8

Cooperative Global Localization in Multi-robot System

Luo Ronghua
School of Computer Science and Engineering, South China University of Technology
China

1. Introduction

With the development of robotic technology, the application areas of robots have been greatly widened. The robots may serve as assistants (Elena et al., 2004), rescuers (Robert & Arvin, 2003) and explorers (Schenker et al., 2003). In many cases a multi-robot system has to be used, since many tasks cannot be completed with a single robot and multi-robot system can finish the tasks much more efficiently. As in a single robot system, knowing their relative positions and their global positions in the environment are the preconditions for performing tasks and coordination. Since a robot can determine the location of another robot relative to its own when they see each other, both robots can refine their internal beliefs based on the other robot's estimate. In this way cooperative localization of multiple robots can greatly improve the localization precision and efficiency (Fox et al., 2000; Trawny et al., 2009). An area that has received some attentions in the single-robot case and very little attention in the multi-robot case is active localization (Fox et al., 1998). In active localization, the robot(s) may actively choose actions so as to aid in localization. Active localization has the potential to increase the speed and accuracy of localization further. In this chapter a mechanism of making robots coordinate their action actively during localization is proposed. In order to determine the exploration strategy, the ability to stably track multi-hypotheses of the robot's own position and his partners' positions is very important in active localization. However using traditional particle filters for localization tends to produce premature convergence, i.e. the hypothesis represented by particle filters converge to a small area of the state space with high likelihood too quickly. To overcome this problem a new version of particle filters termed co-evolution particle filters (CEPF) is proposed. Another problem of using particle filters in multi-robot localization system is the communication problem. When several robots want to share their estimates they have to transmit a large set of samples from one robot to another, so a great bandwidth is needed. To solve this problem the reduced set density estimator (RSDE) (Girolami & Chao, 2003) is used to estimate the density over robots' pose, so that only a small sub-set of the original samples should be transmitted between robots, which can reduce the communication data considerably.

2. Related works

2.1 Robot localization

Localization is a basic problem of mobile robot systems. Whenever the robots explore in an unknown environment or a known one, determining their own positions is of

great importance for them. Localization in unknown environments is called simultaneous localization and mapping (SLAM) (Dissanayake et al., 2001; Guivant & Nebot, 2001; Dasvison & Murray, 2002), whose purpose is to estimate the pose of robot so as to build a consistent map of the environment. The most popular method for SLAM is extended Kalman filter (EKF). Cooperative simultaneous localization and mapping (CSLAM) in multi-robot system has also been studied in recent years (Fenwick et al., 2002; Wu et al., 2009; Gil et al., 2010) . Compared with SLAM of single robot system, CSLAM can improve the mapping efficiency and localization precision based on the information sharing. Localization in known environment can be divided into two sub-problems: pose tracking and global localization. In pose tracking, the initial robot pose is known, and localization seeks to identify small, incremental errors in a robot's odometry. Global localization is a more challenging problem, in which the robot is required to estimate its pose by local and incomplete observed information under the condition of uncertain initial pose. Most recently, several approaches based on probabilistic theory have been proposed for global localization, including grid-based approaches (Burgard et al., 1996), topological approaches (Dayoub & Duckett, 2008; Kaelbling et al., 1996), particle filters (PF) based approaches (Dellaert et al., 1999) and multi-hypothesis tracking (Jensfelt & Kristensen, 2001). Since exploring actively can get more useful information for global localization, several methods for active localization in single robot system are proposed: a method using entropy to evaluate the utility of the robot's action was proposed (Hongjun & Shiyeyuki, 2002), Jensfelt proposed a active localization method with a topological map (Jensfelt & Kristensen, 2001) and a method based on Bayes network was proposed for sensor planning in localization (Kaelbling et al., 1996). In multi-robot system, the ability to exchange information is particularly attractive in global localization, in which the fusion of information can reduce the uncertainty in the estimated location. Particle filters are applied to localization in multi-robot system (Fox et al., 2000). But it is a passive localization method with no mechanism for the robots to actively determine their locations. In this paper we mainly discuss the active localization of multi-robots in a known environment based on a new version of particle filters.

2.2 Particle filter

Over the last years, particle filters have been applied successfully in many areas for state estimation (Doucet et al., 2000; Arulampalam et al., 2002). The key idea of particle filters is to represent the posterior density with a set of weighted samples. Particle filters include the following three steps:

- Re-sampling: resample N samples randomly from sample set S_{t-1} , according to the distribution defined by weights of samples w_{t-1};
- Importance Sampling: sample pose $x_t^{(j)}$ from $p(x_t|x_{t-1}^{(j)}, u_{t-1})$ for each of the N possible pose $x_{t-1}^{(j)}$, and evaluate the importance factor $w_t^{(j)} = p(y_t|x_t^{(j)})$.
- Summary: normalize the importance factors $w_t^{(j)} = w_t^{(j)} / \sum_{k=1}^{N} w_t^{(k)}$; and calculate the statistic property of sample set S_t to estimate the pose of the robot.

By representing probability densities with samples and using the sequential Monte Carlo importance sampling (Arulampalam et al., 2002), particle filters can represent non-linear and non-Gaussian models well and especially can focus the computational resources on regions with high likelihood, where things really matter. Particle filters have been successfully used in single robot localization (Dellaert et al., 1999) and have also been tried for multi-robot

localization (Fox et al., 2000). But using traditional particle filters for multi-robot localization has some shortcomings. Since samples are actually drawn from a proposal density, if the observation density moves into one of the tails of the proposal density, most of the samples' non-normalized importance factors will be small (Doucet et al., 2001). So a large sample size is needed to represent the true posterior density to ensure stable and precise localization. However in multi-robot cooperative localization, since all the samples should be transmitted from one robot to another, large sample size will not only lead to a heavy computational burden but also a heavy communication burden. Another problem of particle filters is that samples often converge to a single, high likelihood pose too quickly. In active localization this will not only lead to erroneous localization but also lead to inefficient exploration strategy. How to improve efficiency and to prevent premature convergence of particle filters are the key concerns of the researchers. To make the samples represent the posterior density better, Thrun et al. proposed mixture-MCL (Thrun et al., 2001), but it needs much additional computation in the sampling process. To improve the efficiency of MCL, a method adjusting sample size adaptively over time has been proposed (Fox, 2003), but it increases the probability of premature convergence. Although clustered particle filters are applied to solve premature convergence (Milstein et al., 2002), the method loses the advantage of focusing the computational resources on regions with high likelihood because it maintains the same sample size for all clusters.

3. Dynamic architecture of cooperative localization

We assume that robots can determine their relative positions when they see each other. And the relative positions between robots will also be updated according to their motion model for some time after one robot disappears from the sight of the other robots. Our active localization approach utilizes a dynamically evolving coordination architecture. At each point of time, the state of the system can be summarized by a graph structure where the nodes are individual robots and the edges represent the relationship between robots (see Fig.1). An isolated node represents that the robot doesn't know its relative positions to the other robots and cannot communicate with the other robots, in this case the robot tries to determine its global position by itself as in a single robot system. Another kind of relationship between robots is that they do not know their relative positions but they can communicate with each other, which is represented by dotted line in Fig.1 (for example robot 2 and robot 3). In this case there may be several hypotheses of their relative positions. In order to actively verify their relative positions, the two robots are arranged to meet each other at a rendezvous point. As is shown in the left picture of Fig.1, robot 2 will manage to meet robot 3 so as to determine its position using the information gotten by robot 3. If the robots fail to meet, the hypothesis will be rejected and they continue to select the other hypothesis to verify. A key sub-structure of our architecture is the connected group indicated by the shade areas in Fig.1. In the connected group, two robots that know their relative positions and can communicate with each other are connected with a solid line. The robots in a connected group can fuse their sensor information to improve the precision of localization. And each group determines one robot as a leader to be responsible for coordination of their exploration (for example robot 4 and robot 9). But if the poses of the robots in a connected group are merged to form a single state for coordination, it is infeasible for a small number of robots due to the too high dimensional state space. So a hierarchical structure is proposed in which each robot maintains its own belief function, i.e. the information of its position estimated by other robots will be fused by each robot itself. Only several summarized hypotheses of their global positions will be transmitted to their

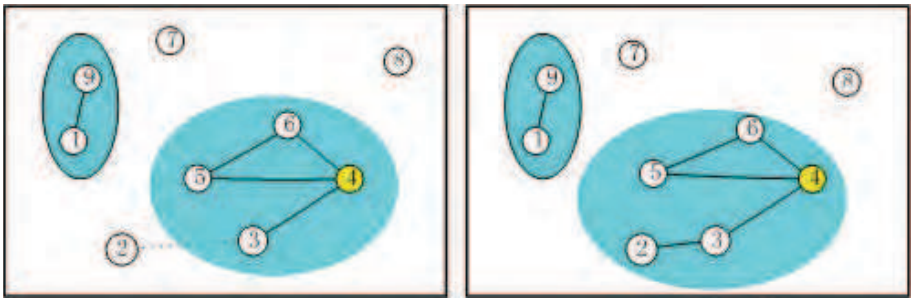


Fig. 1. The dynamic relationship between robots

leader, and the leader will estimate the most likely positions where the robots are located. Then the leader will choose an action for each robot to maximize utility-cost trade-off of the group. The active coordination structure of multi robots is shown in Fig.2. In order to reduce the communication data between robots, a density estimator is used before the data transmission.

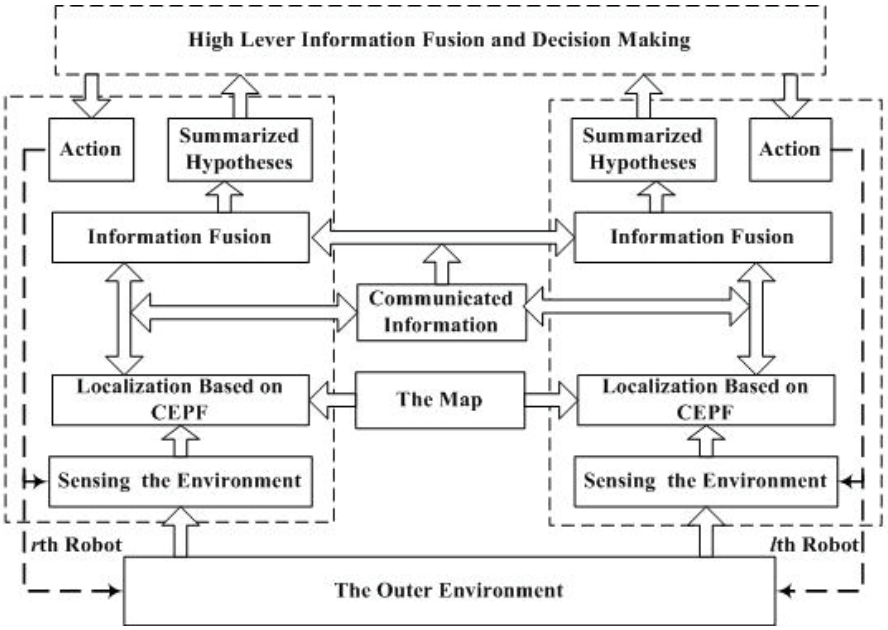


Fig. 2. Multi-robot active localization architecture

4. Cooperative localization based on co-evolution particle filters

In this section we discuss multi-robot localization based on a new version of particle filters called co-evolution particle filters (CEPF). In the first sub-section we present co-evolution particle filters; in the second sub-section, the algorithm of cooperative localization of multiple robots based on CEPF is described, in which an efficient communication mechanism is proposed to reduce the data that should be transmitted between robots.

4.1 Co-evolution particle filters

In order to satisfy the needs of the dynamic architecture of active localization and overcome the limitations of particle filters, samples are clustered into groups which are also called species. And a co-evolutionary model derived from the competition of ecological species is introduced to make the species evolve cooperatively, so the premature convergence can be prevented. And genetic operators are used for intra-species evolution to search for optimal samples in each species. So the samples can represent the desired posterior density better, and precise localization can be realized with a small sample size. The improved version of particle filters is termed co-evolution particle filters (CEPF).

1) Inter-Species Competition

The concept of co-evolution is derived from ecologic science. In ecology, much of the early theoretical work on the interaction between species started with the Lotka-Volterra model of competition (Shang & Cai, 1996). The model itself was a modification of the logistic model of the growth of a single population and represented the result of competition between species by the change of the population size of each species. It is simple and easy to use the model although it could not embody all the complex relations between species and it has been accepted by most of the ecologists. In this paper, Lotka-Volterra model is merged into particle filters to solve the premature problem of particle filters. Let us assume the samples of the r-th robot can be clustered into $\Omega_r^{(t)}$ species (clusters) at time t. Inspired by ecology, when competing with other species the population growth of species can be modeled using the Lotka-Volterra competition model. The Lotka-Volterra competition model for two species, which are denoted using species 1 and species 2, includes two equations of population growth, one for each of the two competing species.

$$\frac{dN_r^{(1)}}{dt} = \eta_r^{(1)} N_r^{(1)} \left(1 - \frac{N_r^{(1)} + \alpha_r^{(12)} N_r^{(2)}}{K_r^{(1)}}\right) \tag{1}$$

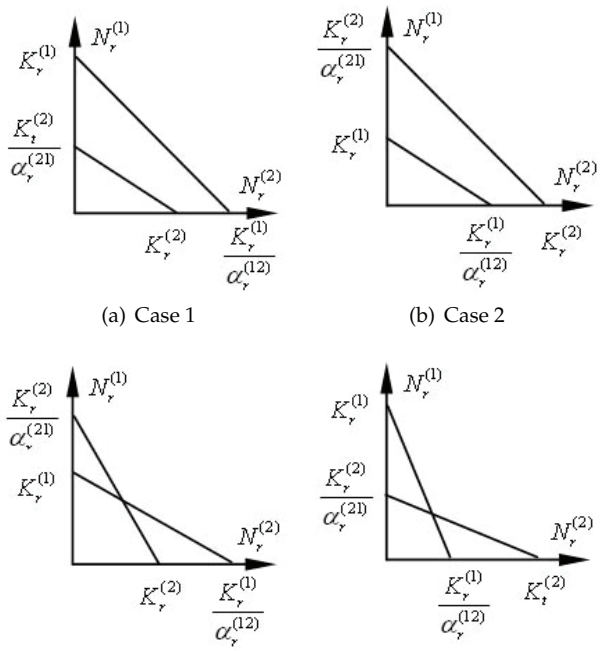
$$\frac{dN_r^{(2)}}{dt} = \eta_r^{(2)} N_r^{(2)} \left(1 - \frac{N_r^{(2)} + \alpha_r^{(21)} N_r^{(1)}}{K_r^{(2)}}\right) \tag{2}$$

Where $\eta_r^{(1)}$and $\eta_r^{(2)}$ are the maximum possible rates of population growth, $N_r^{(1)}$and $N_r^{(2)}$ are the population sizes, $K_r^{(1)}$ and $K_r^{(2)}$ are the upper limit of population size the environment resources can support of species 1 and species 2 respectively, and $\alpha_r^{(12)}$ refers to the impact of an individual of species 2 on population growth of species 1. Actually, The Lotka-Volterra model of inter-specific competition also includes the effects of intra-specific competition on population of the species. When $\alpha_r^{(12)}$ or $N_r^{(2)}$ equals 0, the population of the species 1 will grow according to the logistic growth model which models the intra competition between individuals in a species. These equations can be used to predict the outcome of competition over time. To do this, we should determine equilibria, i.e. the condition that population growth of both species will be zero. Let $dN_r^{(1)}/dt = 0$ and $dN_r^{(2)}/dt = 0$. If $\eta_r^{(1)} N_r^{(1)}$ and $\eta_r^{(2)} N_r^{(2)}$ do not equal 0, we get two line equations which are called the isoclines of the species. They can be plotted in four cases, as are shown in Fig. 3. According to the figure, there are four kinds of competition results determined by the relationship between $K_r^{(1)}$, $K_r^{(2)}$, $\alpha_r^{(12)}$ and $\alpha_r^{(21)}$.

1. When $K_r^{(2)}/\alpha_r^{(21)} < K_r^{(1)}, K_r^{(1)}/\alpha_r^{(12)} > K_r^{(2)}$, species 1 will always win and the balance point is $N_r^{(1)} = K_r^{(1)}, N_r^{(2)} = 0$.
2. When $K_r^{(2)}/\alpha_r^{(21)} > K_r^{(1)}, K_r^{(1)}/\alpha_r^{(12)} < K_r^{(2)}$, species 2 will always win and the balance point is $N_r^{(1)} = 0, N_r^{(2)} = K_r^{(2)}$.
3. When $K_r^{(2)}/\alpha_r^{(21)} < K_r^{(1)}, K_r^{(1)}/\alpha_r^{(12)} < K_r^{(2)}$, they can win each other; the initial population of them determines who will win.
4. When $K_r^{(2)}/\alpha_r^{(21)} > K_r^{(1)}, K_r^{(1)}/\alpha_r^{(12)} > K_r^{(2)}$, there is only one balance point and they can coexist with their own population size.

For an environment that includes Ω species, the competition equation can be modified as:

$$\frac{dN_r^{(i)}}{dt} = \eta_r^{(i)} N_r^{(i)} \left(1 - \frac{N_r^{(i)} + \sum_{j=1, j\neq i}^{\Omega} \alpha_r^{(ij)} N_r^{(j)}}{K_r^{(i)}}\right). \tag{3}$$



(a) Case 1 (b) Case 2

(c) Case 3 (d) Case 4

Fig. 3. The isoclines of two co-evolution species

2) Intra-Species Evolution

Since genetic algorithm and particle filters have many common aspects, Higuchi(Higuchi, 1997) has merged them together. In CEPF the genetic operators, crossover and mutation, are applied to search for optimal samples in each species independently. The intra-species

evolution will interact with inter-species competition: the evolution of individuals in a species will increase its ability for inter-species competition, so as to survive for a longer time. Because the observation density $p(o_r^{(t)}|x_r^{(t)})$ includes the most recent observed information of the robot, it is defined as the fitness function. The two operators: crossover and mutation, work directly over the floating-points to avoid the trouble brought by binary coding and decoding. The crossover and mutation operator are defined as the following: Crossover: for two parent samples $(x_{r1}^{(t)}, w_{r1}^{(t)})$, $(x_{r2}^{(t)}, w_{r2}^{(t)})$from the samples of a species of the r-th robot. The crossover operator mates them by Equation 4 to generate two children samples.

$$\begin{cases} x'^{(t)}_{r1} = \xi x_{r1}^{(t)} + (1-\xi) x_{r2}^{(t)}, \\ x'^{(t)}_{r2} = (1-\xi) x_{r1}^{(t)} + \xi x_{r2}^{(t)}, \\ w'^{(1)}_{r} = p(o_r^{(t)}|x_{r1}^{(t)}), \\ w'^{(t)}_{r2} = p(o_r^{(t)}|x_{r2}^{(t)}). \end{cases} \tag{4}$$

Where$\xi \sim U[0,1]$, and $U[0,1]$represents uniform distribution. And two samples with the largest importance factors are selected from the four samples for the next generation. Mutation: for a parent sample $(x_{r1}^{(t)}, w_{r1}^{(t)})$ from the samples of a species of the r-th robot, the mutation operator on it is defined by Equation 5.

$$\begin{cases} x'^{(t)}_{r1} = x_{r1}^{(t)} + \tau, \\ w'^{(t)}_{r1} = p(o_r^{(t)}|x_{r1}^{(t)}). \end{cases} \tag{5}$$

Where $\tau \sim N(0,\Sigma)$ is a three-dimensional vector and $N(0,\Sigma)$ represents normal distribution. The sample with larger importance factor is selected from the two samples for next generation. In CEPF, the crossover operator will perform with probability p_c and mutation operator will perform with probability p_m. Because the genetic operator can search for optimal samples, the sampling process is more efficient and the number of samples required to represent the posterior density can be reduced considerably.

3) Splitting and Merging of Species

We assume the samples of the r-th robot are clustered into $\Omega_r^{(t)}$ species at time step t and each sample is a n-dimensional point. Samples of the i-th species $S_{ri}^{(t)} = \{({}_i x_{rj}^{(t)}, {}_i w_{rj}^{(t)}) | j = 1, \cdots, N_r^{(i)}\}$ are contained in a sub-domain $D_r^{(i)}$ which is an hypercube of the state space. If the sub-domains of two species such as sub-domain $D_r^{(i)}$ and $D_r^{(j)}$ cover each other, the two species will be merged and the their corresponding sub-domain will also be merged. We call this the merging process. Let us assume there are $\Omega'^{(t)}_r$ species after the merging process. In the splitting process the sub-domain $D_r^{(i)}$ of the i-th species is partitioned into small hyper-rectangular grids of equal size. And samples in the i-th species are mapped into the grids. The weight of each grid is the average importance factor of the samples that fall in it. A threshold $T = \mu$is used to classify the grids into two groups; here the coefficient$\mu \in (0,1)$. Grids with weight larger than T are picked out to form a grid set V. Using the network defined through neighborhood relations between the grids, the set V is divided into connected regions (i.e. sets of connected grids). Assuming there are $B_r^{(i)}$ connected regions, these connected regions are used as seeds for the clustering procedure. A city-block distance is used in the network of grids. As in image processing field, the use of distance and seeds permits to define

influence zones, and the boundary between influence zones is known as SKIZ (skeleton by influence zone) (Serra, 1982). So the sub-domain $D_r^{(i)}$ will be split into B_r^i parts. This process will perform for each species and the number of species of t+1 step is $\Omega_r^{(t+1)} = \sum_{i=1}^{\Omega'^{(t)}_r} B_r^i$. The merging and splitting process is shown in Table 1.

Input:	$\Omega_r^{(t)}$ species of time step t;
Output:	the new $\Omega_r^{(t+1)}$ species;
1.	$\Omega'^{(t)}_r := \Omega_r^{(t)}$;
2.	**for** all species **do**
3.	**if** $D_r^{(i)}$ and $D_r^{(j)}$ cover each other;
4.	merge($S_{ri}^{(t)}$,$S_{rj}^{(t)}$);
5.	$\Omega'^{(t)}_r := \Omega'^{(t)}_r - 1$; // Update the number of species
6.	**end if**
7.	**end for**
8.	$\Omega_r^{(t+1)} := 0$;
9.	**for** $i := 1$ to $\Omega'^{(t)}_r$ **do** //Split the species
10.	split the domain $D_r^{(i)}$ into grids;
11.	calculate the weight of each grids;
12.	select the grids whose weight larger than T to form set V;
13.	calculate $B_r^{(i)}$; //the number of connected regions in V
14.	split D_r^j into $B_r^{(i)}$ sub-regions and samples in each sub-region form a new species;
15.	$\Omega_r^{(t+1)} := \Omega_r^{(t+1)} + B_r^{(i)}$; //Update the number of species;
16.	**end for**
17.	**end**

Table 1. Merging-splitting process of species

4.2 Cooperative localization based on CEPF

Multi-robot localization is to integrate measurements taken at different platforms. The simplest way for integrating the information from different platforms is to maintain a single state for all the robots i.e. if there are R robots the state of the system will be of $3R$ dimension. But using particle filters for state estimation the dimension of state of the system should be small, thus estimating the distribution of the pose of all the robots is infeasible for a few robots. So a distributed representation is used in our system similar to the approach proposed by Fox (Fox et al., 2000), in which each robot maintains its own belief function that models its own uncertainty. The posterior of position is given by:

$$p(x_1^{(t)}, \cdots, x_R^{(t)} | d^{(t)}) = p(x_1^{(t)} | d^{(t)}) \cdot \cdots \cdot p(x_R^{(t)} | d^{(t)}), \tag{6}$$

Where R is the number of the robots, $d^{(t)}$is the data items collected by all robots up to time t. The distributed representation enables the estimation of the posteriors to be conveniently carried out locally on each robot. When the$l$-th robot knows the position of the r-th robot

relative to itself, information is transmitted from the l-th robot to the r-th robot and integrated in the following way:

$$\begin{aligned} p(x_r^{(t)}|d^{(t)}) &= p(x_r^{(t)}|d_r^{(t)})p(x_r^{(t)}|d_l^{(t)}) \\ &= p(x_r^{(t)}|d_r^{(t)}) \int p(x_r^{(t)}|x_l^{(t)},o_{lr}^{(t)})p(x_l^{(t)}|d_l^{(t)})dx_l^{(t)}. \end{aligned} \tag{7}$$

Where $o_{lr}^{(t)}$ is the relative positions between the two robots estimated by the l-th robot. Two cases are considered in the calculation of $o_{lr}^{(t)}$: (1) the l-th robot can see the r-th robot; (2) the r-th robot has just gone out the eyesight of the l-th robot. In the first case, $o_{lr}^{(t)}$ is observed by the l-th robot, and $p(x_r^{(t)}|x_l^{(t)},o_{lr}^{(t)})$ is the detection model of the l-th robot. In the second case, $o_{lr}^{(t)}$ is calculated using their odometry data. Since the relative positions of the two robots according to their odometry data are much more certain than the global positions, the information of their relative positions is used to refine their global positions. In both cases $p(x_r^{(t)}|x_l^{(t)},o_{lr}^{(t)})$ is learned from data. CEPF can be applied to solve Equation 7. Since samples in $S_r^{(t)}$and $S_l^{(t)}$are drawn randomly, it is not straightforward to establish correspondence between individual samples in $p(x_r^{(t)}|d_r^{(t)})$ and $\int p(x_r^{(t)}|x_l^{(t)},o_{lr}^{(t)})p(x_l^{(t)}|d_l^{(t)})dx_l^{(t)}$. And the sample set which represents the position of the r-th robot in the eyes of the l-th robot will be transmitted from the l-th robot to the r-th robot. If the sample size is large, a too wide band is needed for communication. These problems can be solved, if we turn the discrete density represented by a large set of samples into continuous density function represented by much fewer parameters or a small sub-set of samples. This can be implemented by many density estimation methods such as Gaussian mixture density estimation, SV based density estimation. But density estimation is a time- consuming process, in order to satisfy the real time requirement we choose reduced set density estimator (RSDE) (Girolami & Chao, 2003). RSDE is a kernel-based density estimator which is optimal in L_2 sense. The advantage of RSDE is that it only requires $O(n^2)$ optimization routines to estimate the required kernel coefficients, but the SV based method requires $O(n^3)$ optimization routines. The general form of density estimation using a sample set with N samples can be denoted as:

$$\hat{p}(x;h,\gamma) = \sum_{i=1}^{N} \gamma_i \kappa_h(x,x_i) \tag{8}$$

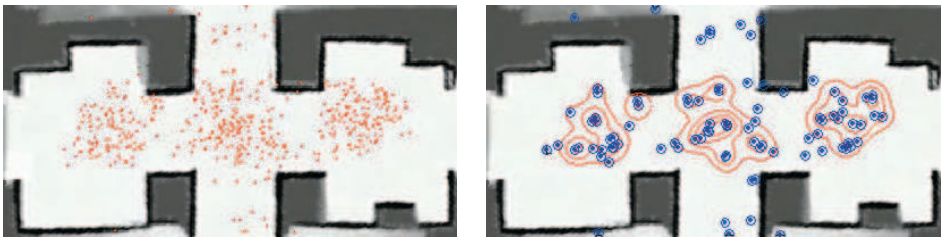
Where $\kappa_h(x,x_i)$ are kernel functions, and γ are weighting coefficients and h is the window width. For a fixed window width h, process of estimation in RESDE is to find the parameters $\hat{\gamma}$ which provides the minimum integrated squared error (ISE):

$$\begin{aligned} \hat{\gamma} &= \arg\min_{\gamma} \int_{R^d} |p(x) - \hat{p}(x;h,\gamma)|^2 dx \\ &= \arg\min_{\gamma} \int_{R^d} \hat{p}^2(x;h,\gamma)dx - 2E_{p(x)}\{\hat{p}(x;h,\gamma)\} \end{aligned} \tag{9}$$

Where $\int_{R^d} p^2(x)dx$ has been dropped from the above due to its independence of the γparameters and $E_{p(x)}\{\hat{p}(\cdot)\}$ denotes the expectation with respect to the desired density $p(x)$. Equation 9 can be represented in the following form:

$$\hat{\gamma} = \arg\min_{\gamma} \sum_{i,j=1}^{N} \gamma_i \gamma_j \int_{R^d} \kappa_h(x,x_i)\kappa_h(x,x_j) - 2\sum_{i=1}^{N} \gamma_i E_{p(x)}\{\kappa_h(x,x_i)\} \tag{10}$$

Since the required optimization of ISE will cause many of the $\gamma_i$terms to be driven to zero, the desired density can be presented with a small sub-set of samples. An example of density estimation based on RESDE is shown in Fig. 4, in which the density represented by 500 samples shown in Fig.4(a) can be estimated using 90 samples shown in Fig.4(b). Suppose the position of the r-th robot estimated by the l-th robot is represented by ${}^lN_r$ samples ${}^lS_r^{(t)} = \{({}^lx_{rj}^{(t)}, {}^lw_{rj}^{(t)}) | j = 1, \cdots, {}^lN_r\}$ after using RSDE, the algorithm of cooperative localization is shown in Table 2.



(a) Original samples (b) Sub-set of samples after density estimation

Fig. 4. Density estimation based on RDSE

5. Decision-theoretic active localization

In this section we propose a strategy based on decision theory to coordinate the robots actively. For a robot that is in a connected group, the strategy includes two steps. In the first step, the leader of a connected group will determine the position of the group (i.e. the most likely position of the robots) according to summarized global position hypotheses of robots in the group; in the second step, each robot will generate several candidate actions according to the position of the connected group and the utility and cost of the actions are calculated, then the leader will solve the conflictions between robots and choose the next action for each robot so as to maximize the utility-cost trade-off. For a robot that is not connected with any other robots, it will summarize its global position using the species with the largest average importance factor, and will explore actively taking itself as the leader.

5.1 Position estimation of the connected group

In multi-robot cooperative localization based on CEPF, each species represents a hypothesis of the position of the robot. In order to estimate the position of the connected group, the summarized hypotheses of positions and the relative positions between robots are transmitted to their leader. And the position of the leader will be estimated according to the information from other robots. Suppose the leader of a group is the l-th robot and the r-th robot is a member under its leadership. The probability that the leader will be in the position represented by the hypothesis $h_{li}^{(t)}$according to the data collected by the r-th robot and the relative positions between them can be represented by the following equation:

$$p(h_{li}^{(t)} | d_r^{(t)}) = \sum_{j=1}^{\Omega_r^{(t)}} p(h_{li}^{(t)} | h_{rj}^{(t)}, o_{rl}^{(t)}) p(h_{rj}^{(t)} | d_r^{(t)}) \tag{11}$$

Input:	NULL
Output:	the position hypotheses summarize from the species;
1.	cluster samples into $\Omega_r^{(0)}$ species,$dN_r^{(i)}/dt := 0$ and t:=1; //Initialization
2.	**for** i:=1 to $\Omega_r^{(t)}$ **do**
3.	$S_{ri}^{(t)} := \phi$;
4.	normalize importance factors of species i ; // Importance factor normalization
5.	$N_r^{(i)} := \max(N_r^{(i)} + dN_r^{(i)}/dt, 0)$ //Calculate the sample size
6.	resample $N_r^{(i)}$ samples from $S_{ri}^{(t-1)}$; // Resampling from species i
7.	**for** j:=1 to $N_r^{(i)}$ **do** // Importance sampling
8.	sample ${}_ix_{rj}^{(t)}$ from $p(x_t|{}_ix_{rj}^{(t-1)}, u_{t-1}^{)}$; // Predict next state using motion
9.	${}_iw_{rj}^{(t)} := p(o_r^{(t)}|{}_ix_{rj}^{(t)})$; // Calculate importance factor
1.	$S_{ri}^{(t)} = S_{ri}^{(t)} \cup ({}_ix_{rj}^{(t)}, {}_iw_{rj}^{(t)})$ //Incorporate the sample to $S_{ri}^{(t)}$
10.	**end for**
11.	intra-species evolution of sample set $S_{ri}^{(t)}$; // Intra-species evolution
12.	**end for**
13.	**if** the r-th robot should send information to the l-the robot
14.	estimate the state of the l-the robot and perform density estimation;
15.	**end if**
16.	**if** the robot receives the position information estimated by the l-th robot ;
17.	Merge the information from the l-th robot;
18.	**end if**
19.	**if** not end
20.	$t = t+1$ goto step 2;
21.	**end**.

Table 2. Cooperative localization of multi robots based on CEPF

Where $h_{li}^{(t)}$is a hypothesis of position summarized by the species i of the l-th robot, $h_{rj}^{(t)}$is a hypothesis of position summarized by the species j of the r-th robot, $\Omega_r^{(t)}$is the number of species of the r-th robot, $o_{rl}^{(t)}$ is the relative position between the l-th robot and the r-th robot which may not be observed directly but may be calculated according to the relative positions between the other robots in the cluster, $p(h_{rj}^{(t)}|d_r^{(t)})$ represents the probability that the r-th robot is at the position $h_{rj}^{(t)}$ and $p(h_{li}^{(t)}|h_{rj}^{(t)}, o_{rl}^{(t)})$ is a probability model learned from experiment data. The probability that the l-th robot is at the position represented by the hypothesis $h_{li}^{(t)}$ according to the data collected by all the robots in the connected group will be:

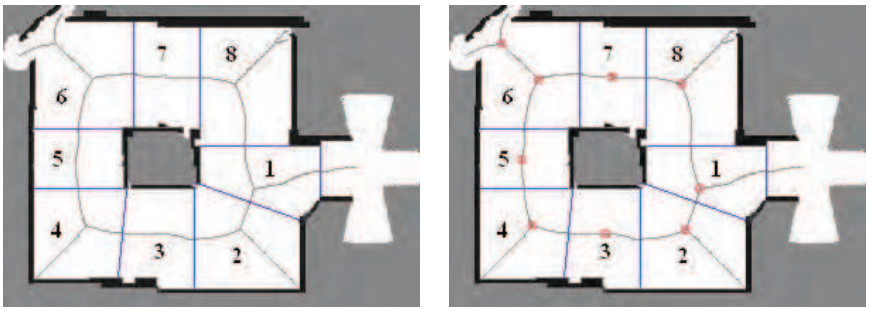
$$p(h_{li}^{(t)}|d^{(t)}) = \alpha \sum_{r=1}^{R} p(h_{li}^{(t)}|d_r^{(t)}) \tag{12}$$

Where $\alpha$is a normalization parameter to make probability of all hypotheses sum to 1, and R is the number of robots in the group. The hypothesis $h_{l\max}^{(t)}$ with the largest probability is supposed to be the position of the l-th robot. We call $h_{l\max}^{(t)}$ the position of the connected group. The hypothesis $h_{r\max}^{(t)}$ that has the largest value of $p(h_{l\max}^{(t)}|h_{rj}^{(t)},o_{rl}^{(t)})$is supposed to be the position of the r-th robot:

$$h_{r\max}^{(t)} = \arg\max_{h_{rj}^{(t)}} p(h_{l\max}^{(t)}|h_{rj}^{(t)},o_{rl}^{(t)}) \tag{13}$$

5.2 Exploration strategy

The world model in our system is represented by a hybrid map of grid and topology. For a grid map a Voronoi Diagram is produced using an approach similar to the one proposed by Thrun (Thrun, 1998). The grid map is partitioned into disjoint regions using the critical lines, as shown in Fig.5(a), and each region corresponds to one or more topological nodes which is the furcate point of Voronoi Diagram or the middle point if there is no furcate point as shown in Fig.5(b).



(a) Disjoint regions of grid map (b) Topological nodes in Voronoi Diagram

Fig. 5. Hybrid map of grid and topology

By using the position hypothesis $h_{l\max}^{(t)}$, we can get the topological nodes in the environment around the group. At any point of time the robot in each group can be assigned either to a topological node or to a rendezvous point of another robot. Coordination can be phrased as the problem of finding the assignment that maximizes the utility-cost trade-off similar to the method in (Burgard et al., 2000). More specifically, let E denote an assignment that determines the exact target (topological nodes and rendezvous point) of each robot and $E(r,j)$=1 if the r-th robot in the connected group is assigned to the j-th target. Among all assignments we choose the one that maximizes expected utility and minus expected cost:

$$E^* = \arg\max_{E} \sum_{(i,j)\in E} E(r,j)(U(r,j) - C(r,j)) \tag{14}$$

The utility and cost of each robot target pair (r,j) can be calculated as the following. **Utility:** The entropy of the belief, obtained by the following formula:

$$H(x_r^{(t)}) = \sum_{i=1}^{N_r^{(t)}} p(x_{ri}^{(t)}) \log p(x_{ri}^{(t)}), \tag{15}$$

measures the uncertainty in the robot position. Suppose the position of the r-th robot is $h_{r\max}^{(t)}$ at time point t, and it was assigned to the target j. If command a can drive the robot from the position $h_{r\max}^{(t)}$ to the target, we can measure the utility $U(r,j)$ of performing an action a by the decrease in uncertainty:

$$U(r,j) = H(x_r^{(t)}) - E_a(H(x_r^{(t+1)})). \tag{16}$$

If the target is a topological node, $E_a(H(x_r^{(t+1)}))$ denotes the expected entropy after having performed action a and having fired the sensors of the robot.

$$\begin{aligned} E_a(H(x_r^{(t+1)})) &= \sum_s H(x_r^{(t+1)}|s,a)p(s|a) \\ &= -\sum_s \sum_{x_r^{(t+1)}} p(s|x_r^{(t+1)})p(x_r^{(t+1)}|a)\log\frac{p(s|x_r^{(t+1)})p(x_r^{(t+1)}|a)}{p(s|a)} \end{aligned} \tag{17}$$

Where s is the sensor information. If the target is a rendezvous point with another robot that is not in the group, $E_a^(H(x_r^{(t+1)}))$ is the expected entropy after having performed action a and having received the information from the other robot.

$$\begin{aligned} E_a(H(x_r^{(t+1)})) &= \sum_{o_{rk}} H(x_r^{(t+1)}|d^{(t+1)},o_{rk})p(o_{rk}|a) \\ &= \sum_{o_{rk}} \sum_{x_r^{(t+1)}} p(o_{rk}|a)p(x_r^{(t+1)}|d^{(t+1)},o_{rk})\log p(x_r^{(t+1)}|d^{(t+1)},o_{rk}). \end{aligned} \tag{18}$$

Where $p(o_{rk}^|a)$ represents the probability that the relative position between the r-th robot and the k-th robot is o_{rk} after having performed the action a and the probability $p(x_r^{(t+1)}|d^{(t+1)},o_{rk})$ can be calculated by:

$$p(x_r^{(t+1)}|d^{(t+1)},o_{rk}) = p(x_r^{(t)}|d_r^{(t)}) \int p(x_r^{(t+1)}|x_k^{(t+1)},o_{rk}^{(t+1)})p(x_k^{(t+1)}|d_k^{(t)})dx_k^{(t+1)}. \tag{19}$$

Cost: using the occupancy grid map we can get the cost-optimal path from the current location of the robot to the target location. And we use the cost of following the optimal path as the cost of assigning the r-th robot to the target j. Let $p_{occ}(x_r^{(t+1)})$ denote the probability that location $x_r^{(t+1)}$ is blocked by an obstacle. The robot has to compute the probability that the target point is occupied. Recall that the robot does not know its exact location; thus it must estimate the probability that the point relative to the robot according to the action a is occupied:

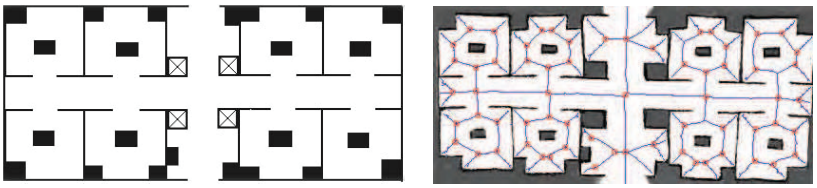
$$p_{occ}(a) = \sum_{x_r^{(t+1)}} p(x_r^{(t+1)})p_{occ}(f_a(x_r^{(t+1)})). \tag{20}$$

Where $f_a(x_r^{(t+1)})$ represents the location relative to the robot after performing action a when the robot is at the position $x_r^{(t+1)}$. Based on $p_{occ}(a)$, the expected path length and the cost-optimal policy can be obtained through value iteration. And the cost $C(r,j)$ can be calculated according to the length of the optimal path.

6. Experiment results

In this section we present experiments conducted using three robots: one is the HIT-Pioneer2 which is equipped with 16 sonar sensors and a CCD camera; one is the HIT-Pioneer3 which is equipped with 16 sonar sensors, a front laser range finder and a CCD camera and the third one is the HIT-Ghost which is equipped with two cameras (see Fig.6). The robots can detect each other using their CCD cameras, and communicate with each other using wireless Ethernet. Experiments are carried out in our lab building, and the map of the environment is shown in Fig.7(a). In the topological map are created before experiment (see Fig.7(b)). And some color marks are pasted on the wall or on the floor near the topological nodes so that HIT-Ghost can localize with visual information. Since even in multi-robot system the robots have to perform

Fig. 6. Picture of robots used for experiment

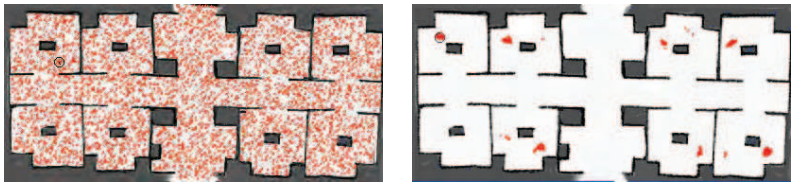


(a) The map drawn by hand (b) The map built by robot

Fig. 7. The map of the environment

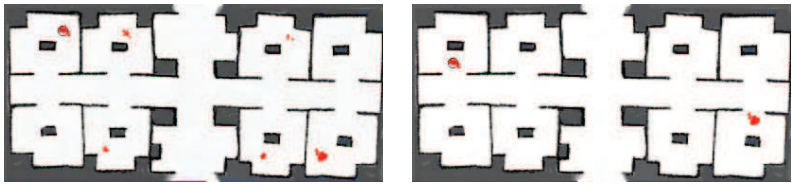
localization by itself when they do not connect with each other, we first evaluate the quality of CEPF for single robot localization. The HIT-Pioneer3 was placed in one of the rooms and its destination is the door of the building. During the process of going to its destination, the initial position of the robot is unknown and it has to make global localization using the laser

range finder (see Fig.8(a)). After randomly running several meters, most of the samples will move to several small areas because of the symmetry of the environment, so CEPF can cluster the samples into clusters naturally (see Fig.8(b)). And the robot can make decision according to the estimated location of the species. At the moment shown in Fig.8(a), the robot will make a right turning and go to another corner of the rooms to determine the room it will be in as is shown in Fig.8(c). Then the robot will go to the door of the room and only two most likely hypotheses are remained (see Fig.8(d)). Then we evaluated the multi-robot localization



(a) Initial samples

(b) Samples after running some time randomly

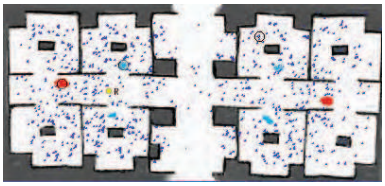


(c) Samples after making one decision

(d) Samples after making another decision

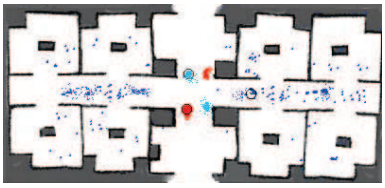
Fig. 8. Active localization of a single robot

based on CEPF with 1500 samples for each robot. The three robots are randomly placed in the environment and explore actively by themselves as in single robot localization at the first stage (see Fig.9). In figure 9 HIT-Pioneer3 is represented by red particles, HIT-Pioneer2 is represented by green particles and HIT-Ghost is represented by blue particles. When the two HIT-Pioneer robots can communicate with each other, a rendezvous point R is determined by them to manage to meet each other (see Fig.9(a)). When they can see each other their observation information will be exchanged, so their global positions can be refined and only two localization hypotheses are remained for each robot (see Fig.9(b)). And HIT-Pioneer3 is assigned to be the leader to coordinate their actions to go to an optimal topological node (see Fig.9(c)). Since the color marks in the environment are sparse and specious, the global position of the HIT-Ghost is very uncertain. When the HIT-Ghost can communicate with the two HIT-Pioneer robots, the utility to meet the other two robots is much larger than go to a topological node, so the HIT-Ghost will manage to meet the HIT-Pioneer3 (see Fig.9(d)). The time to determine the global position of the robots using no-cooperative localization, cooperative localization and cooperative active localization based on CEPF, which are termed CEPF, CCEPF and CACEPF for short respectively, are compared. In 20 times of experiments, the time needed for localization of CACEPF and CCEPF are 62% and 89% of that of CEPF respectively.



(a) Initial samples

(b) Samples after HIT-Pioneer3 meets HIT-Pioneer2



(c) Samples after making a active exploration

(d) Samples after HIT-Ghost meets HIT-Pioneer3

Fig. 9. Multi-robot active localization based on CEPF

7. Conclusion

A novel method for active localization of multi-robot is proposed. Based on the new version of particle filters called co-evolution particle filters (CEPF) the problem of premature convergence can be solved, so the hypothesis of the robots' positions can be tracked stably. And the decision theory- based coordination strategy can efficiently coordinate the action of the robots so as to maximize the utility-cost trade-off. Experimental results have proved the efficiency of the method of the active localization in multi-robot system.

8. References

Arulampalam, S. M., Maskell, S., Gordon, N. & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking, *IEEE Trans. Signal processing* Vol. 50(No. 2): 174–187.

Burgard, W., D. Fox, M. M., Simmons, R. & Thrun, S. (2000). Collaborative multi-robot exploration, *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. Vol. 1, San Francisco, USA, pp. 476 – 481.

Burgard, W., Fox, D., Hennig, D. & Schmidt, T. (1996). Estimating the absolute position of a mobile robot using position probability grids, *Proceedings of Thirteenth National Conference on Artificial Intelligence*, Oregon, USA, pp. 896–901.

Dasvison, A. J. & Murray, D. W. (2002). Simultaneous localization and map-building using active vision, *IEEE Trans. Pattern Analysis and Machine Intelligence* Vol. 24(No. 3): 865–880.

Dayoub, F. & Duckett, T. (2008). An adaptive appearance-based map for long-term topological localization of mobile robots, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, pp. 3364–3369.

Dellaert, F., Fox, D., Burgard, W. & Thrun, S. (1999). Monte carlo localization for mobile robots, *Proceedings of IEEE International Conference on Robotics and Automation*, Michigan,

USA, pp. 1322–1328.

Dissanayake, G., Newman, P. M., Durrant-Whyte, H. F. & Csorba, M. (2001). A solution to the simultaneous localization and map building (slam) problem, *IEEE Trans. Robotics and Automation* Vol. 17(No. 3): 229–241.

Doucet, A., Freitas, D. N. & Gordon, N. (2001). *Sequential Monte Carlo in Practice*, Springer-Verlag,, New York.

Doucet, A., Godsill, S. & Andrieu, C. (2000). On sequential monte carlo sampling methods for bayesian filtering8, *Statist. Computing* Vol. 10(No. 3): 19–20.

Elena, L. M., Rafael, B., Miguel, B. L. & Soledad, E. M. (2004). A human-robot cooperative learning system for easy installation of assistant robots in new working environments, *J. Intelligent and Robotic Systems* Vol. 40(No. 3): 233–265.

Fenwick, J. W., Newman, P. M. & Leonard, J. J. (2002). Cooperative concurrent mapping and localization, *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. Vol. 2, IEEE, Washington, DC, USA, pp. 1810–1817.

Fox, D. (2003). Adapting the sample size in particle filters through kld-sampling, *Internat. J. Robotic Res* Vol. 22: 985–1004.

Fox, D., B., W., Kruppa, H. & Thrun, S. (2000). A probabilistic approach to collaborative multi-robot localization, *Autonomous Robots* Vol. 8(No. 3): 325–344.

Fox, D., Burgard, W. & Thrun, S. (1998). Active markov localization for mobile robots, *Robotics and Autonomous Systems* Vol. 25(No. 3-4): 195–207.

Gil, A., Reinoso, ı., Ballesta, M. & Juliá, M. (2010). Multi-robot visual slam using a rao-blackwellized particle filter, *Robotics and Autonomous Systems* Vol. 58(No. 1): 68–80.

Girolami, M. & Chao, H. (2003). Probability density estimation from optimally condensed data samples, *IEEE Trans. Pattern Analysis and Machine Intelligence* Vol. 25(No. 10): 1253–1264.

Guivant, J. E. & Nebot, E. M. (2001). Optimization of the simultaneous localization and map-building algorithm for real-time implementation, *IEEE Trans. Robotics and Automation* Vol. 17(No. 3): 242–257.

Higuchi, T. (1997). Monte carlo filtering using genetics algorithm operators, Vol. 59: 1–23.

Hongjun, Z. & Shiyeyuki, S. (2002). Sensor planning for mobile robot localization using bayesian network representation and inference., *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, pp. 440–446.

Jensfelt, P. & Kristensen, S. (2001). Active global localization for a mobile robot using multiple hypothesis tracking, *IEEE Trans. Robotics Automation* Vol. 17: 748–760.

Kaelbling, L. P., Cassandra, A. R. & Kurien, J. A. (1996). Acting under uncertainty: Discrete bayesian models for mobile-robot navigation, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. Vol. 2, Osaka, Japan, pp. 963–972.

Milstein, A., Sanchez, J. N. & Wiamson, E. (2002). Robust global localization using clustered particle filtering, *Proceedings of Eighteenth National Conference on Artificial Intelligence*, Edmonton, Canada, pp. 581–586.

Robert, L. D. & Arvin, A. (2003). Simulation and control of distributed robot search teams, *Computers and Electrical Engineering* Vol. 29(No. 5): 625–642.

Schenker, S. P., Terry, L., Pirjanian, H. P., Bamgartner, E. T. & Tunsrel, E. (2003). Planetary rover developments supporting mars exploration, *Autonomous Robots* Vol. 14(No. 2-3): 103–126.

Serra, J. (1982). *Image Analysis and Mathematical Morphologym*, New York: Academic Press.

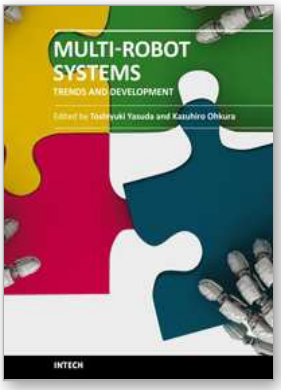
Shang, Y. & Cai, X. (1996). *Common Ecology (in Chinese)*, Beijing University Press, Beijing.

Thrun, S. (1998). Learning metric-topological maps for indoor mobile robot navigation, Vol. 99(No. 1): 21–71.

Thrun, S., Fox, D., Burgard, W. & Dellaert, F. (2001). Robust monte carlo localization for mobile robots, *Artificial Intelligence* Vol. 128: 99–141.

Trawny, G. H. N., Mourikis, A. & Roumeliotis, S. (2009). On the consistency of multi-robot cooperative localization, *Proceedings of 2009 Robotics: Science and Systems Conference*, Seattle, Washington.

Wu, M., Huang, F., Wang, L. & Sun, J. (2009). Cooperative multi-robot monocular-slam using salient landmarks, *Proceedings of the 2009 International Asia Conference on Informatics in Control, Automation and Robotics*, IEEE Computer Society, Washington, DC, USA, pp. 151–155.



Multi-Robot Systems, Trends and Development

Edited by Dr Toshiyuki Yasuda

ISBN 978-953-307-425-2
Hard cover, 586 pages
Publisher InTech
Published online 30, January, 2011
Published in print edition January, 2011

This book is a collection of 29 excellent works and comprised of three sections: task oriented approach, bio inspired approach, and modeling/design. In the first section, applications on formation, localization/mapping, and planning are introduced. The second section is on behavior-based approach by means of artificial intelligence techniques. The last section includes research articles on development of architectures and control systems.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ronghua Luo (2011). Cooperative Global Localization in Multi-Robot System, Multi-Robot Systems, Trends and Development, Dr Toshiyuki Yasuda (Ed.), ISBN: 978-953-307-425-2, InTech, Available from: http://www.intechopen.com/books/multi-robot-systems-trends-and-development/cooperative-global-localization-in-multi-robot-system

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821