The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

An Improved Clustering based Monte Carlo Localization Approach For Cooperative Multi-robot Localization

Guanghui Luo[1], Dan Wu[1], and Libing Wu[2]

[1]University of Windsor, Windsor, Ontario, Canada
[2]Wuhan Univeristy, Wuhan, P.R.China
[1]danwu@uwindsor.ca

Abstract: We describe an approach for cooperative multi-robot localization based on Monte Carlo Localization. In our approach, each of the robots maintains its own clustering based MCL algorithm, and communicates with each other whenever it detects another robot. We develop a new information exchange mechanism, which makes use of the information extracted from the clustering component, to synchronize the beliefs of detected robots. By avoiding unnecessary information exchange whenever detection occurs through belief comparison, the proposed approach solves the delayed integration problem which further improves the effectiveness and efficiency of multi-robot localization. This approach has been tested in both real and simulated environments. Compared with single robot localization, the experimental results demonstrate that proposed approach notably improves the performance, especially when the environments are highly symmetric.

Keywords: clustering, MCL, multi-robot localization

I. INTRODUCTION

Mobile robot localization, the process of determining the position and orientation (pose) of a robot within its operating environment from sensor reading, is a prerequisite for subsequent high level navigation tasks. It has been considered as one of the fundamental problems in mobile robotics [6]. The most widely studied localization problems are: Local localization (position tracking), which is to compensate incremental errors in a robot's motion under the assumption that the initial position is known as prior, and the more challenge Global localization, in which the robots are required to estimate their pose by local and incomplete observed information under the condition of uncertain initial position [7].

During the past two decades, most existing work has focused on single robot localization, such as Grid-based approaches [1], Monte Carlo Localization [5, 11], and multi-hypothesis approaches [2]. These approaches have been applied to many different applications in solving localization problem and achieve remarkable successes.

However, more and more researchers are interested in using multiple robots to improve efficiency and robustness. In this paper, we propose an efficient probabilistic approach for cooperative multi-robot localization in indoor environment. Our approach is based on Monte Carlo Localization that has been applied with great practical success to single robot localization. In our approach, the robots, capable of sensing and exchanging information one with another, localize themselves by maintaining their own beliefs which are the proposed clustering based MCL algorithm. Our new developed information exchange mechanism is employed to synchronize each robot's belief whenever one robot detects another in order to speed up the localization process with possible higher accuracy. We utilize the information extracted from the clustering component of the proposed approach which analyzes the distribution of the whole particle set to quantify robot's belief and transfers information across different robots if necessary. We always compare the beliefs of both detected robots before synchronizing their poses' estimate to control information integration. By doing so, we can prevent the localization process from suffering the problem of delayed integration, which means to avoid unnecessary information exchange. In addition, robots themselves are implicitly used as landmarks rather than using only external landmarks, as in other works.

In section II we briefly review previous cooperative localization approaches, the MCL algorithm, and the clustering method. Section III describes details of our proposed method. Section IV presents the experimental results. Finally, section V provides the conclusions and future work.

II. RELATED WORKS

Many robotic applications require that robots work cooperatively in order to perform a certain task [9]. Knowing their global positions is the first step in multi-robot systems.

One early cooperative localization technique with multiple robots was proposed by Kurazume and Nagate [8] ans is known as "portable beacons". The basic underlying idea is that each robot repeats move-and-stop actions and serves as a landmark for the other robots. Only a part of entire robots can move in a certain time instant and this dramatically slows down the overall localization speed.

An EKF-based approach for cooperative multi-robot localization was introduced by Roumeliotis [11]. This approach allows all robots in a group to move simultaneously, and to propagate their state and covariance estimates independently by decomposing the centralized EKF-based cooperative localization into N communicating filters. However, during each update cycle, all robots need to communicate with each other and update the covariance matrix for all pose estimates. The main drawback is that the high cost of computation and communication limits this approach to small robot teams in real-time operation.

Fox and Burgard [4] have proposed a different implementation of cooperative multi-robot localization schema which extends the Monte Carlo Localization (MCL) algorithm. The sample-based version of Markov localization enables localizing mobile robots in any-time fashion. Each robot in the system maintains a probability distribution describing its own pose estimate. When one robot detects another, the "detection model" is used to synchronize the individual robot's beliefs, thereby introducing additional probabilistic constrains which ties one robot's belief to another robot's belief to reduce the uncertainty. This method suffers from the problem of delayed information exchange, which means the information exchange between two detected robots cannot ensure it is always benefit the localization process.

Our proposed approach is a clustering based MCL algorithm for cooperative multi-robot localization. The details of MCL and the related clustering algorithm will be given in this section. We first review some basic concepts in mobile robot localization. The *state* of a robot is defined as the collection of all aspects of the robot and its environment. The state at time t is denote as x_t. The pose of a robot is denoted (x, y, θ), where x and y represent a two-dimensional coordinate and θ represents orientation of the robot. The notion of *belief* is used to represent the robot's internal knowledge about the state of the environment [10]. The expression of belief over state x_t is $bel(x_t) = p(x_t | z_{1:t}, u_{1:t})$. It is a posterior probability over states conditioned on all the past motion data $u_{1:t}$ and all the past measurement data $z_{1:t}$. Localization algorithms based on probabilistic methods have two different components to process these two kinds of data [13]: measurement model, denoted as $p(z_t | x_t)$, and motion model, denoted as $p(x_t | u_{t,} x_{t-1})$.

A. Monte Carlo Localization

Monte Carlo Localization (MCL) is one of the latest and commonly used probabilistic approaches for single robot localization. MCL is an implementation of Bayes Filter, which represents uncertainty by maintaining a set of weighted samples that are randomly drawn from the probability density [19].

MCL is a recursive algorithm, and Fig. 1 [19] shows one iteration. The inputs of the MCL algorithm are particle set X_{t-1} from previous iteration, control data u_t, measurement data z_t, and the given map m of the environment. Line 1 initializes the particle set $\bar{X}_t$ and X_t. And then for each particle, line 3 do sampling from the motion model, line 4 calculates the importance weight of that particle from the measurement model. Line 7 to line 10 is the resampling phase. The algorithm draws with replacement N particles from the temporary set $\bar{X}_t$. The probability of drawing each particle is given by its importance weight. Finally, the posterior particle set X_t, which contains the particles that with higher importance weights is returned.

Algorithm MCL ($\boldsymbol{X_{t-1},\ u_t,\ z_t,\ m}$)
1: $\bar{X}_t = X_t = \emptyset$
2: for n =1 to N do
3: $X_t^{[n]}$ = sample_motion_model (u_t, $X_{t-1}^{[n]}$)
4: $w_t^{[n]}$ = measurement_model (z_t, $X_t^{[n]}$, n)
5: $\bar{X}_t = \bar{X}_t + \langle X_t^{[n]}, w_t^{[n]} \rangle$
6: endfor
7: for n = 1 to N do
8: draw i with probability $\propto w_t^{[i]}$
9: add $x_t^{[i]}$ to X_t
10: endfor
11: Return X_t

Fig. 1. The Monte Carlo Localization (MCL) algorithm [19].

Note that, if MCL finishes successfully, most particles are concentrated on a small region which represents the location of robot, although there is not such a stop condition in the MCL algorithm.

B. Clustering Algorithm BSAS

Clustering is the assignment of a set of observations (particles) into clusters so that observations in the same cluster are similar to each other. Two important parts in almost all clustering algorithms are to select a proximity measure, which measures the dissimilarity between particles or clusters, and the representative of a cluster, which highly depends on the shape of a cluster. As in MCL all poses of particles are represented by a two dimensional coordinate system, it's more effective for us to choose the Euclidean distance measure $d(P_i, P_j) = \sum \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ as our proximity measure. A Compact cluster is what one mostly encounters in MCL, thus the mean point of a cluster is chosen as representative, denoted as $P_{mean} = \frac{1}{N} \sum P_i$, N is the number of particles contained in that cluster.

We choose one of the most efficient clustering methods called Basic Sequential Algorithm Scheme (BSAS) shown in Fig. 2. to be used in the proposed method, because most probabilistic robotics algorithms need real time performance.

The BSAS algorithm takes the whole particle set $X(x_i, \ldots x_N)$ that need to be clustered and the user defined threshold of dissimilarity θ as inputs. Line 1 initialize the first cluster, which contains the first particle x_1 presented to the algorithm. From line 2 to line 9 it is a large for loop sequentially going through all the remaining particles. The dissimilarity measures between current particle and every existing cluster is calculated in order to find a minimum one in line 3. From line 4 to line 8, if the minimum measure calculated in line 3 is greater than θ, a new cluster that containing current particle will be created. Otherwise, the considered particle will be assigned to the existing cluster which has the minimum dissimilarity measure to it and update its representative of this cluster.

Algorithm BSAS ($\boldsymbol{X(x_i, \ldots x_N), \theta}$):

1: $m = 1, C_m = \{x_1\}$
2: for $i = 2$ to N do
3: find $C_K: d(x_i, C_k) = min_{1 \le j \le m} d(x_i, C_j)$
4: if $d(x_i, C_k) > \theta$ then
5: $m = m + 1, \; C_m = \{x_i\}$
6: else
7: $C_k \cup \{x_i\}$, update its representative
8: endif
9: endfor

Fig. 2. The Basic Sequential Algorithm Scheme (BSAS) [12].

III. COOPERATIVE CLUSTERING BASED MCL ALGORITHM

Motivated by both the MCL and the BSAS algorithms above, one novel method incorporating clustering into the conventional MCL is proposed for cooperative multi-robot localization in order to further improve the localization performance, as outlined in Fig. 3.

A. MCL+BSAS part

Our proposed approach consists of three parts. In the first part, each robot maintains its own belief by executing the MCL and the BSAS algorithms. The conventional MCL is used and the generated particle set is then supplied to the BSAS algorithm for clustering (the clustering component). By doing so, the proposed approach could monitor the localization process of each robot in real time by making use of the cluster information obtained from BSAS, such as the number of clusters, denoted as n_c, the locations of these clusters, denoted as $pose(a_1, \cdots, a_n)$, the possibility of each cluster represent the true location of the robot. If the percentage p_{max} of particles in the cluster which contains the largest number of particles exceeds a predefined threshold η, then the considered robot is assumed to have been localized, and return the pose of representative of the cluster to be used as a landmark for the other robots. If p_{max} is less than the predefined threshold η, then it indicates the robot fails to localize itself at the moment. The ability that recognizing whether a robot has been l localized or not by itself significantly improves its robustness and autonomy.

B. Detection part

The core idea of multi-robot localization is to let the robot benefit from information collected by others, therefore the interactions between robots is essential to the proposed approach. Robot can perceive other robots in the environment, and then exchange their information with each other if possible to reduce their uncertainty about their external environment. In part 2, the proposed approach manages the behavior of both detected robots in a principled way in order to dealing with the delayed information exchange problem.

Algorithm: Clustering based MCL for multi-robot localizatoin

Part 1:

Initially, each robot executes its own MCL+BSAS to monitor its own pose estimate and cluster information.

If the degree of certainty p_{max} exceeds a predefined threshold η, the considering robot will stop to indicate it has been localized and return its location to be used as landmark for other robots.

Part 2:

When one robot detects another:

(a) If the values of both detected robots' status variables are false, there is no information exchange;

(b) Else there is one robot's status variable is true or both of them are true, then do belief comparison:
If they are not with the same degree of certainty, do information exchange which use the one with higher certainty to help the other one to refine its pose estimate;
Else there is no information exchange;

Part 3:

Process of information exchange:

(a) After determining the exchange direction, a predefined distance threshold γ is used to help in refining process:
Those clusters of the robot with lower certainty, whoever is within the range of predefined distance threshold γ to any clusters of the robot with higher certainty, will be kept;

(b) Return a new set of clusters which will focus on the more possible locations of the robot;

(c) Uniformly resample N total particles within the newly returned set of clusters;

Fig. 3. The Clustering based MCL for cooperative multi-robot localization

The behavior is mainly to determine whether to allow an information exchange and the direction of information exchange, with the help of one important Boolean type status variable for each robot described in Fig. 4. Initially, each status variable is set to be "False". During the entire localization process, once a robot has detected any landmark, its status variable will be set to "True", and once a robot has exchanged its belief with another robot, the status variable will be set to "False". If one robot has succeeded in its localization, its Boolean value will always stay in "True"

We manage the detection between robots in two scenarios: (1) in the first scenario, both status variables of two detected robots are false. This scenario includes two situations: (a) both robots have not detected any landmarks yet. Here we can assume that the internal beliefs of both robots about their poses relative to the environment are very blurry.

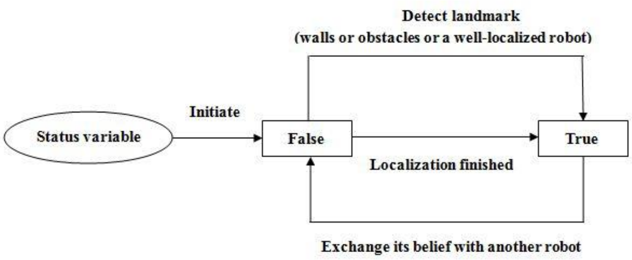


Fig. 4. The status variable diagram.

We skip the information exchanging at the moment since the blurry knowledge will not benefit the localization process. (b) Both robots have already exchanged their beliefs at last encounter or detection, and no landmarks have been observed by both robots yet at current detection. Without a newly observation of landmarks the level of uncertainty will most likely stay the same level or even worse. Normally there is no need to exchange information again. Therefore our approach does not allow the information exchanging under this kind of situations. (2) In the second scenario, there is at least one status variable value is true or both are true. Since perceiving a landmark helps robot gather more information about its environment so it could help in its pose estimate, we consider it as necessary to exchange their beliefs to refine the pose estimates. The direction of refining belief is to use the robot's belief which is with higher certainty about where it is to refine the other with lower centainty about its pose and hence it requires a comparison of the level of certainties about their locations. In order to quantify the belief, our approach makes use of the variable p_{max} extracted from the clustering component. Through comparing of p_{max} of both robots we can find which one is with higher certainty about its position. If they are with the same level of certainty, then we consider it as no need to exchange their information, however, this situation rarely happened in experiments.

Through this strategy our approach makes sure that the information exchange always benefits the localization process.

C. Information exchange part

As mentioned above, each particle or cluster represents one possible position of robot. If two robots detect each other, their estimated poses which are particles or the representatives of clusters should be within a certain distance. Our proposed approach makes use of this geometric relationship to build a mechanism for information exchange. Fig. 5 shows the pseudo code of the process of information exchange. It takes both sets of clusters of robot A and B, and the predefined threshold distance γ as inputs. The direction of information exchange has been determined by part 2, thus, we have two cases: (1) robot A refines robot B from line 1 to 10; (2) vice versa robot B refines robot A. In case 1, line 2 initializes a new set of cluster C'_B. From line 3 to line 8, it is a for loop going through all clusters of robot B. Line 4 calculates the distance between the current cluster of robot B and all clusters of robot A to find the minimum one, if the minimum distance is smaller than the threshold γ, then add it to the set of cluster C'_B. Line 10 uniformly resamples the total number of particles within the newly returned set of clusterC'_B. The process of case 2 is the same as case 1 but with different direction of information exchange.

Information exchange
($\mathbf{C_A}(\mathbf{a_1},\cdots,\mathbf{a_m}), \boldsymbol{C_B}(\boldsymbol{b_1},\cdots,\boldsymbol{b_n}),\boldsymbol{\gamma}$):

1: *Case 1: Robot A refines Robot B*
2: $C'_B = \emptyset$
3: for *i =1* to *n* do
4: find $d(b_i, C_A) = min_{1\le j \le m} d(x_i, C_j)$
5: if $d(b_i, C_j) < \gamma$ then
6: add b_i to C'_B
7: endif
8: endfor
9: return C'_B
10: uniformly resample particles within C'_B
11: Case 2: Robot B refines Robot A
12: *vice versa*

Fig. 5. The process of information exchange.

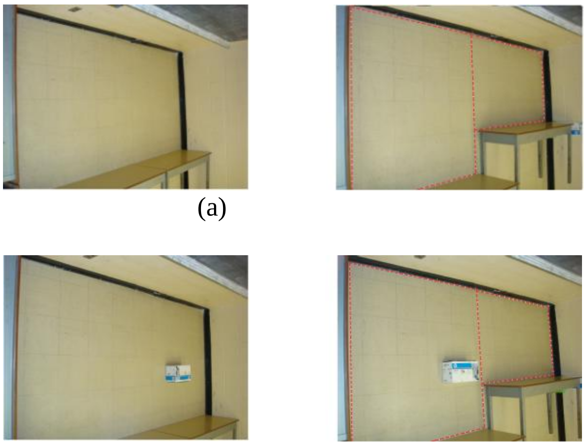
IV. EXPERIMENTAL RESULT

Our approach is tested on both real and simulated robots. In both situations, we firstly test the performance of single mobile robot localization, and then apply our approach to multi-robot localization. Three important data: the localization time *t(s)*, the successful rate ε, and the error distance d, will be collected in every experiment. Finally we compare these characteristic results of multi-robot localization with single robot localization, and demonstrate improvement.

A. Experiments Using Real Robots

Fig. 6. Two iRobot Create using in our experiments. The left Create is with black label, while the right one is regular Create.

Fig. 6 shows the two real robots iRobot Create using in our experiments. The experiments are conducted in four different environments shown in Fig. 7. The two Creates started from different locations. The paths of both Create is set to turn left 120° when it bumps to the wall or obstacle, otherwise keep going forward. The total number of particles used to represent the belief of Create is set to 5000. The dissimilarity measure threshold θ is set to 17cm which is the radius of Create. Therefore the shape of each cluster will be the same size as Create. The threshold γ used in the process of information exchange is set to be 60cm since the setup of detection is determined by the range of infrared signals emitted by Virtual wall sensor. Threshold η is set to 70%, which means if the number of particles contained in the largest cluster is equal or larger than 70% of total, the Create will stop to indicate it has been localized. We repeat each experiment 50 times and compare the performance to MCL+BSAS for single robot which ignores robot detections.



(a)

(b)

(c)

(d)

Fig. 7. Four environments of real robot experiments. (a) A rectangle field with the shape of 300cm×150cm; (b) Two rectangle fields, the left one is 150cm×150cm, the right one is 150cm×100cm; (c) A rectangle field of 300cm×150cm, and an obstacle of 31.5cm×19.5cm placed in the field; (d) Two rectangle fields, the left one is 150cm×150cm, the right one is 150cm×100cm, and an obstacle of 31.5cm×19.5cm placed in the field.

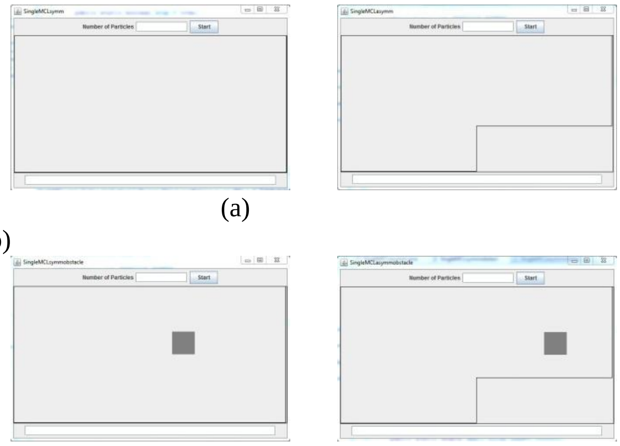
TABLE I.
COMPARE MULTI-ROBOT LOCALIZATION WITH SINGLE ROBOT LOCALIZATION USING CREATE

Environment		t (s)	Time saving (%)	ε	Increasing of successful rate (%)
Symmetric	Single robot	230.88	37.6%	42%	30%
	Multi robot	144.04		72%	
Asymmetric	Single robot	81.6	25.6%	84%	2%
	Multi robot	60.73		86%	
Symmetric with obstacle	Single robot	92.62	24.9%	80%	4%
	Multi robot	69.54		84%	
Asymmetric with obstacle	Single robot	74.38	30.4%	90%	1%
	Multi robot	51.76		91%	

The results shown in Table I demonstrate that compare to single robot localization, our proposed approach applied in multi-robot localization not only reduces the time for localization, but also increase the successful rate for each robot. We can see without the help of detection of other Create and exchange information between them the time for single robot localization in symmetric environment is 230.88s, and the successful rate is only 42%. With the help of another Create, the results show significant improvement, which reduce the time for localization by 37.6% to 144.04s in average of two Create, and increase the successful rate by 30% to 72%. The large improvement of time saving also shows in the other three environments. However our proposed approach can only increase the successful rates by relatively small ranges in the other three environments. This is because these three environments are very distinctive which means even single Create can achieve high successful rate.

B. Simulation Experiments

The simulation experiments are also implemented in four different environments shown in Fig. 8. Same as in the real robots experiments, both multi-robot localization using our proposed approach and single robot localization have been tested under these four environments. The total number of particles is also 5000. The threshold γ is set to 120 pixels which is the same setup of detection with experiments using Create. The dissimilarity threshold θ is set to be 60 pixels, and threshold η is also set to 70%. We repeat each experiment 50 times.



(a)

(b)

(c)

(d)

Fig. 8. Four environments of simulation experiments. (a) A rectangle field of 600×300pixel; (b) Two rectangle fields, the left one is 300×300pixel, the right one is 300×200 pixel; (c) A rectangle field of 600×300pixel, and a gray rectangle obstacle of 50×50pixel placed in the field; (d) Two rectangle fields, the left one is 300×300 pixel, the right one is 300×200pixel, and a gray rectangle obstacle of 50×50pixel placed in the field.

Table II summarizes that our proposed approach can significantly shorten the time for localization in both symmetric environment and featured environments compare with single robot localization. As to the successful rate, it depends on the type of environments. The improvement is obviously while localizing in symmetric environments.

TABLE II
COMPARE MULTI-ROBOT LOCALIZATION WITH SINGLE ROBOT LOCALIZATION IN SIMULATION EXPEIMENTS

Environment		t (s)	Time saving (%)	ε	Increasing of successful rate (%)
Symmetric	Single robot	368.02	70.6%	44%	32%
	Multi robot	109.56		76%	
Asymmetric	Single robot	31.4	28.3%	86%	3%
	Multi robot	22.485		89%	
Symmetric with obstacle	Single robot	33.24	27%	84%	4%
	Multi robot	24.25		88%	
Asymmetric with obstacle	Single robot	28.74	31.6%	92%	1%
	Multi robot	19.65		93%	

C. Study of parameter η

As we mentioned the value of η determines the stop point of each robot's localization which is specified by the minimum percentage of total particles contained in the largest cluster. In this group of experiments, we increase the value of η into 80%, while other parameters stay the same values with previous simulation experiments, to see how it can affect the performance.

TABLE III
COMPARISON OF MULTI-ROBOT LOCALIZATION UNDER TWO VALUES OF η

Environment	Multi robot	t (s)	Time increasing (%)	ε	Increasing of successful rate (%)
Symmetric	η=70%	109.56	18%	76%	2%
	η=80%	129.41		78%	
Asymmetric	η=70%	22.485	14.5%	89%	3%
	η=80%	25.75		92%	
Symmetric with obstacle	η=70%	24.25	16.9%	88%	4%
	η=80%	28.36		92%	
Asymmetric with obstacle	η=70%	19.65	8.5%	93%	1%
	η=80%	21.33		94%	

The results show that a large value of η will take more time to finish the localization process because it requires more information about its external environment to reduce the uncertainty of position estimation. In the mean time, the fact that more particles fall in the largest cluster indicates more likely the pose of representative of this cluster represents the true pose of the robot, which means the successful rate or effectiveness will be increased. However, there is a trade-off between efficiency and effectiveness controlled by η. If η is too big, it will take significantly longer time to localize itself while achieving a relatively small increase of successful rate. On the other hand, if η is too small, despite the rapid reduction of localization time, without guaranteeing the successful rate it will not make any sense to the localization.

V. Conclusion and Future Works

This paper presents a clustering based MCL algorithm for cooperative multi-robot localization, in which all robots moved simultaneously. Each robot maintains its own clustering based MCL algorithm, and communicates with each other whenever it detects another robot. The newly developed mechanism for the communication aims to achieve improve the efficiency and effectiveness of localization process. In addition, the characteristic of without fusion center and the instant communication between two detected robots allow our proposed approach scaling to large group of robots. Compare with single robot localization, experimental results performed in both real and simulated environments demonstrate the improvements in both efficiency and effectiveness of our proposed approach applied in cooperative multi-robot localization.

In the future work, we want to design a more robust way of movement based on already gathered information rather than the simple static path setting. We also want to test our approach on a large group of robots.

ACKNOWLEDGEMENTS The second author would like to thank NSERC for financial support.

REFERENCES

[1] A, Howard, M, J Mataric, and G, S Sukhatme. Cooperative relative localization for mobile robot teams: An ego-centric approach. In Proc. of The naval Reasearch Laboratory Workshop on Multi-Robot Systems (2003), 65-95.
[2] A, Gasparri, S, Panzieri, F, Pascucci, and G, Ulivi. A Hybrid Active Global Localization Algorithm for Mobile Robots. International Conference on Robotics and Automation (April 2007), 10-14.
[3] D, Wu, J, Chen, and Y, Wang. Bring Consciousness to Mobile Robot Being localized. Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (741-746 2009).
[4] D, Fox, W, Burgard, H, Kruppa, and S, Thrun. A Probabilistic Approach to Collaborative Multi-robot Localization. Autonomous Robots on Heterogeneous Multi-Robot System, vol.8, no.3 (June 2000), 325-344.
[5] Esha, D Nerurkar, Stergios, I Roumeliotis, and Agostino, Martinelli. Distributed Maximum A Posteriori Estimation for Multi-robot Cooperative Localization. IEEE International Conference on Robotics and Automation (May 12-17 2009), 1402-1409.
[6] H, Hose and H, L Akin. The Reverse Monte Carlo localization algorithm. Robotics and Autonomous Systems (2007), 480-489.
[7] Maja, J Mataric. The robotics primer. The MIT Press, 2007.
[8] P, Trahanias and E, Scordalakis. An efficient sequential clustering method. Pattern Recognition, vol.22(4) (1989), 449-453.
[9] R, Kurazume, S, Nagata, and S, Hirose. Cooperative positioning with multiple robots. In Proc. 1994 IEEE Int. Conf. Robotics and Automation, vol.2 (May 8-13 1994), 1250-1257.
[10] S, Thrun, D, Fox, and W, Burgard. Monte carlo localization with mixture proposal distribution. Proceedings of the AAAI National Conference on Artifical Intelligence (2000), 859-865.
[11] S, I Roumeliotis and G, A Bekey. Distributed multi-robot localization. IEEE Transactions on Robotics and Automation, vol.18, no.5 (October 2002), 781-795.
[12] S, Theodoridis and K, Koutroumbas. Pattern Recognition. Academic Press, 2006.
[13] S, Thrun, W, Burgard, and D, Fox. Probabilistic Robotics. MIT Press, 2006.