# Improving Self localization and Object Localization by a Team of Robots Using Sensor Fusion

Abdolkarim Pahliani

Institute for Systems and Robotics
Instituto Superior Técnico
1049-001 Lisboa
apahliani@isr.ist.utl.pt

Pedro Lima

Institute for Systems and Robotics
Instituto Superior Técnico
1049-001 Lisboa
pal@isr.ist.utl.pt

**Abstract**

*In this paper we propose an approach to handle the cooperative localization of an object by a team of robots. Our technique uses propagation of beliefs to combine information provided by a team of robots so as to support a teammate in critical situations when it can not see an object, is lost or equipped with low quality sensors and need cleaner data. The performance of our approach has been tested in realistic simulations.*

## 1 Introduction

Currently mobile robots are engaged in a lot of human like activities where they play a vital role when the task is harmful, too hard or requires more accuracy. In many situations robots need to get assistance from other agents to carry out their tasks. Instead of performing a complex task using only one comprehensive robot, an alternative is to employ a group of robots with distributed work and collective responsibility. One fundamental and challenging problem for mobile robots is localization, *i.e.* the ability of a robot to determine its location precisely. The problem of localization can be divided in three categories: global localization, relative localization and robot kidnapping. Global localization is the more difficult case comparing to relative localization. It deals with situations where the robot has no initial information about its location. The robot attempts to estimate its location with respect to an external global frame. Most approaches fail to localize the robot from scratch and in some cases the accuracy is not acceptable. Compare to relative localization methods, global localization solutions provide less accurate answers, are computationally expensive and memory demanding. Local tracking or relative localization occurs when the robot knows its current location with respect to other object and continuously keeps updating its relative location. The problem of robot kidnapping robot is to estimate the new position of a displaced robot. This is similar to global localization, but become harder due to the fact that the robot has to change its current belief based on its location, without being aware of a sudden displacement to a new location. It is the most difficult part, since robot is not aware of that. First it should catch its displacement and then find its new position.

Markov Localization is a probabilistic framework which is able to cope with the problem of global localization and is adaptable to cooperative localization[1][2][3][4][5][6][7][8]. Actually most of work in localization has been focused on single robot cases while a few in the field of cooperative localization exist. In Markov Localization a pdf (probability density function) is maintained all over the possible robot locations. No assumption about the form of the pdf is made, contrary to other methods such as Kalman filter based in which Gaussian pdf is assumed. Fox *et al*[1] proposed a Multi-Robot localization algorithm based on Markov Localization algorithm for a single robot. It has some drawbacks that are explained in Section 3.3. To solve those drawbacks, we address the problem of cooperative self localization and object localization based on a Bayesian propagation of beliefs for a team of robots which can be seen as an extension of Markov Localization. Each robot determines its own estimate of the object position, while the team is able to handle situations where one of the teammate can not observe the object or a better estimation of object position is considered through belief communication.

In section 2 we review related work. In Section 3 we briefly explain the Markov Localization approach and its extension for a team of robots introduced in [1]. Then in Section 4, we explain how to adapt this approach for object localization by a team of robots . In section 5 we present simulations that validate our method. In the experiment we are aimed to run a simulation to test the performance of our approach. In Section 6 conclusions are drawn.

**for** each location $l$ **do**

$$Bel_n(l) \leftarrow P(L_n^{(0)} = l)$$

**end for**
**forever do**
  **if** the robot receives new sensory input $\mathbf{o}_n$ **do**
    **for each** location $l$ do

$$Bel_n(l) \leftarrow \alpha P(o_n|l) Bel_n(l)$$

      **end for**
  **end if**
  **if** the robot receives a new odometry reading $a_n$ **do**
    **for** each location $l$ **do**

$$Bel_n(l) \leftarrow \int P(l|a_n, l^{'}) Bel_n(l^{'})$$

    **end for**
  **end if**
**end forever**

Table 1: Markov Localization Algorithm

**if** $nth$ robot is detected by the $mth$ robot **do**
  **for each** location $l$ **do**
$Bel_n(l) \leftarrow\ Bel_n(l) int P(L_n = l|L_m = l', r_m) Bel_m(l^{'}) dl^{'}$
  **end for**
**end if**

Table 2: Markov Multi-Robot Localization Algorithm

## 2 Related works

This work is connected to two important area in mobile robotic: Multi-Robot Localization and Sensor Fusion. Most researches in the field of Robot Localization have been concentrated on the single localization of a single robot. On the other hand, using multi-agent systems because of their high performances and simplicity have become widely spread. Fox, Burgard and thrun in [1][2][3][4][5][6][7] [8] proposed an algorithm based on first order Markov assumptions for a single robot which is called Markov Localization. They extended the algorithm to adapt to different situation, *i.e.* dynamic environment, team of robots and etc. Roumeliotis and Bekey [9] used a centralized approach to fuse observation of a team. They used only a Kalman filter to optimally combine the observations of team of robots.

A group of researchers have been working on simultaneously map building and localization by a team of robot in an unknown environment [10][11].

Howard [12] used a combination of maximum likelihood estimation and distributed numerical optimization for cooperative localization. They considered members of a team of robots as a landmark to perform relative localization and reduce uncertainty. Howard *et al*[13] used Bayesian method for multi robot localization. In his approach each member of a team keep information about relative position to other teammate and use it as a particle filter. Ioannis*et al* examined the effect of size of a team of robot on cooperative localization. They applied particle filetr and used a kind of robot called as tracker to find out relative positioning[14]. In[15] Marcelino *et al* developed an approach to increase coverage field of view of robots in a team. A group of researchers adapted Markov Localization algorithm to a highly dynamic environment for a team of robot to self-localize a soccer player in the RoboCup Simulation League[16][17]. Pinheiro *et al*[18] adjusted Durrant-Whyte approach[19] for sensor fusion and introduced a new algorithm for representing, communicating and fusing distributed, noisy and partial observations of an object by multiple robots. Using this technique, team members were able to do more frequent cooperation between team members, so as to solve conflict situations and achieve a team consensus faster.
There are lots of work done in the area of sensor fusion. Durrant-whyte[19], considered a multi-sensor system as a team of decision makers. Each sensor is able to make a local decisions and also participate in team decision. Sensors use a dependency model to combine information if certain condition is met.

Some researchers applied machine learning technics or combination of machine learning algorithms to fuse and integrate sensory information. Carpenter *et al* used a neural networks to perform real-time localization and ranging using multiple sensors on a mobile robot. The robot learned how to fuse visual and sonar information to determine the distance to objects in its environment[20]. Pasika[21] and Joseph[22] applied Neural Network and a combination of Neural Network and Support Vector Machine to fuse the the information of sensors. The main idea was to build a virtual sensor at lower cost. Mahajan*et al*[23] developed a fuzzy logic inference system to integrate and fuse data from different types of sensors with different resolutions based on the sensors uncertainty. Shekhar*et al*[24]proposed a framework to integrate information of multiple sensors to determine position and orientation of an object.

## 3 Markov Localization

Markov Localization is one of the best and most effective approaches in global localization that has recently been

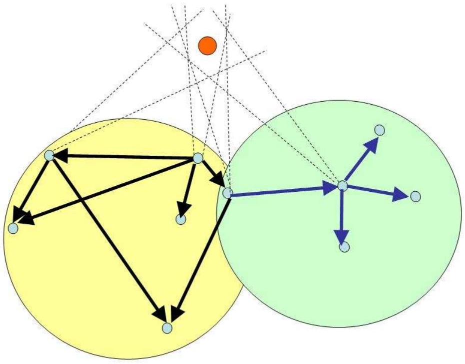

Figure 1: An example of a team of robots that is divided into two sub-teams. It also shows the propagation of information in two sub-teams for cooperative localization of an object. Two sub-teams can also exchange information via common member.

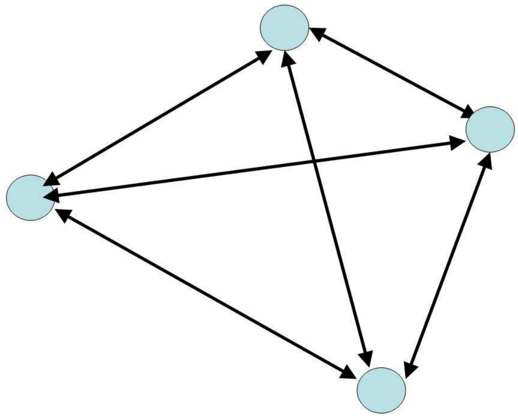

Figure 2: An example of exchanging information in a sub-team for improving self localization of robots.

successfully applied to a wide variety of localization problems. Contrary to other popular algorithms such as those based on Kalman filtering, it makes no assumption about uncertainty distributions. Instead of maintaining a single hypothesis concerning the best estimate of the robot position, this approach keeps a probability density over the space of every possible positions and orientations, which is called Belief. Initially, when there is no information available about the initial location, the pdf is a uniform distribution. In further steps, the robot combines sensory and odometry information and the pdf typically evolves to a multimodal pdf. Finally when the robot finds its exact location, the pdf is converted to an unimodal distribution. The drawback is that, Markov Localization is computationally expensive and requires large memory.

### 3.1 Markov Localization algorithm

The basic Markov Localization algorithm is shown in Table 1. We use the same notation as in [1]. In a team of $N$ robots, each robot collects data up to time $t$ in a vector $d_n^t$, with $1 \leq n \leq N$. $d_n^t$ includes data from odometry and environmental sensors such as vision ,laser and etc. $Bel_n^{(t)}(L_n^t = l) = P(L_n^t = l|d_n^t)$ denotes the belief of the $nth$ robot at time $t$ being at position $l$. The algorithm can be divided into two major steps:

1-If a robot receives new environmental information it updates the likelihood of all new possible places given that data.

2-If robot receives new odometry data it updates the likelihood of all new possible places.

### 3.2 Markov Localization-Multi Robot algorithm

Fox et al[1] extended Markov Localization to a collaborative mobile robot localization in which a team of $N$ robots uses teammate's information in order to improve uncertainty, when one robot "observer" discovers another robot "observed". In this case data from the observer passes to observed robot. The observation of the observed robot by the observer provides extra constraints for the observed robot to improve its belief. The proposed algorithm is presented in table 2.

### 3.3 Drawbacks of Markov Localization

One of major drawbacks of this algorithm is that it is computationally expensive and memory demanding. Besides that, the communication of high amounts of data in a realtime system is challenging. For example, in[1] the team of $N$ robots need to communicate and keep $N(N-1)$ believes while exchanging data only with one robot might be enough. Another disadvantage of the Markov Localization algorithm is that there is no strategy to fuse data. In some cases fusing a belief with another one is useful while the others do not improve the uncertainty. Also to apply the Markov Localization extension approaches to a team of robots [1], robots should be active agents, capable of observing other agents, send and receive data. However, when locating commonly observed objects, those are passive agents, not able to measure their travelled distance, or observe its location and unable to communicate with the observing agents.

## 4 Distributed Team Localization Algorithm

In this section we briefly explain our algorithm. The algorithm is designed to improve uncertainty of self localization of robots and also uncertainty of robots for localiza-

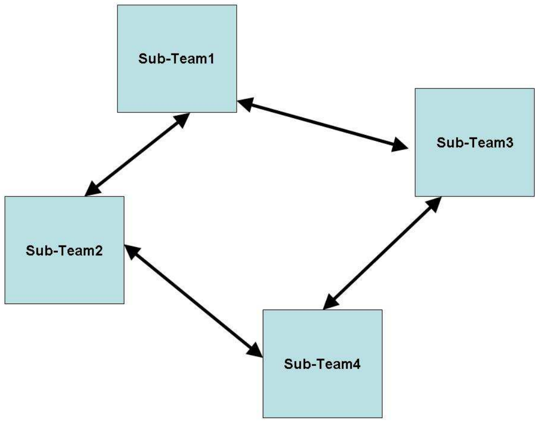


Figure 3: An example of exchanging information between sub-teams via common members.

tion of an object. First we define some concepts and then introduce the algorithm.

### 4.1 Uncertainty Measure

Entropy is a measure for uncertainty which is widely utilized. Although it was first introduced by Boltzmann in the context of Thermodynamics, it has been used in different areas such as Quantum Physics and Information Theory. Shannon used the concept of entropy in Information Theory, defined as:

$$H(L) = -\sum_i p_i \ln p_i \tag{1}$$

Where $p_i$ is probability that random variable $L$ takes on the value $l_i$. Entropy is a positive number: $H \succeq 0$. when *H* is zero we are fully certain. As the value of *H* increases we become less certain.
In probabilistic localization, we are confronted with uncertainty and use entropy as a measure of uncertainty. Suppose two robots *n* and *m* observe an object *Obj*. $Bel^{(t)}_{Obj}(L^t_{Obj} = l) = P(L^t_{Obj} = l|o^t_n)$ is defined as the belief of robot $n$ about $Obj$. A self entropy can be defined as:

$$H(Bel^{(t)}_{Obj}|o_n) = -\sum_l Bel^{(t)}_{Obj}(L^t_{Obj} = l) \\ lnBel^{(t)}_{Obj}(L^t_{Obj} = l)$$

Now first robot receives information about the location of the object from second robot. Robot $n$ to improve its uncertainty, takes into account the observation of robot $m$ and:

$$Bel^{(t)}_{Obj}(L^t_{Obj} = l) = P(L^t_{Obj} = l|o^t_n, o^t_m)$$

Based on this we can define cooperative entropy:

$$H(Bel^{(t)}_{Obj}|o_n, o_m) = -\sum_l Bel^{(t)}_{A}(L^t_{A} = l) \\ lnBel^{(t)}_{A}(L^t_{A} = l)$$

Entropy of a belief $Bel$ can be interpreted as amount of uncertainty contains in belief $Bel$. Uncertainty might be reduced by fusing different observations. We use entropy as a condition for fusing beliefs. We only fuse a belief with another one if entropy of combined Beliefs decreases.

### 4.2 Communication

Due to the restrictions in communications and also other physical limitations, each robot can not directly exchange information with all the other robots in a team. Moreover it would be computationally costly if each robot wants to use all other obsevations. For that reason we define an optimal radius $r$ in which fusing the information is likely to be useful. It is clear that the maximum value of $r$ is limited to the communication range.

### 4.3 Sub-team

Although a full cooperation among all the robots of a team can improve individual robot estimate of object locations, it is computationally expensive and sometimes technically impossible. For this reason we divide a team into sub-teams. In each sub-team, team members are in contact and exchange data. Besides that different sub-teams may have common members. Taking advantage of this, two sub-teams can exchange data through common member. In Fig. 1 it is shown how members of a team are divided in two sub-teams. Two sub-teams are able to communicate via a common member.

### 4.4 Fusing information in a sub-team

Assuming that the observations in each sub-team are independent, we can use the following equation to combine the information in a sub-team with $K$ robots:

$$P(L^{(t)}_n = l|y_1...y_k) = \prod_{i=1}^{K} P(L^{(t)}_n = l|y_i) \tag{2}$$

Where $y_i$ is the observation of the $i^{th}$ robot in a sub-team.

### 4.5 Flow of information in a Sub-team

In case of mutual localization the flow of information is bidirectional as it is shown in Fig. 2. For localization of an object, direction of flow of information depends on the relative position of the team member. Members that are closer send the date to the closest teammate and from them to all others. An example of flow of information in sub-teams is shown in Fig. 1. Sub-teams can exchange data bidirectionally and, in case of object localization, direction of propagation of information is from closest member to farthest as is shown in Fig. 3.

### 4.6 Fusing information between two members of a sub-team

Suppose robot $m$ observes $n$. Taking advantage of that, observed robot $n$ can improve its own localization via ([1]):

$$Bel_n^{(t)}(L_n^{(t)} = l) \leftarrow Bel_n^{(t)}(L_n^{(t)} = l) \sum P(L_n^{(t)} = l | L_m^{(t)} = l^{'}, r_{n,m}^{(t)} = d) Bel_m^{(t)}(L_m^{(t)} = l^{'}) \quad (3)$$

Where $r_{n,m}^{(t)}$ is distance between robots $n$ and $m$ at time $t$. In case of an object localization, where the object *Obj* is observed by robot *n* :

$$Bel_{Obj}^{(t)}(L_{Obj}^{(t)} = l) \leftarrow Bel_{Obj}^{(t)}(L_{Obj}^{(t)} = l) \sum P(L_{Obj}^{(t)} = l | L_n^{(t)} = l^{'}, r_{Obj,n}^{(t)} = d)(Bel_n^{(t)}(L_n^{(t)} = l^{'}) \quad (4)$$

and $Bel_n^{(t)}(L_n^{(t)} = l^{'})$ is calculated from (3)

### 4.7 Cooperative Localization

Localization of an object by a team of robots involves many different challenges. In many situation robots due to restrictions such as sensors range or physical obstructions, are not able to observe teammate's or other objects. Another reason to use collaborative localization is that it can enhance the accuracy of localization. To mange this, we used the Markov Localization algorithm with some changes to avoid its drawbacks. The algorithm is shown in table 3. The algorithm can be split in four steps:

1-Sub-team registration
2-Self localization
3-Object localization
4-Sub-teams communication

We divide a team into some sub-teams. We suppose all team members are heterogeneous. At first robots try to register in a sub-team. Sub-team size depends on the size of the field and communication limitations. It is also a function of number of observations that needs to be fused. After registration in a team, robots should determine their locations. They use Markov Localization algorithm for that. After that, robots send their beliefs about others to a sub-team blackboard, where all team members have access to teammate observations. Fusing observations is only useful if both observations agree. For checking agreement an entropy filter is used. Each robot combines information with others if entropy of fused belief drops down after fusing two observations. Suppose robot $m$ observes robot $n$. Robot $m$ sends the information about $n$ to blackboard. Robot $n$ uses equation (3) and builds $m$ belief about its own position. Robot $n$ only uses that information if the following condition is satisfied:

$$H(Bel_n^{(t)}(L_t | o_n, o_m)) \leq H(Bel_n^{(t)}(L_t | o_n)) \quad (5)$$

Then they use equation (2) to fuse the information.
Later robots make their belief about objects in the world. They use environmental sensors to detect objects and then use the Markov Localization algorithm. Again they send information to sub-team blackboard. Each robot uses equation (4) to find out teammate observations about object in field of view of sub-team. The fusing condition is the same as self localization but for an object *Obj* and they use equation (2) to combine the observation.
Sub-teams only need to communicate if there is some ambiguities such as an object not visible by one of the robots.

**Do** ***forever***
  **Do** for each robot *Team Registration*
    register robot in sub-teams
  **End Do**
  **Do** for each sub-team
    Self localize robots using Markov Localization Algorithm.
    Send information to sub-team blackboard.
    Build beliefs for each robot in sub-team, based on other teammate observations, using:
$Bel_n^{(t)}(L_n^{(t)} = l) \leftarrow Bel_n^{(t)}(L_n^{(t)} = l) \sum P(L_n^{(t)} = l | L_m^{(t)} = l^{'}, r_{n,m}^{(t)} = d) Bel_m^{(t)}(L_m^{(t)} = l^{'})$
    Fuse data ***IF***
$H(Bel_n^{(t)}(L_t | o_n, o_m)) \leq H(Bel_n^{(t)}(L_t | o_n))$
    using :
$P(L_n^{(t)} = l | y_1 ... y_k) = \prod_{i=1}^{K} P(L_n^{(t)} = l | y_i)$.
    Localize Objects in the field of view, using Markov Localization Algorithm.
    Send information to sub-team blackboard.
    Build beliefs about the objects in the field of view of sub-team, based on other teammate observations, using:
$Bel_{Obj}^{(t)}(L_{Obj}^{(t)} = l) \leftarrow Bel_{Obj}^{(t)}(L_{Obj}^{(t)} = l) \sum P(L_{Obj}^{(t)} = l | L_n^{(t)} = l^{'}, r_{Obj,n}^{(t)} = d)(Bel_n^{(t)}(L_n^{(t)} = l^{'})$.
    Fuse data considering ***IF***:
$H(Bel_{Obj}^{(t)}(L_t | o_n, o_m)) \leq H(Bel_{Obj}^{(t)}(L_t | o_n))$
    using :
$P(L_{Obj}^{(t)} = l | y_1 ... y_k) = \prod_{i=1}^{K} P(L_{Obj}^{(t)} = l | y_i)$
  **end Do**
Exchange data between sub-teams
***end forever*** :

Table 3: Cooperative Localization Algorithm

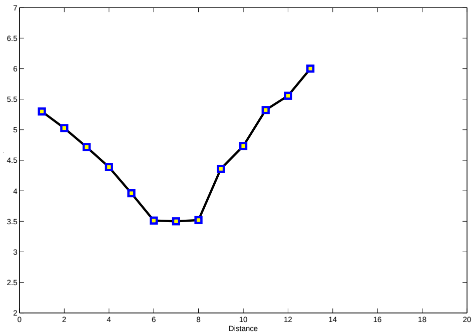


Figure 4: A sample entropy of uncertainty of vision sensor.

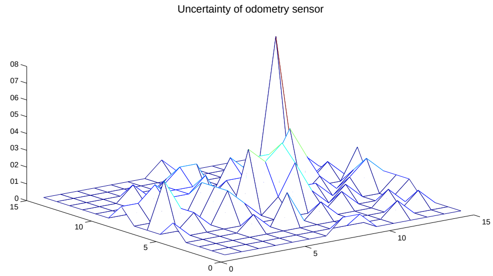


Figure 5: A sample of uncertainty of odometry sensore.

## 5 Simulation Result

We studied effect of cooperation of robots when a robot intends to improve the uncertainty of observation for self localization or common object localization and also if, for any reason it is not able to locate the common object. The world was divided into equal squares. We considred different resolutions ranged from $50 * 50$ to $200 * 200$. Ten robots equipped with two types of sensors: vision sensor and odometry sensors, were simulated. The model of vision is stochastic and only range of the sensors was fixed. A sample of uncertainty of vision sensor versus distance is shown in Fig. 4. The range of the sensor is restricted to cells 1 to 13 ahead of the robot sensor. Sensor delivers the best observation between 6 and 8, where the entropy is lowest. By increasing or decreasing distance, the uncertainty of observation increases. We also considered stochastic model for odometry sensor. A sample of sensor uncertainties is shown in Fig. 5. In the experiment, first we studied the effect of cooperation on mutual localization and then on the localization of a common object. At each step, robots in each sub-team first update their own belief using sensory information and then by considering odometry data. Then, robots exchanged the information with teammates. By that we mean every robot received belief of others about itself. In our example since robots were located in different positions all over the field and because of restrictions of vision sensors, only one could be observed by some of robots. Size of the sub-teams ranged from two to six depend on the situation. Before using those information, robot took advantage of entropy filter and chose effective beliefs. In other words, it checked whether the combination of that information with its own belief improved uncertainty. For the common object localization, after self localization each robot that observed that object, send the information to teammate and in this way they propagated the uncertainty in the sub-team and network. Using this strategy we not only reduced the entropy of localization of common object but also in the situations where robots were not able to observe the object, got an belief via neighbors. Beside that we needed less computational power, memory and communication than Markov method for Multi-Robot Localization. In Fig. 6 plot of entropy of original belief versus entropy of cooperative belief is shown . As the number of contributors increased we see a reduction in amount of entropy. In Fig. 7 we can see the the average entropy reduction due to number of cooperative robots. As the number of cooperative robots increased entropy of cooperative belief decreases.

## 6 Conclusion

The objective of this study was to investigate whether the cooperation of robots in a team could improve uncertainty of team individuals for self localization and also for localization of an object and when some robots losed other agents or objects. In the experiment we justified the suggested algorithm in a complex simulated environment. Sensory and odometry uncertainties have got unknown distributions which were generated stochastically. It was shown that cooperative localization were useful either when all team member were able or unable to see the target.

## References


[1] D. Foxy, W. Burgardz, H. Kruppayy, S. Thruny, A Probabilistic Approach to Collaborative Multi-Robot Localization, Autonomous Robots, 8(3), 2000.

[2] W. Burgard, D. Fox, D. Hennig, T. Schmidt, Estimating the absolute position of a mobile robot using position probability grids. In Proc. of the National Conference on Artificial Intelligence (AAAI),1996.

[3] W. Burgard, D. Fox, S. Thrun, Active mobile robot localization. In Proc. of the International Joint Conference on Artificial Intelligence (IJCAI),1997.

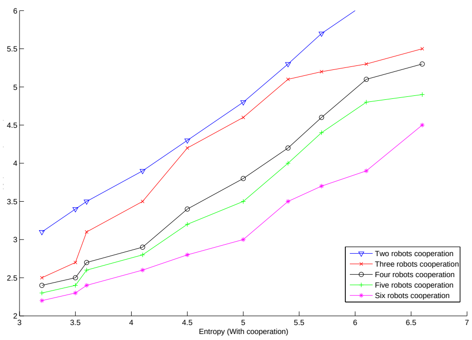


Figure 6: Entropy of multi-robots localization team versus single robot localization. Team size ranged from two to six.

[4] D. Fox,W. Burgard and S. Thrun, Active Markov localization for mobile robots. Robotics and Autonomous Systems, 25:195-207.

[5] D. Fox, W. Burgard, F. Dellaert, S. Thrun, Monte Carlo localization: Efficient position estimation for mobile robots. In Proc. of the National Conference on Artificial Intelligence (AAAI)1999.

[6] D. Fox, W. Burgard, S. Thrun, Markov localization for mobile robots in dynamic environments. Journal of Artificial Intelligence Research, 11,1999.

[7] D. Fox, W. Burgard, S. Thrun, A.B. Cremers, Position estimation for mobile robots in dynamic environments. In Proc. of the National Conference on Artificial Intelligence (AAAI) 1998.

[8] W. Burgard, A. Derr, D. Fox, A.B. Cremers, Integrating global position estimation and position tracking for mobile robots: The Dynamic Markov Localization approach. In Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 1998.

[9] S. I. Roumeliotis and G. A. Bekey. Collective localization, A distributed kalman filter approach to localization of groups of mobile robots. In IEEE Int. Conf. on Robotics and Automation, pg. 2958-2965, Apr. 2000.

[10] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots, International Journal of Robotics Research, 20(5):335-363, 2001.

[11] S.Thrun, D. Fox, and W. Burgard, A probabilistic approach to concurrent mapping and localisation for mobile robots. Machine Learning, 31(5):29-55, 1998. Joint issue with Autonomous Robots.

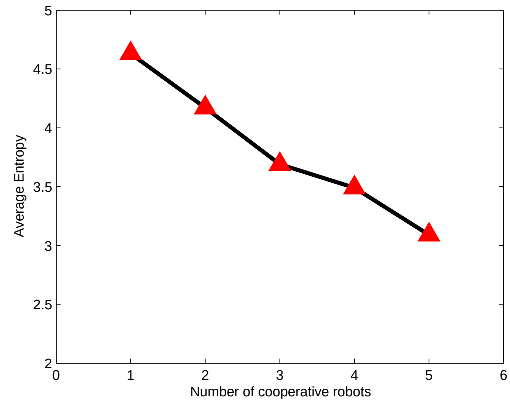


Figure 7: Average entropy of localization versus size of team.

[12] A. Howard, M. J Mataric, G. S. Sukhatme, Localization for Mobile Robot Teams: A Distributed MLE Approach. Experimental Robotics VIII, Bruno Siciliano and Paulo Dario (eds), Springer-Verlag, 2003, pp.146 - 155.

[13] A. Howard, M. Mataric, G. Sukhatme, Putting the 'i' in team: an ego-centric approach to cooperative localization,in Proceeding of the IEEE Int. Conference on Robotics and Automation, to appear, Taipei, Taiwan, May 2003.

[14] I. M. Rekleitis, G. Dudek, E. E. Milios, Multi-Robot Cooperative Localization: A Study of Trade-offs Between Efficiency and Accuracy. In Proceedings of Intelligent Robots and Systems, pages 2690-2695, Lausanne, Switzerland, September-October 2002.

[15] P. Marcelino, P. Nunes, P. Lima, M.I. Ribeiro, Improving object localization through sensor fusion applied to soccer robots, Proceedings do Encontro Cientfico do 3 Festival Nacional de Robtica, ROBOTICA 2003, Lisboa, 2003.

[16] C.C.F. Penedo, J.P.G.N. Pavo, P.Lima, M.I.Ribeiro,Localization in the Robocup Simulation League, Robotica, 4 trimestre, pp. 16-21, 2003.

[17] C. C. F. Penedo, J. P. G. N. Pavo, P. Lima, M.I.Ribeiro, Markov Localization in the RoboCup Simulation League,Proceedings do Encontro Cientfico do 3 Festival Nacional de Robtica, ROBOTICA 2003, Lisboa, 2003

[18] P. Pinheiro, P. Lima,Sensor Fusion for Cooperative Object Localization and World Modeling,Proc. 8th Conference on Intelligent Autonomous Systems, IAS-8, Amsterdam, The Netherlands, 2004.

[19] H.F. Durrant-Whyte, ”Sensor Models and Multisensor Integration,” Int. J. Robot. Res., vol. 7, no. 6, pp. 97-113, 1988.

[20] S. Martens, G.A. Carpenter, P. Gaudiano,Neural sensor fusion for spatial visualization on a mobile robot. Proceedings of the SPIE International Symposium on Intelligent Systems and Advanced Manufacturing (Boston, 11/98). Technical Report CAS/CNS TR-98-028, Boston, MA: Boston University.

[21] H.J.C. Pasika, ’Weural Network Sensor Fusion Engines for Remote Sensing”, PhD thesis, McMaster University, 1999.

[22] H. Joseph, Neural Network Sensor Fusion: Creation OF A Virtual Sensor For Cloud-Base Height Estimation PhD thesis, McMaster University, 1999.

[23] A. Mahajan, K. Wang, Multisensor Integration and Fusion Model that Uses a Fuzzy Inference, System and Probir,IEEE/ASME TRANSACTIONS ON MECHATRONICS, VOL. 6, NO. 2, JUNE 2001

[24] S. Shekhar, O. Khatib, M. Shimojo, Object localization with multiple sensors, Int. J. Robot. Res., vol. 7, no. 6, pp. 34-44, Dec. 1988.

[25] A. Stroupe, M.C. Martin, T. Balch, Merging gaussian distributions for object localization in multi-robot systems, Proc. of the ISER ’00 Seventh International Symposium on Experimental Robotics, Springer-Verlag.