $[0,1]$ denotes the risk of agent a to cooperate with agent a' in coalition C for goal G in $GS(a)$ with respect to the information on a' in the list $ACLIG$.
- The real value $crl(a', C)$ denotes the worst acceptable risk of agent a to cooperate with agent a' in coalition C .
- The real value $rrl(a', C)$ in $[0,1]$ denotes the worst acceptable risk of agent a to remove agent a' from valid coalition C the agent a is leading. This risk value might be computed with respect to the implied payment of trust penalty $tp(a', a)$ and the penalty payment limit $ppl(a)$.
- The real value $tp(a', a)$ is the trust penalty to be paid by a to a' in case a breaks a coalition agreement with a' .
- The real value $ppl(a)$ denotes the upper limit of penalty payments by agent a .

In addition, we define the following values, sets, and functions:

- The integer value $MaxSim$ denotes the maximum number of steps for each simulation round.
- $CCLIG$ is the set of candidates for forming a coalition with respect to goal G . These candidates come from the current set CS of valid coalitions and determined by the function $Match$.
- $Match(CS, G, ACLIG)$ determines the set of agents in the set CS of all trusted agents (individual agents and valid coalitions which are actually known to agent a), each of

which is capable of contributing to the accomplishment of goal G . The capability-based matching determines to what degree the agents' capability descriptions in $ACLIG$ match the description of the goal G .

- $Request(ACLIG, WUA)$ and $Update_Agt_Information(ACLIG, RecentAC)$ concern the request of the nearest world-utility agent for information to (periodically) update the list $ACLIG$. The update relies, in particular, on an appropriate learning mechanism to approximate incomplete or vague information.
- $SelectAgt_MinRisk_MaxValue(CC, HCIG, ACLIG)$ returns an agent a' from the set CC of agents with estimated minimum risk of cooperation in, and maximum value of contribution to, the (simulated) joint coalition $HC \cup \{a'\}$ with respect to goal G regarding the attributes of a' stored in the agent information list $ACLIG$. Only agents a' are selected which payoff in $HC \cup \{a'\}$ is individual rational.
- $SelectAgt_MaxRisk_MaxValue(HCIG, ACLIG)$ returns an agent a' from the coalition $HCIG$ with estimated maximum risk of cooperation in, and maximum value of, the coalition $C \cup \{a'\}$ regarding the information on a' in $ACLIG$.
- $Events(BestCLIG)$ returns the set of events that have occurred, influencing the coalition's formation, which consists of all agents in the list $BestCLIG$.
- $Value(CLIG)$ determines the value $v(C)$ of the coalition C , which consists of all agents in the list $CLIG$.
- $BilateralNegotiation(a, a', Value(BestCLIG), C \cup \{a'\})$ returns true if the bilateral multi-attribute negotiation of agent a with agent a' on a joint coalition $C \cup \{a'\}$ with respect to its value is successful; otherwise, it returns false.
- $Evaluate(ACLIG)$ updates the agent information list $ACLIG$ according to the local evaluation of the recent negotiation processes of the agent and returns the updated list $ACLIG$. This evaluation gives input to the agent's internal learning mechanism for adapting to changes in its environment.
- $StopNegotiation(BestCLIG)$ stops all running negotiation processes with all agents in $BestCLIG$ on a coalition for the goal G and updates the list $CLIG$ by keeping those agents with which the agent has already successfully negotiated.
- $RedundancyCheck(BestCLIG)$ returns a nonredundant list $BestCLIG$.

This information includes statements on an individual agent's problem-solving capability and the evaluation of its quality of service by other agents. Such evaluation might concern an agent's reliability and trustworthiness and affect its cooperation with other agents. These evaluation records are safe against possible manipulation and are securely distributed to and updated by the networked WUAs. However, each agent in the considered agent society is free to request its nearest WUA to obtain information on its environment.

We define a goal-oriented cooperative game $(A, v)|G$ as a cooperative game with respect to a given goal G . Such a game is determined by a given set A of agents and a real-valued function v assigning each coalition C its total expected outcome with respect to the accom-

plishment of the goal G . In particular, computing the individual utility of the set of productions of coalition members in C is restricted to the set of productions related to G .

We can represent any coalition to the outside world with an appropriate coalition-leading agent (CLA). We consider each coalition to be one entity or agent. Because we can consider one agent to be a single-agent coalition, we can use the terms "agent" and "coalition" interchangeably. Initially, the set of all possible, nonempty coalitions is the set of single-agent coalitions. Each agent is a CLA of a stable coalition for accomplishing one of its goals as a solution of the corresponding game. Any CLA is supposed to act on behalf of the members of its coalition, including negotiating and controlling the dis-

tribution of resources and payoffs among the coalition members according to the coalition contract. This structure is similar to the structure of holonic multiagent systems.

DCF-S scheme

In the DCF-S scheme, each CLA concurrently simulates, selects, and negotiates coalitions, each of which is able to accomplish one of its goals with an acceptable ratio between estimated risk of failure and individual profit. Figure 1 outlines the scheme in pseudocode, and the "DCF-S Scheme Functions, Numbers, and Sets" sidebar defines the necessary concepts. We can summarize the main steps of the DCF-S scheme executed by each CLA as follows: preparation, simulation, negotiation, and evaluation.

```

for each G in GS(a) do concurrently until external termination
{ halt:= false;
while not halt do
{

Preparation
CCIG = ∅; CLIG, LastCLIG, BestCLIG := null; op:=""; z,penalties:=0;
if periodic(date, ACLIG) then
{RecentAC := Request(ACLIG, WUA); ACLIG := Update_Agt_Information(ACLIG, RecentAC);}
CCIG := Match(CS, G, ACLIG); HCIG:=CLIG;

Simulation
for z:= 1 to MaxSim do
{ op := Random({noop, add_agent, remove_agent}); LastCLIG:= CLIG;
if op = add_agent then
{ agt:= Select_Agt_MinRisk_MaxValue(CCIG, HCIG, ACLIG);
if crv(agt, ACLIG, HCIG) ≤ crl(agt, HCIG) then{CLIG:= CLIG + (agt, add); HCIG:=HCIG∪{agt};}
else
if op = remove_agent then
{ agt:= Select_Agt_MaxRisk_MaxValue(HCIG, ACLIG);
if crv(agt, ACLIG, HCIG) > rrl(agt, HCIG) then { CLIG := CLIG + (agt, remove); HCIG:= HCIG\{agt}; penalties:=
penalties + tp(agt, a)
}
}
}
if Value(CLIG) > Value(LastCLIG) then BestCLIG:= RedundancyCheck(CLIG);
if Value(BestCLIG) >> v(CIG) && penalties < ppl(a) && Events(BestCLIG) =∅ then halt:= true;
}

Negotiation
for each (a', op) in BestCLIG do concurrently
{ try
if op = add then {if BilateralNegotiation(a, a', Value(BestCLIG), CIG∪{a'}) then CIG:= CIG ∪ {a'}}
else { CIG:= CIG \ {a'}; penalty_payment( tp(a', a), a'); }
catch(event: if Events(BestCLIG) <>∅ then { StopNegotiation(BestCLIG); Goto (Evaluation) });
}

Evaluation
EvalRes:=Evaluate(ACLIG); [if desired the Send(EvalRes, WUA);]
}
    
```

Figure 1. The DCF-S scheme in pseudocode. Each coalition-leading agent a executes the steps illustrated here, where CIG is a valid coalition led by a for one of its goals G in $GS(a)$.

In the preparation phase, the CLA determines the set of goals to be accomplished in cooperation with other agents and periodically updates its knowledge of the environment. The local knowledge base includes information on the partially known problem-solving capabilities of other agents as well as individual evaluations of past collaborations with these agents. To obtain this information, the CLA might request its nearest WUA. Because this environment knowledge might be incomplete or vague, the agent uses appropriate learning mechanisms for approx-

imating the needed information.

In the simulation phase, the CLA simulates the formation of coalitions, each of which might be able to accomplish a given goal with an acceptable ratio between the estimated individual profit and risk of forming the coalition.

In the negotiation phase, the CLA negotiates all coalitions it has determined in the previous simulation step. The CLA negotiates each goal-oriented coalition bilaterally with each potential member of the coalition. The complete set of negotiation sequences

can be performed concurrently. The result of a successful negotiation is a binding agreement between agents on the constraints and attributes of their cooperation in the new coalition.

In the case that one bilateral negotiation fails or an event changing the value or structure of the considered coalition is detected, the negotiation process for that coalition is immediately halted. The CLA then evaluates the negotiation process for this coalition and restarts the simulation of potential coalitions for the particular goal. For the restart, it keeps those agents in its coalitions with which it has already successfully negotiated and considers the current situation of the environment. This way, the CLA might avoid a complete restart, thereby avoiding possible penalty payments for removing agents from valid coalitions and a corresponding decrease of its reliability.

In the evaluation phase, the CLA evaluates its recent negotiations and reports these evaluations to the nearest WUA for distribution. Concurrently, it controls the distribution of payoffs and resources to members of the newly formed coalitions according to the successfully negotiated contracts.

Discussion of the scheme

In the DCF-S scheme, each agent simulates, selects, and negotiates coalitions, each of which is able to accomplish one of its goals with an acceptable ratio between estimated risk of failure and individual profit. In other words, the agents strive to solve a set of single goal-oriented cooperative games $(A, v) \setminus G$ by forming potentially overlapping coalitions with stable payoff distributions. Each of these goal-oriented cooperative games might change at any time subject to different kinds of nondeterministically occurring events such as agents leaving or entering the society. Each detected change can induce new cooperative games for the agents to solve.

According to the DCF-S scheme, each CLA reacts to these changes through a partial rather than complete restart. The agent tries to keep those agents in the affected coalitions with which it has already successfully reached a coalition agreement. However, the DCF-S scheme does not guarantee in general an optimal solution to these games. Rather, the agents continuously approximate the best solutions given their current knowledge of the dynamic environment. A different, but similarly opportunistic and high-risk DCF scheme, has been proposed elsewhere.¹¹

The DCF-S scheme assumes the existence of a set of networked WUAs, which each agent in the society is free to contact for obtaining needed information on the environment. In addition, the update of local knowledge by an agent is assumed to use results from a continuous adaptation process to approximate the needed information on its environment. That might improve the quality of its decision-making independent from the WUAs, and thereby reduce the overall complexity in computation and communication.

The complexity of any DCF-S based CA as an instantiation of the DCF-S scheme largely depends on the complexity of the implemented methods that the designer chooses for capability-based matching, learning, selection, and negotiation. For example, for low-complexity computation of stable payoff distribution in superadditive environments, we propose adopting the distribution according to the bilateral Shapley value with equal or proportional share among coalition members.¹⁵

Application-specific instantiations of the DCF-S scheme lead to the development of different DCF-S-based CAs. For this purpose, relevant approaches and theoretical work stemming from different disciplines are available, including work on temporal social reasoning, machine learning, and fuzzy and stochastic cooperative games.

DCF algorithms promise to be particularly well suited for applications of ubiquitous and mobile computing, including mobile commerce in wireless network environments. However, further basic research is needed to investigate the potential of the new research field of DCF, which remains in its infancy. ■


References

1. M. Tsvetov and K. Sycara, "Customer Coalitions in the Electronic Marketplace," *Proc. 4th Int'l Conf. Autonomous Agents*, ACM Press, New York, 2000, pp. 263–264.

The Authors



Matthias Klusch is a senior researcher in the Multiagent Systems Research Group at the German Research Center for Artificial Intelligence. He is also assistant professor in the department for Artificial Intelligence at the Free University of Amsterdam, Netherlands. His research interests include the application of AI and agent technology to databases and intelligent information systems. He received a PhD in computer science from the University of Kiel, Germany. Contact him at dfki.de, Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany; klusch@dfki.de; www.dfki.de/~klusch.



Andreas Gerber is a researcher in the Multiagent Systems Research Group at the German Research Centre for Artificial Intelligence. His research interests include distributed artificial intelligence, agent-based coordination mechanisms for electronic markets, and integrated services for e-business. He received a diploma from the University of the Saarland. Contact him at dfki.de, Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany; agerber@dfki.de; www.dfki.de/~agerber.

2. J. Yamamoto and K. Sycara, "A Stable and Efficient Buyer Coalition Formation Scheme for E-Marketplaces," *Proc. 5th Int'l Conf. Autonomous Agents*, ACM Press, New York, 2001, pp. 576–583.

3. O. Shehory, "Optimality and Risk in Purchase at Multiple Auctions," *Proc. 5th Int'l Workshop Cooperative Information Agents (CIA 2001)*, vol. 2, 182, Springer-Verlag, New York, 2001, pp. 142–153.

4. R. Conte and J.S. Sichman, "DEPNET: How to Benefit from Social Dependence," *J. Mathematical Sociology*, vol. 20, 1995, pp. 161–177.

5. R.D. Luce and H. Raiffa, *Games and Decisions: Introduction and Critical Survey*, John Wiley & Sons, New York, 1957.

6. S. Kraus, "Negotiation and Cooperation in Multi-agent Environments," *J. Artificial Intelligence*, vol. 94, nos. 1–2, 1997, pp. 79–98.

7. G. Vauvert and A. El Fallah-Segrouhni, "Coalition Formation among Strong Autonomous and Weak Rational Agents," *Proc. 10th European Workshop Modelling Autonomous Agents in a Multi-agent World (MAAMAW)*, Springer-Verlag, Heidelberg, 2001.

8. J.P. Kahan and A. Rapoport, *Theories of Coalition Formation*, Lawrence Erlbaum Associates, London, 1984.

9. M.J. Osborne and A. Rubinstein, *A Course in Game Theory*, MIT Press, Cambridge, Mass., 1994.

10. T. Sandholm, "Distributed Rational Decision Making," in *Multiagent Systems: A Modern Introduction to Distributed Artificial Intelligence*, MIT Press, Cambridge, Mass., 1999, pp. 201–258.

11. L-K. Soh and C. Tsatsoulis, "Real-Time Satisficing Multiagent Coalition Formation," *Proc. Int'l AAAI Workshop Coalition Formation in Dynamic Multiagent Environments*, AAAI Press, Menlo Park, Calif., 2002.

12. S. Ketchpel, "Coalition Formation among Autonomous Agents," *Proc. European Workshop Modeling Autonomous Agents in a Multi-agent World (MAAMAW 93)*, Springer-Verlag, Heidelberg, Germany, 1993, pp. 73–88.

13. M. Klusch and O. Shehory, "Coalition Formation among Rational Information Agents," *Proc. European Workshop Modeling Autonomous Agents in a Multi-agent World (MAAMAW 96)*, *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Heidelberg, Germany, vol. 1,038, 1996, pp. 204–217.

14. M. Klusch and O. Shehory, "A Polynomial Kernel-Oriented Coalition Formation Algorithm for Rational Information Agents," *Proc. Int'l Conf. Multi-agent Systems*, AAAI Press, Menlo Park, Calif., 1996, pp. 157–164.

15. J. Contreras, M. Klusch, and J. Yen, "Multi-agent Coalition Formation in Power Transmission Planning: a Bilateral Shapley Value Approach," *Proc. 4th Int'l Conf. Artificial Intelligence Planning Systems*, AAAI Press, Menlo Park, Calif., 1998, pp. 19–26.

16. M. Mares, *Fuzzy Cooperative Games-Cooperation with Vague Expectations*, Physica Verlag, 2001.

17. J. Suijs et al., "Cooperative Games with Stochastic Payoffs," *European J. Operational Research*, vol. 113, 1999, pp. 193–205.

For more information on this or any other computing topic, please visit our digital library at <http://computer.org/publications/dlib>.