

# Sharing Belief in Teams of Heterogeneous Robots

Hans Utz<sup>1</sup>, Freek Stulp<sup>2</sup>, and Arndt Mühlenfeld<sup>3</sup>

<sup>1</sup> University of Ulm, James-Franck-Ring, D-89069 Ulm, Germany

<sup>2</sup> Technische Universität München, Boltzmannstr. 3, D-85747 München, Germany

<sup>3</sup> Technische Universität Graz, Inffeldg. 16b/II, A-8010 Graz, Austria

[hans.utz@informatik.uni-ulm.de](mailto:hans.utz@informatik.uni-ulm.de), [stulp@in.tum.de](mailto:stulp@in.tum.de), [muehlenf@igi.tugraz.at](mailto:muehlenf@igi.tugraz.at)

**Abstract** This paper describes the joint approach of three research groups to enable a heterogeneous team of robots to exchange belief. The communication framework presented imposes little restrictions on the design and implementation of the individual autonomous mobile systems. The three groups have individually taken part in the RoboCup F2000 league since 1998. Although recent rule changes allow for more robots per team, the cost of acquiring and maintaining autonomous mobile robots keeps teams from making use of this opportunity. A solution is to build mixed teams with robots from different labs. As almost all robots in this league are custom built research platforms with unique sensors, actuators, and software architectures, forming a heterogeneous team presents an exciting challenge.

## 1 Introduction

Due to scientific as well as pragmatic reasons, there is a growing interest in the robotics field to join the efforts of different labs to form mixed teams of autonomous mobile robots. In RoboCup, the pragmatic reasons are compelling. The recent rule change in the F2000 league allows for more robots per team, and in the RoboCup Rescue league a group of heterogeneous robots with diverse capabilities is likely to perform better than one system that tries to encapsulate them all. However, the limited financial resources and the additional maintenance effort for further robots exceeds the capabilities of many research labs. Also, the threshold for new research groups to participate in RoboCup is lowered if they only need to contribute one or two robots to a mixed team, instead of having to build an entire team. Mixed teams are also motivated from a scientific perspective. They introduce the research challenge of cooperation within teams of extremely heterogeneous autonomous mobile systems.

As most robots in the F2000 league are custom built, or at least customised commercial research platforms with unique configurations of actuator and sensor configurations, mixed teams from different laboratories are extremely heterogeneous. There are few commonly used high level libraries for sensor data processing and reactive actuator design in the community. Furthermore there is a

multitude of methods and schools, each deliberately designing the control architecture of their robots fundamentally different to their competitors. This makes the unification of the software of the different robots of a potential mixed team almost impossible without substantial rewriting of at least one of the team’s software. In our opinion it is also undesirable. Why should an autonomous mobile robot have to commit to any kind of sensor processing or control paradigm to be able to cooperate with another team mate, if both are programmed to interact in the same problem domain?

For cooperation between robots, the sharing of information about the environment is initially sufficient for successful cooperation. If all robots share both the same belief about their environment, as well as the same set of goals, similar conclusions should be drawn. This is known as the Intentional Stance [1], and has proven to be a successful way of coordinating behaviour in RoboCup scenarios [2,3]. Therefore, a central prerequisite for successful team cooperation is the unification of the beliefs about the world of the different agents. The limitations of the individual sensors usually provide each robot with quite limited information about the state of its environment. So it is unlikely that the beliefs derived solely from the robots’ own sensors are automatically sufficiently similar to coordinate behaviour in a shared environment. Sharing of information might solve this problem. This becomes especially obvious if the different autonomous mobile platforms robots are equipped with different sensor suites that each provide their own unique perception of the environment. Compare for instance laser range finders, which provide precise depth information, with colour cameras, which provide more certainty about object identity.

This paper presents the early stage of an approach of robotic labs from three different research groups, of the universities of Ulm, Munich and Graz, to be able to play interchangeably with their different robot platforms in a mixed team. The main contributions of this paper are along two research directions. First, the design and implementation of a communication framework for sharing information within a team of extremely heterogeneous autonomous robots is presented. It is hardware and software independent, and can extend existing software architectures in a transparent way. Major code rewrites are not necessary to use this framework. Second, we specify an expressive shared belief state that our robots can communicate using this framework, to complement own beliefs, and to coordinate behaviour.

The remainder of this paper is organised as follows. Section 2 presents the three robotic teams. Section 3 shows the design of belief exchange between these teams. The implementation of the communication framework is presented in section 4. Related work is discussed in section 5, and we present future work and conclude with section 6.

## 2 An Overview of the Three Teams

In this section we will describe the differences between the three teams, emphasising those that are relevant to the sharing of a belief state. One of the most

important differences lies in the sensors used. On the level of belief state representation, we have found that all teams use different methods of modelling uncertainty and inaccuracy. This is mostly caused by the different ways in which the sensors capture the world. Also, some teams use an egocentric frame of reference, and others allocentric. Furthermore, fusing information from other players or the coach into the belief states of individual robots is treated in different ways.

## 2.1 Ulm: Ulm Sparrows

The Ulm Sparrows [4] are custom built robots. Their sensor suites consist of infrared based near range finders and a directed camera. The available actuators are a differential drive, a pneumatic kicking device and a pan unit to rotate the camera horizontally ( $180^\circ$ ). Currently, the Ulm Sparrows team is actually a mixed team in itself as it switches to a new generation of the robot platform. Its sensor and actor suites are similar, but due to the technical progress, the properties of the devices vary significantly between the two platforms: The new robots are three times as fast as their predecessors, they have four times the camera resolution, etcetera. Nevertheless, they use the same code base.

Each robot acts upon an egocentric belief state, using the camera as its main sensor. The belief state does not only model the inaccuracy over the positions of detected objects, but also the uncertainty of the observations.

Currently, fusion only takes place for observations of the ball. The robots derive the position of the ball from the shared beliefs and compute the role assignment based on the exchanged allocentric dynamic poses of the other robots. The fusion of the different ball position hypotheses is computed by averaging the different observations if they differ less than a certain threshold.

## 2.2 München: Agilo RoboCuppers

The AGILO RoboCuppers [5,6] are customised Pioneer I robots. Their only sensor is a fixed forward facing colour CCD camera with an opening angle of  $90^\circ$ . The robots use differential drive, and have a spring-based kicker.

The vision based localisation provides accurate self-localisation. For this reason the AGILO RoboCuppers use an allocentric belief state, in which each robot represents itself with a  $(x, y, \phi)$  pose in a global frame of reference. Our localisation module also computes a covariance matrix for each pose, that represents its inaccuracy. This is discussed more elaborately in section 3. There is no representation of the uncertainty of what an observation might be. Positions of static objects (goal-posts, corner-flags) depend upon the allocentric self-localisation, and dynamic objects (ball, teammates, opponents) can be distinguished well enough by their colours to allow for this simplification.

There are three methods by which the AGILO RoboCuppers fuse their observations to reach a more consistent and complete belief state [7]. Robots communicate the estimates of their own pose and ball position to each other. Self localisation estimates of other robots are always fused into the belief state, and ball positions only if another estimate is more accurate, or the robot itself has

not perceived a ball at all. Ball and opponent positions are communicated to a central coach computer which uses Multiple Hypothesis Tracking [8] to fuse the observations, and so achieve a global belief state, which is communicated back to the robots.

### 2.3 Graz: Mostly Harmless

The robots of Mostly Harmless [9] are custom-built robots with a modular design of hardware and software, motivated by the intention of using the robots in different domains, i.e. not only playing soccer, but also executing delivery tasks in office environments.

The robots do not rely solely on their omni-directional camera, but are also equipped with a forward facing laser range finder (LRF) with an opening angle of  $180^\circ$ , and ultra-sonic sensors for  $360^\circ$  distance measurements.

Objects recognised in LRF data, are not classified with such ease as objects found in colour images. By matching both results, we get a more accurate position for objects classified by their colour. To cope with unclassified objects, or more precisely, with objects that are not classified uniquely, the internal representation of the object class is a set of significance values, where the relation to each of the known classes is expressed by a probability estimate. The internal representation of object positions in the numerical model is based on an egocentric coordinate system, uncertainty in position information is expressed by an uncertainty estimate covariance. An allocentric view of the robots pose  $(x, y, \phi)$  and observed object positions is defined and used, whenever the robots operate in a known environment and self-localisation is possible, thus providing the robots with a common frame of reference. In these environments a robot is able to use position information from its teammates.

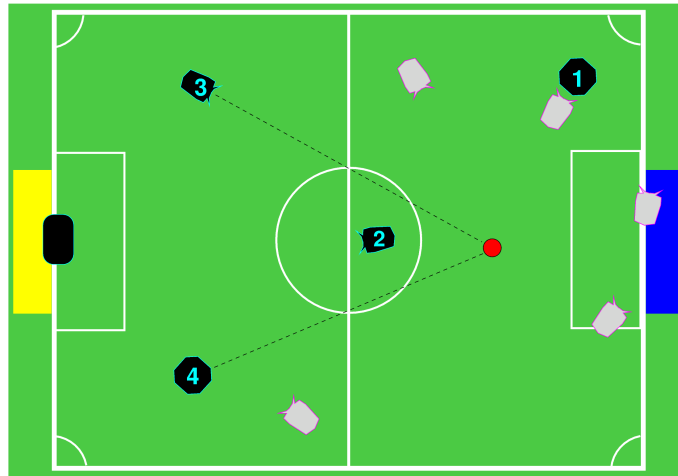
Sensor fusion is done locally, all information available from built-in sensors is integrated into the local belief state by Multiple Hypothesis Tracking methods [8] and is broadcasted in certain intervals to teammates and the coach computer.

Cooperative behaviour is coordinated by locker-room agreements, that is, by deriving the active role from the actual belief state. In special cases, additional symbolic information is exchanged within the team to aid decisions, e.g. in ambiguous situations [10].

## 3 Design of Belief Exchange

The scenario depicted in figure 3 shows how belief exchange can be useful. Player 1 has just shot at the goal, but the keeper has deflected the ball, which is now lying behind player 2. This player has a good chance at scoring, but cannot see the ball because it is outside of the field of view of its camera. Fortunately, two defenders observe the ball. One has a laser range finder, and can determine the location of the ball accurately. However, since colour information is lacking, it is not that certain about what the object is. Fortunately, the second defender has a colour camera, with which it recognises the ball, even if the localisation of it

is not very accurate. Player 2 receives this information from the two defenders, and fuses it to derive there is a ball behind it. All players compute that player 2 is closest to the ball, and player 2, determined by locker-room agreement, will turn and try to score a goal.



**Figure1.** A RoboCup Scenario

This scenario shows that exchanging information allows individual robots with a limited field of view to acquire information about hidden parts of the state, and to generate a more consistent and certain belief about the world by fusing observations from different robots, possibly with different sensors.

To share beliefs, the teams must agree upon structures that encapsulate this information. In this section we will discuss the design of these structures. The three main elements are a time-stamp, the probabilistic dynamic pose of the robot itself, and a list of observed objects. The main design principle is that we want an expressive belief exchange structure, but not simply a superset of all the information that the three teams have used so far.

*Time-stamped message-based communication.* The beliefs of the robots are exchanged within the team by a message-based protocol. Messages that are corrupted by packet loss in the wireless network do not influence each other, and can safely be ignored. As a basic requirement for sharing information in a highly dynamic environment each message is accurately time-stamped. This allows for interpolation of the beliefs to minimise the estimation error between messages.

*The own dynamic pose.* Many sensor data processing algorithms assume that the probability of measuring a certain quantity is distributed according to a Gaussian normal distribution. We also use this concept, and represent the robot's

pose by a three-dimensional vector and covariance matrix. The vector represents the mean of the Gaussian distribution, with  $(0, 0, 0)$  being the centre of the field, facing the opponent goal. The covariance matrix holds the associated uncertainty as a three-dimensional Gaussian distribution around this mean. How these values can be computed is discussed more elaborately in [7]. This probabilistic representation allows robots to communicate not only where they believe they are, but also how certain they are about it. Larger values in the covariance matrix mean that the robot is less certain about where it is.

Apart from its pose, the robot also communicates its velocity as a tuple  $(v_x, v_y, \omega)$ . There are no uncertainty measures for the speed. We call the combination of pose, covariance matrix and speed a probabilistic dynamic pose.

*Observed objects.* Apart from their own pose, the shared belief state also contains a representation of all the observations of objects the robots have made. Each object is represented by the same dynamic pose and covariance matrix, which have been discussed in the previous section. Although this representation is somewhat redundant (goal-posts will never have a velocity), it is very general. The observed objects are projected in an egocentric frame of reference, in which the observer always has pose  $(0, 0, 0)$ .

Each observation must be mapped to one of the objects that can be expected to be encountered in the robot’s environment. On a soccer field these objects are the ball, the teammates, the opponents, the corner-flags and the goal-posts. Any other objects (referee, assistants, flashers) are deemed obstacles. Since observations can usually not be mapped to one object with perfect certainty, each observation is accompanied by a list of all possible objects, and the estimated probability that the observation was this object. Therefore, a very general list of observations would be the matrix of the observations and object assignment probabilities, with each column summing up to one. See table 3 for an example.

Observation	p(Ball)	p(Teammate1)	p(Teammate2)	...	p(CornerFlag1)	$\sum p$
1	0.89	0.01	0.01	...	0.00	1.0
2	0.03	0.24	0.27	...	0.00	1.0
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
N	0.01	0.00	0.00	...	0.51	1.0

**Table1.** The observation-object assignment matrix

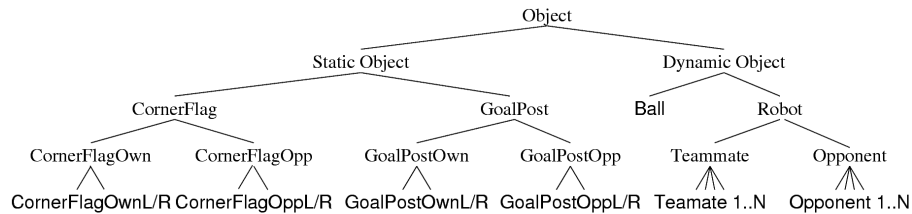
### 3.1 Design issues

*Allocentric vs. Egocentric.* The most obvious way to fuse observations of different robots is by the use of a shared frame of reference. Therefore the robots use the allocentric frame of reference defined in the previous section, to communicate their poses.

However, using an allocentric frame of reference to coordinate behaviour requires knowing where you are with sufficient accuracy, something that cannot always be guaranteed in the F2000 league. An egocentric frame of reference, in which objects are localised relatively to the robot, can be sufficient for the execution of many tasks. Knowing where the ball and opponent goal are can be enough to score a goal. Passing is another example where the accuracy of the relative robot poses is more important than global localisation of these robots. For these reasons we have decided to use an egocentric frame of reference for the observations.

*Inaccuracy and uncertainty.* The differences between the sensor systems of the teams allow them to derive different information about the objects they observe. The laser range finder of the Graz team provides accurate information about object locations, but the uncertainty as to what kind of object it is is not as certain. The other colour-camera based teams are much more certain about the type of object (*orange* ball, *yellow* goal), but cannot locate the objects as precise. We have chosen to represent both aspects in the shared belief state, to allow all teams to express their different types of uncertainty. Appropriate fusion of this distributed multi-modal perception increases the accuracy and certainty of each robot's local belief state.

*Simplification of the observation assignment matrix.* In the communicated beliefs, the matrix is actually sent as a set of object-probability pairs for each observation. First, the matrix is usually quite sparse, as most classes of objects are clearly discernible from each other. For instance, robots are usually easy to distinguish from goal-posts, but less easily from each other. A second reason is that probabilities below a certain threshold will probably not influence the decisions of a teammate, and need therefore not be communicated. The sum  $s$  of the probabilities in each set is less or equal than one, with  $1 - s$  representing the robot's ignorance about the observation.



**Figure2.** Tree of Object Classes

To make the list of observation-object more compact, superclasses of the individual objects have been defined. Often it is the case that a robot recognises a robot as being an opponent, but not which number it has. Instead of sending

that it can be Opponent 1,2,3 or 4, each with the same chance, the robot has the option of sending the superclass Opponent. If really nothing useful can be said about what kind of object the observation is, the robot could even send that it is of class Object (in this case the assignment probability is always 1.0). The full tree of objects and superclasses can be seen in figure 3.1.

## 4 Implementation of the communication framework

The team communication uses a message-based, type safe high-level communications protocol that is transferred by IP-multicast. A type safe high-level protocol is necessary to keep the communicated data easily accessible and prevent subtle programming errors that are hard to trace through different teams. It also facilitates adjustments in the communicated data structures, as needed during development. As the communication in a team of autonomous mobile robots has to use some kind of wireless LAN, that is notoriously unstable especially in RoboCup tournaments, a connection-less message based protocol is mandatory. This way, network breakdowns and latencies do not block the sending robot. And as messages do not depend on each other, messages from other teammates can be integrated into the shared belief state as soon as the network becomes available again. To save bandwidth, IP-multicast is used, since this way every message has only to be broadcasted once, instead of  $n$  times for  $n$  clients.

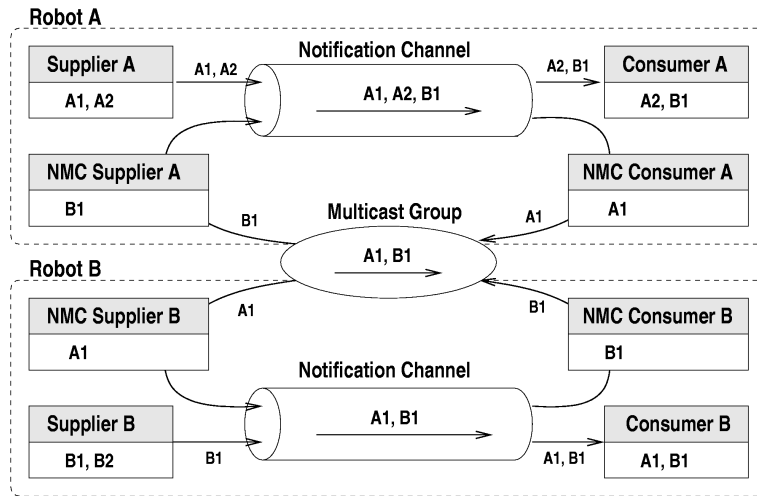
The implementation uses the notify multicast module (NMC) of the Middleware for Robots (MIRO) framework [11]. MIRO is a CORBA based middleware architecture for autonomous mobile robots. It provides generalised sensor and actuator interfaces for various robot platforms as well as higher level frameworks for robotic applications such as behaviour based control and video image processing. Additionally to the method-call oriented interfaces, MIRO also uses the event driven, message-based communications paradigm utilising the CORBA Notification Service [12]. This standardised specification of a real-time event channel is part of various CORBA implementations [13,14]. The Notification Service specifies a variant of the publisher/subscriber architecture. Publishers (suppliers in their terminology) offer events. The so-called consumers subscribe to those events. They then receive the events supplied by the publisher through the event channel. The NMC module customises this service for team communication. The data exchanged is specified in the CORBA interface definition language (IDL). Standardised mappings from IDL to most modern programming languages (C, C++, Java) exist.

CORBA uses a connection oriented (usually TCP/IP based) communication layer by default, the NMC module therefore plugs into the Notification Service architecture and exchanges events between the robots of a team transparently, using IP-multicast. For this purpose a service federation quite similar to the one described in [15] is used. An event channel instance is run locally on each robot. A “NMC event consumer” subscribes for all events that are offered only locally but subscribed by other team mates and sends them to the multicast group. A “NMC event supplier” in turn listens to all events published via IP-multicast



and pushes those into the local event channel, that are subscribed but not offered locally. To keep track of the offered and subscribed message types, NMC utilises two fields of the standard event message format: The domain name and the type name. By its convention, the domain name of a message contains the name of the robot producing the event. The type name describes the payload of the event. As these fields are also part of the native offer/subscription management and filtering protocol of the notification service, robots can easily determine whether events they offer are currently subscribed to in the team, and skip their production entirely if there are no subscribers.

Figure 4 illustrates a sample configuration of the notification channel setup. Two robots ( $A, B$ ) produce two types of events ( $1, 2$ ), the resulting events are  $\{A1, A2, B1, B2\}$ . The events in the supplier and consumer boxes denote the offered and subscribed events. The events labelling the arrows denote the actual flow of events. Note that suppliers and consumers can offer or subscribe for multiple events.



**Figure3.** A Federated Notification Channel Setup

Communicating the IDL-specified belief state discussed in section 3 at 10Hz with all teammates uses, on average, less than 10% of the available bandwidth of a standard 802.11b WLAN (11 MBit/s). This should be available, even on heavily loaded networks, such as those in RoboCup tournaments.

We would like to emphasise that even though our main goal is to communicate belief states, this framework is not limited to this information only. In principle it is possible to communicate any type of information, for instance role assignments, utilities, coaching advice.

## 5 Related Work

Team wide information sharing is a well-known concept in RoboCup. CS Freiburg used a global world model fused on a central coach computer by the use of Kalman and particle filters [16]. It was then sent back to the team mates. The system was based on a very accurate LSR-based self localisation. High accuracy was achieved by using the walls that enclosed the field up to 2001 as landmarks. An early attempt of information sharing in the legged-robot league is described in [17]. However, this approach suffered from the severe network latencies of the communication device available on this platform.

The idea of cross team cooperation has some tradition within the RoboCup leagues. In the simulation league, the source code of many teams was published on the Internet allowing new participants to base their new team on previous participants of simulation league tournaments.

One of the most successful mixed teams in RoboCup has been the GermanTeam, which participates in the legged-league [18]. The GermanTeam is a cooperation of five universities participating with one team and one code repository. The exchange and integration of software is enabled by a standardised hardware platform, as well as a modular software design. The challenge we face is to integrate different hardware systems and software architectures, for which integration has never been a primary goal. A bottom-up design, such as the GermanTeam has, would require complete rewrites of all systems, so instead we have chosen a software package that extends each individual software architecture.

The most similar mixed team cooperation effort was done by the Azzurra Robot Team. They built a middle size league national team from various Italian universities. They also used a (proprietary) publisher/subscriber communication protocol, utilising UDP. However, their focus was on explicit role assignment and coordination strategies among the field players [19]. Unfortunately the Italian national team was dissolved after the RoboCup tournaments in 2000.

We share the concept of distinguishing between that which is communicated and how it is communicated with languages such as the Knowledge Interchange Format [20]. However, this language was developed for exchange of information formalised in different knowledge representation languages between knowledge-based systems, and not primarily for distributed robotic agents, where the emphasis is more on state estimation and modelling dynamic systems.

## 6 Conclusion and Future Work

Currently each team has its own fusion methods. The data exchanged is designed in such a way that each team should be able to use it for their fusion, as well as supply the data required by the other teams. Sharing the same belief state will allow us to compare the different methods better, as the belief state abstracts from the sensory differences of the teams. In the long run, a unified fusion can be expected, as all team will try to get the best out of the shared information.

So far, most coordination in RoboCup is based on explicit communication of role assignments or their utilities [21,19]. One of the goals of this cooperation is

to show that communicating only a belief state can be sufficient to coordinate behaviour and assign roles dynamically. We hope to use models of teammates' dynamics and navigation routines to predict who will be able to go to the ball quickest. Since the robots share their belief state these predictions should be similar. This approach is successfully demonstrated in [6], in which only one robot is going for the ball more than 98% of the time, without explicit communication about role assignment.

In this paper we have presented a CORBA based communication framework for sharing information within a team of extremely heterogeneous autonomous robots is presented. Our three teams have extended their existing software architectures with this framework, enabling transparent communication between these teams. We have also introduced the shared belief state we communicate using this framework, and discussed the underlying design principles and motivations.

The design of the mixed team communication for sharing beliefs was started on the altruistic ground of scientific experimentation. Nevertheless it might turn into a necessity, as it seems that not all three teams can afford to attend this years Robot Soccer World Cup in Lisbon as an independent team.

## Acknowledgements

The work described in this paper was partially funded by the Deutsche Forschungsgemeinschaft (German Research Foundation) in the SPP-1125, "Cooperating Teams of Mobile Robots in Dynamic Environments".

MIRO is freely available as open-source at  
[smart.informatik.uni-ulm.de/Miro/](http://smart.informatik.uni-ulm.de/Miro/)

The specification of the shared belief state is available as open-source at  
[sharedbelief.sourceforge.net/](http://sharedbelief.sourceforge.net/)

## References

1. Dennett, D.C.: *Brainstorms*. Harvester Press (1981)
2. Tews, A., Wyeth, G.: Thinking as one: Coordination of multiple mobile robots by shared representations. In: International Conference on Robotics and Systems (IROS). (2000) 1391–1396
3. Buck, S., Beetz, M., Schmitt, T.: Reliable multi robot coordination using minimal communication and neural prediction. In Beetz, M., Hertzberg, J., Ghallab, M., , Pollack, M., eds.: *Advances in Plan-based Control of Autonomous Robots*. Lecture Notes in Artificial Intelligence, Springer (2002)
4. Kraetzschmar, G., Mayer, G., Utz, H., Baer, P., Clauss, M., Kaufmann, U., Lauer, M., Natterer, S., Przewoznik, S., Reichle, R., Reisser, A., Roth, A., Schmidt, M., Sitter, C., Sterk, F., Palm, G.: The Ulm Sparrows 2003. In: *Proc. of RoboCup-2003 Symposium* (to appear). Lecture Notes in Artificial Intelligence, Berlin, Heidelberg, Germany, Springer-Verlag (2003)

5. Beetz, M., Gedikli, S., Hanek, R., Schmitt, T., Stulp, F.: AGILO RoboCuppers 2003: Computational principles and research directions (2003)
6. Beetz, M., Schmitt, T., Hanek, R., Buck, S., Stulp, F., Schröter, D., Radig, B.: The AGILO 2001 robot soccer team: Experience-based learning and probabilistic reasoning in autonomous robot control. to appear in *Autonomous Robots*, special issue on Analysis and Experiments in Distributed Multi-Robot Systems (2004)
7. Schmitt, T., Hanek, R., Beetz, M., Buck, S., Radig, B.: Cooperative probabilistic state estimation for vision-based autonomous mobile robots. *IEEE Trans. on Robotics and Automation* **18** (2002) 670–684
8. Reid, D.B.: An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control* **24** (1979) 843–854
9. Steinbauer, G., Faschinger, M., Fraser, G., Mühlenfeld, A., Richter, S., Wöber, G., Wolf, J.: Mostly Harmless Team Description. Submitted as Team Description Paper for Robocup 2003 (2003)
10. Fraser, G., Wotawa, F.: Cooperative Planning and Plan Execution in Partially Observable Dynamic Domains. Submitted for publication (2004)
11. Utz, H., Sablatnög, S., Enderle, S., Kraetzschmar, G.K.: Miro – middleware for mobile robot applications. *IEEE Transactions on Robotics and Automation*, Special Issue on Object-Oriented Distributed Control Architectures **18** (2002) 493–497
12. Gore, P., Cytron, R., Schmidt, D.C., O’Ryan, C.: Designing and optimizing a scalable CORBA notification service. In: *LCTES/OM*. (2001) 196–204
13. Schmidt, D.C., Gokhale, A., Harrison, T., Parulkar, G.: A high-performance endsystem architecture for real-time CORBA. *IEEE Comm. Magazine* **14** (1997)
14. Brose, G.: Jacorb: Implementation and design of a java orb. In: *Proceedings of DAIS ’97*, Cottbus, Germany, Chapman & Hall (1997)
15. Harrison, T.H., Levine, D.L., Schmidt, D.C.: The design and performance of a real-time CORBA event service. In: *Proceedings of OOPSLA ’97*, Atlanta, ACM (1997)
16. Dietel, M., Gutmann, S., Nebel, B.: Cs freiburg: Global view by cooperative sensing. In: *Proc. of RoboCup-2001 Symposium*. Volume 2019 of *Lecture Notes in Artificial Intelligence*., Berlin, Heidelberg, Germany, Springer-Verlag (2001) 86–95
17. Roth, M., Vail, D., Veloso, M.: A world model for multi-robot teams with communication (2002)
18. Röfer, T.: An architecture for a national RoboCup team. In Kaminka, G., Lima, P.U., Rojas, R., eds.: *Proc. of RoboCup International Symposium 2002*. (2002)
19. Castelpiitra, C., Iochhi, L., Nardi, D., Piaggio, M., Scalzo, A., Sgorbissa, A.: Communication and coordination among heterogeneous mid-size players: ART99. In: *Proc. of RoboCup-2001 Symposium*. Volume 2019 of *Lecture Notes in Artificial Intelligence*., Berlin, Heidelberg, Germany, Springer-Verlag (2001) 86–95
20. Patil, R., Fikes, R.F., Patel-Schneider, P.F., McKay, D., Finin, T., Gruber, T., Neches, R.: The DARPA knowledge sharing effort: Progress report. In Nebel, B., Rich, C., Swartout, W., eds.: *KR’92. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference*. Morgan Kaufmann, San Mateo, California (1992) 777–788
21. Spaan, M.T.J., Groen, F.C.A.: Team coordination among robotic soccer players. In Kaminka, G., Lima, P.U., Rojas, R., eds.: *Proceedings of RoboCup International Symposium 2002*. (2002)