Behaviour of Multiagent System with Defined Goal

Jana Púchyová[*]
Department of Technical Cybernetics
Faculty of Management Science and Informatics
University of Žilina
Univerzitná 8215/1, 010 26 Žilina, Slovakia
jana.puchyova@fri.uniza.sk

Abstract

The subject of this work is the area of multiagent and multirobot systems focusing on their behaviour in the field of terrain exploration. The main goal is to design the multirobot system which is able to complete exploration task with the possibility of any robot failure. The proposed techniques consider the inaccuracies and errors arising in the system. In paper is described the design, as well as the verification of marking coverage algorithm with shortened return for speeding up of enviroment coverage. The next part verifies behaviour of multirobot system with use of diferent level of information fusion. Created conclusions about system composition and the way of cooperative information processing provides basic starting points for experiments, that verify cooperative behaviour of multirobot system and its resistance against inaccuracies and failures of robots. The task was experimentally verified on target localization during the exploration, while for error estimation the theory of probabilistic robotics is used. The errors were suppressed using technique of cooperation, Extended Kalman filter (EKF), Particle filter (PF) and Particle filter with weighting. Experiments were performed on static and also mobile system.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*; I.4.3 [**Artificial Intelligence**]: Enhancement—*Filtering*; B.8.1 [**Performance and reliability**]: Reliability, Testing and Fault-Tolerance

Keywords

multiagent system, multirobot system, probabilistic robotics, robotic failure, filtration, Particle filter, Extended Kalman filter

1. Introduction

The use of robotics in terrain exploration task is standard task, though designed algorithms are often built only for single agent solution. Some algorithms can be used only in testbed experiments, the others can be used only in known terrain. Latest algorithms serve for shortering of time coverage with usage of reactive robots and are capable of covering unknown terrain. Their simplicity is their big advantage, however the speed of whole covering process is still not optimal. Therefore one of the task is to create such an algorithm, which can be used by a group of robots, which shortens the time of exploration and at the same time can be robust in case of any agent failure.

One of the advantages of multiagent systems is effective utilization of computing performance. System, which is capable of cooperation, can processes available information with higher accuracy thanks to its resources. So one of the tasks is to verify how the amount of resources and the level of parallelism in multirobot system can affect the quality of solution, and also how the previous factors are affecting the communication requirements.

The process of searching and also the precise localization of target is suitable task for usage of fault-tolerant multirobot system. It is expected that in target localization task it will be possible to suppress the failures not just using the collective task accomplishment. The goal is to find appropriate technique which results will be comparable with already applicated methods. It is necessary to point on the fact that the usage of multirobot systems can not just suppress the failures arising in system, but it is also robust for possible failure of any system element.

After the application of proposed suggestions, the result will be fault-tolerant multirobot system for exploration tasks.

2. Matematical Model of MRS in exploration task

Proposed macroscopic model of multirobot system for terrain exploration which consider the failure possibility, is described here. This model comes from mathematical modeling of swarm robotics [5, 6].

Figure 1: Model of MRS for terrain exploration

In Figure 1 is depicted model of multirobot system for terrain exploration where S_e - state of exploration (coverage and target searching), S_t - state of announcing the

[*] Recommended by thesis supervisor: Prof. Juraj Miček and Assoc. Prof. Peter Gubiš. Defended at Faculty of Management Science and Informatics, University of Žilina on August 20, 2013.

Púchyová, J.: Behaviour of Multiagent System with Defined Goal. Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 4 (2013) 15-24

target finding and its position, S_f - state of robot failure, p_t - probability that the target was found, p_{ft} - probability of robot failure in announcing state, p_{fe} - probability of robot failure in exploration state, t_t is time used for announcing the target position, N_e is number of robots in exploration state and the number is changing in time according to relation:

$$N_e(k+1) = N_e(k) - p_t.N_e(k) - p_{fe}.N_e(k) + p_t.N_e(k-t_t), \quad (1)$$

N_f is number of robots in failure state and their amount is changing according to relation:

$$N_f(k+1) = N_f(k) + p_{fe}.N_e(k) + p_{ft}.N_t(k) \quad (2)$$

and N_t is number of robots in announcing state and it is changing according to relation:

$$N_t(k+1) = N_0 - N_e(k+1) - N_f(k+1), \quad (3)$$

where N_0 is total number of robots in multirobot system for terrain exploration and it stands:

$$N_e(k) = N_t(k) = N_f(k) = 0,\ k < 0. \quad (4)$$

Index k is time index corresponding to continuous time t_k, $T_{k-1} \triangleq t_k - t_{k-1}$.

Let $N(k) = [N_e(k)\, N_t(k)\, N_f(k)]^T$, then in initial state are all the robots are in exploration state, so:

$$N(0) = [N_e(0)\, N_t(0)\, N_f(0)]^T = [N_0\, 0\, 0]^T\,. \quad (5)$$

It is good to note, that the failure state is permanent in this model and the failed robots stays in this state for all the time:

$$\begin{aligned} N_e^* &= \lim_{k\to\infty} N_e(k) = 0 \quad (6)\\ N_t^* &= \lim_{k\to\infty} N_t(k) = 0\\ N_f^* &= \lim_{k\to\infty} N_f(k) = N_0\\ N^* &= \left[N_e^*\, N_t^*\, N_f^*\right]^T = [0\, 0\, N_0]^T \end{aligned}$$

where $N^*(k) = \lim_{k\to\infty} N(k)$.

3. Marking Coverage Algorithm with Shortened Return

The task of terrain exploration is one of the array where the multiagent system is used very often. But recent algorithms are usually design for known terrain or robots are using complex devices for exploration.

Proposed algorithm comes from Spanning Tree Covering (STC) [1, 9]. The aim of algorithm is to reduce number of robots steps.

3.1 Principles of algorithm

The aim of robots is to cover whole terrain together. Terrain is divides into square cells and their size is dependent of used sensor:

$$w < d < r_{sensor}, \quad (7)$$

where w width of robot, d is size of the cell and r_{sensor} is sensor range.

Robot used marks during the covering:

- *visited* - cell which robot covered already,
- *occupied* - cell is occupied by obstacle or it was covered by other robot,
- *viewed* - cell is without obstacle, robot saw this cell during the coverage but it does not cover it,

Robot used in its coverage the left hand rule. Thanks to this algorithm, after first round is able to know, how many cells is in the terrain. Principle of algorithm is described in algorithm 1.

Algorithm 1 Marking coverage algorithm with shortened return

```
1: Check if this cell has mark. If not and there is no obstacle, mark this cell with sequential number - first visited cell number is 1, etc. Inform other robots.
2: Mark the 4 adjacent cells:
     • if you have never been there and there is no visible obstacle - mark as viewed cell,
     • if there is an obstacle and is not possible to go there, set as occupied,
     • if the cell was visited before, do not change the mark.
3: Choose the viewed cell as the next cell in this order - at first left cell, then the cell in front of you, then right cell and then go back.
4: If none of the adjacent cell is not only viewed, check, how many cells were only viewed. If none of the cell has the mark viewed, choose as the next cell the cell with the lowest mark number. Otherwise choose the cell with lower number than your current cell has.
5: If you are at visited cell No. 1 stop; otherwise go to step 1.
```

If we assume that shortened return is possible only when every cell has mark *visited* or *occupied*, algorithm is similar to STC algorithm, according to return path. But if we know information about the terrain characteristics, it is possible to shorten the return. Coefficient c_{env} is set according to terrain complexity and it tell about percentage of that cells that can be inaccessible because of the obstacles. If the number of unknown cells does not reach the percentage bigger than c_{env}, shortened return can be used.

3.2 Experimental results

Experiments was made in multirobot simulation system called Player/Stage [3]. This system is able to simulate also the errors and inaccuracy of robots, so the simulation is similar to real robot behaviour. So in case, where is need to turn robot for 90°, coverage of this cell lasts for 2 basic steps, in case of turn for 180°, it lasts for 3 basic steps. Coefficient of terrain complexity is set to 0.04.

Algorithm was tested on terrain exploration with one robot. Explorated terrain is depicted in Figure 2 (a). Terrain is devided into square cells and some of them are with obstacles.

Final number of steps for terrain covering with one robot is depicted in Table 1.

Table 1: One robot terrain covering

Whole number of cells: 256 , number of occupied cells: 86			
Agent	*No. of covered cells*	*No. of steps in shortened return*	*No. of steps from start to return*
A1	166	41	275

Table 2: Two agent terrain coverage

Whole number of cells: 256 , Number of occupied cells: 86			
Agent	*No. of covered cells*	*No. of steps in shortened return*	*No. of steps from start to return*
A1	93	49	
A2	72	60	
Together	162		161

(a) (b)

Figure 2: Terrain and simulation of covering by (a) one agent, (b) by two agents

Terrain for experiments with two homogeneous robots is the same as in one robot coverage (Figure 2 (b)). Agents start from the same cell. Agents made marks and according to them were made the spanning trees - Figure 3 (a). From this figure is obvious that agents covered whole terrain. Cell with mark $\check{S}$ is starting cell, cell P_i is the last cell covered with i-th agent.

In Figure 3 (b) is depicted shortened return of both agents. Both agents used also the cells covered with other agent for shortened return. Experiment evaluation in the way of number of steps is in Table 2.

Despite of similarity of algorithm, there is no problem to assure the robustness of the system. In case, when som agent fail, it stops to send information about terrain coverage and its task is made with other agents.

(a)(b)

Figure 3: (a) Coverage spanning trees made by two agents, (b) shortened returns of robots after coverage

At the end of the Table 2 is show summarization for the whole multiagent system. There is written the number of steps from robot with longer coverage. In comparison with Table 1, whole number of basic steps is decreased from 275 na 161, which is improvement of 41,5 %.

4. Information Fusion in Multirobot System

One of the suspected advantage of using multirobot system is to increase the quality of solution. In that case is system used often in tasks, where all the system element jointly sense information from environment and they evaluate it cooperatively. Quality of solution is dependent not only on the number of elements in systems, but also level of information fusion.

Use of different level of information fusion was experimentally verified on command classification task. The goal of the system is to sense the commands with robots and then to recognize it. For recognition a simple algorithm with low computational complexity is used. The aim of this task is to improve quality of recognition, but also to decrease the demands on communication subsystem.

Figure 4: Multirobot system for command recognition task

In Figure 4 is depicted MRS for command recognition, where individual robots are assigned as MR_k, k represents number of robot and dk represents the distance of k-th robot from the speaker. Suppose the acoustic signal, received by the sensors of individual robots, is:

$$\mathbf{S}_k(t) = a_k\mathbf{x}(t-\tau_k) + e_k(t), \qquad (8)$$

where a_k is a parameter, which expresses the attenuation of signal generated by the speaker and it is dependent on the distance between the sensor and the speaker d_k, τ_k is time delay of acoustic signal on the way between the sensor of k-th robot and the speaker, $e_k(t)$ is white Gaussian noise interference, with zero mean value, $e_k(t) = \mathrm{N}(0, \sigma)$.

Active robot (AR) is robot which identifies the speech's beginning.

Principle of command recognition algorithm is described in [7]. Result of the whole process of recognition is cepstral coefficients vector. Each new-sensed vector of acoustic signal goes through recognition process and its cepstral coefficients vector is compared according to Euclidean distance with vector from reference group. The smallest Euclidean distance determines the biggest similarity with corresponding command from reference group. Euclidean distance is given by the Pythagorean theorem.

In experiments the sampling frequency of $16\,kHz$ was used in resolution of 12 bits. After identification of beginning of speech, each robot obtain 16384 samples, it is $1024\,ms$. In reference group, there is 5 commands: ”back”, ”go”, ”left”, ”right” and ”stop”.

Despite the information fusion was software tested, in design the hardware restrictions of multirobot system Georges were take into account [8].

4.1 Experimental results

Experiments of command recognition with different level of information fusion was made on MRS with changing number of active robots. In testing the value of signal-to-noise ratio was changing. The goal is to point out, how the selection of proper level of information fusion can increase the reliability of classification.

4.1.1 Data fusion

Data fusion (*scenario 1*) was first tested level of information fusion. Used MRS has centralized architecture and it is focused on robot computational cost reduction. System is composed of M robots evenly distributed within the area and one - reference robot (RR) evaluates the command for the whole system. Data fusion puts high requirements on the communication subsystem, due to amount of the transmitted data form RR to other participants and vice versa.

Table 3: Euclidean distances obtained while recording of signal by (a) two (b) four active robots; data fusion

(a)

<table>
<tr><th colspan="6">Euclidean distances; SNR = 30 dB, 2 active robots</th></tr>
<tr><th><i>Pronounced command</i></th><th><i>EB</i></th><th><i>EG</i></th><th><i>EL</i></th><th><i>ER</i></th><th><i>ES</i></th></tr>
<tr><td><i>”back”</i></td><td>2</td><td>2349</td><td>1127</td><td>1981</td><td>1025</td></tr>
<tr><td><i>”go”</i></td><td>2359</td><td>12</td><td>2051</td><td>2126</td><td>2447</td></tr>
<tr><td><i>”left”</i></td><td>1131</td><td>2053</td><td>0</td><td>1610</td><td>941</td></tr>
<tr><td><i>”right”</i></td><td>1972</td><td>2106</td><td>1604</td><td>5</td><td>1402</td></tr>
<tr><td><i>”stop”</i></td><td>1033</td><td>2433</td><td>940</td><td>1409</td><td>0</td></tr>
</table>

(b)

<table>
<tr><th colspan="6">Euclidean distances; SNR = 30 dB, 4 active robots</th></tr>
<tr><th><i>Pronounced command</i></th><th><i>EB</i></th><th><i>EG</i></th><th><i>EL</i></th><th><i>ER</i></th><th><i>ES</i></th></tr>
<tr><td><i>”back”</i></td><td>0</td><td>2350</td><td>1128</td><td>1977</td><td>1021</td></tr>
<tr><td><i>”go”</i></td><td>2364</td><td>6</td><td>2056</td><td>2117</td><td>2446</td></tr>
<tr><td><i>”left”</i></td><td>1127</td><td>2055</td><td>0</td><td>1612</td><td>942</td></tr>
<tr><td><i>”right”</i></td><td>1977</td><td>2110</td><td>1607</td><td>2</td><td>1398</td></tr>
<tr><td><i>”stop”</i></td><td>1032</td><td>2431</td><td>941</td><td>1407</td><td>0</td></tr>
</table>

As it is visible from Table 3, Euclidean distances between pronounced command and correct element from reference group decreases with increasing number of active robots in system. According to expectations, data fusion ensures that recognition is more reliable, when the signal was sensed with higher number of robots.

A load distribution during the classification is uneven, which results from a centralized system architecture. There is however a reduction of computational requirements for individual robots and the transfer of the requirements to RR.

Dependence of Euclidean distance from number of robots and also from the level of noise is better visible on the graph shown in Figure 5. The experiments were made while spoken the ”go” command.

Figure 5: Graph of dependencies of Euclidean distances from number of the robots and noise level

4.1.2 Feature Fusion

For feature fusion, MRS with centralized architecture was used. The goal is to decrease the requirements on communication subsystem.

Each active robot is capable of recording and processing of audio signal into the Mel-frequency cepstral coefficients. So it obtains vector 256 x 12 bits. Each robot then transmits obtained data to reference robot in defined time slot (Time Division Multiple Access is used)(*scenario 2*).

Used fusion can be modified in that way that Mel-frequency cepstral coefficients vector is compared by Euclidean distance with coefficients from reference group. The calculated vector of distances is transmitted by each robot to RR - *scenario 3*.

Table 4: Euclidean distances obtained while recording of signal by (a) two (b) four active robots; feature fusion

(a)

<table>
<tr><th colspan="6">Euclidean distances; SNR = 30 dB, 2 active robots</th></tr>
<tr><th><i>Pronounced command</i></th><th><i>EB</i></th><th><i>EG</i></th><th><i>EL</i></th><th><i>ER</i></th><th><i>ES</i></th></tr>
<tr><td><i>”back”</i></td><td>4,5</td><td>2352</td><td>1126,5</td><td>1980,5</td><td>1023,5</td></tr>
<tr><td><i>”go”</i></td><td>2367,5</td><td>27</td><td>2059</td><td>2127,5</td><td>2451,5</td></tr>
<tr><td><i>”left”</i></td><td>1134</td><td>2054,5</td><td>1,5</td><td>1611</td><td>939</td></tr>
<tr><td><i>”right”</i></td><td>1976,5</td><td>2101,5</td><td>1605</td><td>5,5</td><td>1402</td></tr>
<tr><td><i>”stop”</i></td><td>1031,5</td><td>2431</td><td>939,5</td><td>1406</td><td>0</td></tr>
</table>

(b)

<table>
<tr><th colspan="6">Euclidean distances; SNR = 30 dB, 4 active robots</th></tr>
<tr><th><i>Pronounced command</i></th><th><i>EB</i></th><th><i>EG</i></th><th><i>EL</i></th><th><i>ER</i></th><th><i>ES</i></th></tr>
<tr><td><i>”back”</i></td><td>3,25</td><td>2350,5</td><td>1124,25</td><td>1978,25</td><td>1021</td></tr>
<tr><td><i>”go”</i></td><td>2365,25</td><td>28,25</td><td>2060,25</td><td>2123</td><td>2450,25</td></tr>
<tr><td><i>”left”</i></td><td>1132,25</td><td>2052,5</td><td>1,5</td><td>1610,75</td><td>942</td></tr>
<tr><td><i>”right”</i></td><td>1975,75</td><td>2107,75</td><td>1605,5</td><td>4,5</td><td>1401</td></tr>
<tr><td><i>”stop”</i></td><td>1030,75</td><td>2430,5</td><td>940,25</td><td>1407,75</td><td>0</td></tr>
</table>

<table>
<tr><th colspan="6">Euclidean distances; SNR = 7 dB, 4 active robots</th></tr>
<tr><th><i>Pronounced command</i></th><th><i>EB</i></th><th><i>EG</i></th><th><i>EL</i></th><th><i>ER</i></th><th><i>ES</i></th></tr>
<tr><td><i>”back”</i></td><td>63,75</td><td>2328,25</td><td>1125,75</td><td>1974,5</td><td>1042,5</td></tr>
<tr><td><i>”go”</i></td><td>2389,75</td><td>322</td><td>2069,75</td><td>2090,25</td><td>2377,75</td></tr>
<tr><td><i>”left”</i></td><td>1142</td><td>2042,25</td><td>52,75</td><td>1606,75</td><td>942</td></tr>
<tr><td><i>”right”</i></td><td>1961</td><td>2076</td><td>1598</td><td>78,75</td><td>1404</td></tr>
<tr><td><i>”stop”</i></td><td>1029</td><td>2421,75</td><td>936,25</td><td>1409,75</td><td>9,5</td></tr>
</table>

Results for feature fusion are presented in Table 4. It is obvious, that also in this scenario Euclidean distance between pronounced command and corresponding reference command decreases with increasing number of robots and decreasing level of noise.

Compared to previous scenario, there has been an increase of computational requirements on the individual robots. Such a system has thus more even load distribution between individual elements of the system. The role of the final decision is however still in the hands of RR.

In case of scenario 3, robots send to RR calculated Euclidean distances, not cepstral coefficients. The main difference from scenario 2 is therefore only an increase of the level of distributed data processing and the reduction of communication subsystem load. Though at the same time, there are imposed higher requirements on memory subsystem of the robots. Nevertheless, the results were the same as was mentioned in Table 4.

4.1.3 Decisions Fusion

Decisions fusion (*scenario 4*) is focused on total decrease of communication requirements and uniform computational load of individual robots. In this case each robot computes Euclidean distance between sensed acoustic signal $\mathbf{S}_k(t)$ and reference group, choose the command with the smallest Euclidean distance and sends the decision to RR.

Considering that using the given algorithm all robots achieved the right decision, Euclidean distances in the tables for scenario 4 should match with tables for scenario 2 and 3. The way how can be this scenario modified, is the exclusion of the results of that robot, which achieved the highest Euclidean distance in its decision about the best fit spoken command. Thus achieved results can be seen in the Table 5.

Table 5: Euclidean distances obtained while recording of signal by (a) two, (b) four active robots; decisions fusion

(a)

Euclidean distances; SNR = 30 dB, 2 active robots					
Pronounced command	*EB*	*EG*	*EL*	*ER*	*ES*
"back"	4	2349	1128	1982	1020
"go"	2379	16	2066	2133	2460
"left"	1131	2058	1	1615	941
"right"	1972	2100	1603	5	1400
"stop"	1030	2432	940	1407	0

(b)

Euclidean distances; SNR = 30 dB, 4 active robots					
Pronounced command	*EB*	*EG*	*EL*	*ER*	*ES*
"back"	2,67	2349	1124	1978	1019
"go"	2368,33	25	2063	2123,33	2452,67
"left"	1132	2053,33	1,33	1610,67	941
"right"	1976	2104,67	1606,67	4	1401
"stop"	1030	2430,67	940,67	1408,67	0

In this case was the computational load of individual robots relatively uniform.

4.2 Evaluation of experiments

In each scenario, the same acoustic signals were used, so the scenarios can be compared in conclusion.

Figure 6: Scenarios comparison when using various SNR

From the graph in figure 6 it is obvious, how the increasing SNR influences the accuracy of classification, which is represented by Euclidean distance of cepstral coefficients vector between reference and tested pronounced command. Scenarios comparison is based on the results for signal recording by four AR. Graph in Figure 6 documents the result, that the lowest Euclidian distance was reached in the scenario 1. With the decreasing SNR is this fact visible more expressively.

Figure 7: Load of communication subsystem

Different approaches can be compared also in terms of communication subsystem load. In Figure 7 is shown the load when one active robot sends necessary data to RR with using five model commands. Communication subsystem load in this case is independent from noise level.

From the results it is obvious, that scenario 1, which reaches the best results in recognition, burdens the communication subsystem the most from given scenarios. Valuation of other scenarios is also possible to read from the graphs in the Figure 6 and 7. To select the appropriate scenario it is necessary to consider the solution reliability, distributed data processing level, requirements for memory subsystem of robots and also communication subsystem requirements.

5. Target Localization with Multirobot System With Suppressing Measurement Error

Target localization consists of characteristic measurement evaluation (radioactivity, temperature etc.) that it is possible to set place of source. In this part, the static target is localized. Concerning errors, which arises in system, in position measurement is possible failure of robot.

Task of target $\mathbf{T}$ localization by group of robots $\mathbf{R}_j, j = 1, ..., n$ in time period k is depicted in Figure 8. Each robot provides measurement $\mathbf{z}_k^{(R_j)}$ of target position, uses it for its own estimation of target position $\mathbf{x}_k^{(R_j)}$. This estimation is then sent for collectively evaluation to reference robot. In figure, the reference robot is R_1.

Figure 8: Target localization task with use of multirobot system

Experiments made for verification of reliability and precision of multirobot system was generated for static and also mobile system. So closing to jointly estimated position of target is expanded technique for suppressing the system error.

For the both system were used several techniques for suppressing error - Extended Kalman filter, Particle filter and combination of Particle filter and weighting of partial estimations. Expected errors in localization task are inaccuracy due to sensor characteristics and the environment noise. At the same time the failure of robots is considering in system.

According to results from previous part, system consists of homogeneous robots and for jointly evaluation the decisions fusion is used, so the requirements on communication subsystem are the smallest and the computational load of the robots is relatively uniform.

Suppose that robots knows their initial position. Then the target position can be evaluated according to sensor measurement. Define the vector of robots positions $\mathbf{R} = \{\mathbf{R}_j, j = 1, ..., n\}$ where j is robot number and n is the number of robots. Each robot estimates the position of target as:

$$\mathbf{x}_k^{(R_j)} = (r_k^{(R_j)} \quad \alpha_k^{(R_j)}). \tag{9}$$

It means that target position is determined by the distance between target and $j-th$ robot $r_k^{(R_j)}$, and the angle between target and robot $\alpha_k^{(R_j)}$; k is corresponding to the continuous time t_k.

The jointly estimated position of target

$$\mathbf{T}_k^{est} = (x_k^{(T^{est})} \quad y_k^{(T^{est})}) \tag{10}$$

and then the position of $j-th$ robot

$$(R_j)_k = (x_k^{(R_j)} \quad y_k^{(R_j)}) \tag{11}$$

are expressed by two coordinates in Cartesian coordinate system.

Note that measurements from sensors have form of measurement vector:

$$\mathbf{z}_k^{(R_j)} = (z(r^{(R_j)})_k \quad z(\alpha^{(R_j)})_k). \tag{12}$$

Robots independently localize position of the target using its sensors (algorithm 2). Their estimations are evaluated centrally in reference robot, where the jointly estimation is evaluated. Reference robot compute the target position according to algorithm 3.

Algorithm 2 Target localization - part for each $j-th$ robot of MRS

```
1: $k \leftarrow 1$;
2: set your new position;
3: make new measurement of target position $\mathbf{z}_k^{(R_j)}$ (12);
4: (use filtering;)
5: send $\mathbf{x}_k^{(R_j)}$ to central point;
6: $k \leftarrow k+1$;
7: go to step 2;
```

Algorithm 3 Target localization by multirobot system - reference robot

```
1: $k \leftarrow 1$;
2: make n-angle with the estimated position as vertices;
3: compute the centre $\mathbf{T}_k^{est}$ of this n-angle;
4: $k = k+1$;
5: go to step 2;
```

For the needs of method evaluation, the real position of target was used:

$$\mathbf{T}^{real} = (x^{(T^{real})} \quad y^{(T^{real})}). \tag{13}$$

Error size, which is used for evaluation and comparison of algorithm (in graphs is marked as *err*), was set as:

$$\begin{aligned} \mathrm{err} &= |\mathbf{T}^{real} - \mathbf{T}_k^{est}| \\ &= \sqrt{\left(x^{(T^{real})} - x_k^{(T^{est})}\right)^2 + \left(y^{(T^{real})} - y_k^{(T^{est})}\right)^2}; \end{aligned} \tag{14}$$

A noise, which occurs on sensors that localize target, has normalized normal probability distribution N(0,1) and the matrix of measurement noise goes from real characteristics of ultrasonic sensor SRF08:

$$\mathbf{R}_k = \begin{bmatrix} 0,4 & 0 \\ 0 & 0,131 \end{bmatrix}. \tag{15}$$

Simulation of multirobot system in target localization task was performed in Mobile Robot Programming Toolkit (MRPT) [2].

5.1 Suppression of error using cooperation

When the target is searching by stationary robots, the group of robots can be considered as wireless sensor network (WSN). Many different errors, that occur in the system, are suppressed only using joint calculation. As can be seen from graph in Figure 9 (a), with rising number of robots it comes to more balanced elimination of error and the progress graph of estimation error reaches smother behaviour.

(a)(b)

Figure 9: Dependency of estimation error size according to amount: (a) static robots, (b) mobile robots; use of cooperation

In case of mobile robots, their position changes in every step according to relation:

$$(\mathbf{R_j})_k = \begin{cases} x_{k-1}^{(R_j)} + \cos(\alpha_{k-1}^{(R_j)}).K, & \text{for } x_{k-1}^{(R_j)} < x_{k-1}^{(T^{est})} \\ x_{k-1}^{(R_j)} - \cos(\alpha_{k-1}^{(R_j)}).K, & \text{for } x_{k-1}^{(R_j)} \geq x_{k-1}^{(T^{est})}, \\ & \\ y_{k-1}^{(R_j)} + \sin(\alpha_{k-1}^{(R_j)}).K, & \text{for } y_{k-1}^{(R_j)} < y_{k-1}^{(T^{est})} \\ y_{k-1}^{(R_j)} - \sin(\alpha_{k-1}^{(R_j)}).K, & \text{for } y_{k-1}^{(R_j)} \geq y_{k-1}^{(T^{est})}, \end{cases} \tag{16}$$

So new position of robot depends on previous position of robot, on jointly estimated position of target and from the size of step of robot K. The size of step in the tests was set to 0.05 m.

All the tests were conducted similarly as in the case of static robots. The measurement error was in this case at the same time suppressed by nearing of robots to the goal. Also in case of graph in Figure picture 9 (b) it is obvious, that with rising amount of active robots, the estimation error of position is decreasing.

Table 6: Means and measurement error variance in target locating task in MRS with different number of robots without using filter; (a) static robots, (b) mobile robots

(a)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,60373	0,51197	0,21385
2	0,45894	0,41012	0,0687
3	0,39796	0,35239	0,05272
4	0,34746	0,32467	0,03774
5	0,33936	0,28177	0,04420
6	0,28007	0,27785	0,02007
7	0,25783	0,2233	0,01927
8	0,25769	0,25448	0,01345
9	0,26406	0,25300	0,01396
10	0,22189	0,20244	0,0135

(b)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,47849	0,38283	0,12090
2	0,32258	0,26346	0,06227
3	0,27550	0,22727	0,03428
4	0,22639	0,17089	0,02382
5	0,19803	0,15882	0,02046
6	0,19800	0,17748	0,01698
7	0,17825	0,16244	0,01236
8	0,16041	0,15175	0,00802
9	0,15473	0,13269	0,00872
10	0,15875	0,14188	0,00881

For better comparison, Table 6 was created, which contains arithmetic average, median and error variance with different amount of active elements. It is obvious that the less values of statistical file characteristics are occurred, the more accurate is the target position estimation.

Comparing the both Tables 6 (a) and (b) it can be concluded that the value of all characteristics of statistic file is decreasing with rising amount of robots. In case of Table (a) there was best result achieved by the highest amount of active elements. In case of mobile robots there occurred heteroclitic state, when was the accuracy of calculation higher with less amount of robots. The system in this case reached already in case of 8-th robots very accurate results. As can be seen from values of median, the values for 8 to 10 robots was so close, that can be considered as constant after slight rounding. The table shows in conclusion, that the error of measurement is at the same time suppressed with the motion of robots to estimated position of target.

5.2 Suppression of error using cooperation and Extended Kalman filter

Except the cooperation, the Kalman filter was used for the error filtration in this part. Every robot adjusts its estimation according to new measurement using filter, and into the calculation of joint estimation will contribute with already filtered value. Also in this case it is obvious from the graph, that the error is suppressed and the behavior is smoothed with rising amount of robots (Figure 6.3) (a). Even in technique of suppressing of error with Extended Kalman filter can be seen, that in case of mobile robots, the error of common estimation err is decreasing with rising time (picture 6.3) (b). The error decreases in the same time with the amount of robots, which are localizing the target.

(a)(b)

Figure 10: Dependency of estimation error size according to amount: (a) static robots, (b) mobile robots; use of cooperation and EKF

In Table 7, where characteristics of statistic files for both used types of systems are shown, several deviations from the expected values were appeared. Similarly to the case described in part 5.1, from point of amount of active elements the static system can be considered as saturated by usage of ten robots. The increase of error with four static robots was caused by wrong estimation of robot number 4. By its connecting to the system, it significantly affected the size of jointly estimation error. However the errors arising in system, where are the inaccuracies suppressed with Extended Kalman filter, are so suppressed, that the estimation with static system was more accurate using nine robots as it was in case of mobile system.

5.3 Suppression of error using cooperation and Particle filter

In case of Particle filter, there was used the version Sequential Importance Resampling (SIR) [4]. SIR algorithm is derived from basic SIS algorithm, but the new samples are estimated from prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and the resampling phase is executed in each time step. Considering the oversampling is executed in every time step k, then $w^i_{k-1} = 1/N$ for all particles $i = 1, 2, ..., N$. At he same time the weights computation will simplified:

$$w^i_k \propto p(\mathbf{z}_k|\mathbf{x}^i_k), \tag{17}$$

while the normalization of weights is executed before resampling. For resampling is used algorithm of systematic resampling in this case.

The results of filtration error in jointly estimation with use of Particle filter are displayed in Figure 11. For the test was used filter, which consists from 5000 particles. From the behaviour of both graphs can be seen that the size of error is in static system suppressed more significantly with the rise of amount of robots. In case of mobile elements of system is the error more significantly filtered not just with the rising amount of robots, but also with rising time.

Table 7: Means and measurement error variance in target locating task in MRS with different number of robots using EKF; (a) static robots, (b) mobile robots

(a)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,25270	0,19584	0,04217
2	0,16197	0,16024	0,00752
3	0,15152	0,13553	0,00680
4	0,17715	0,17401	0,00510
5	0,12779	0,09472	0,00899
6	0,10861	0,10331	0,00231
7	0,11599	0,10508	0,00361
8	0,08303	0,07413	0,00176
9	0,07840	0,07211	0,00194
10	0,10550	0,08683	0,00469

(b)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,22309	0,16438	0,04570
2	0,16159	0,12375	0,01839
3	0,12385	0,10741	0,00439
4	0,13505	0,11734	0,00695
5	0,09580	0,08127	0,00505
6	0,10075	0,08534	0,00564
7	0,09766	0,07245	0,00550
8	0,07295	0,06526	0,00253
9	0,08164	0,08018	0,00151
10	0,06940	0,05764	0,00296

(a) (b)

Figure 11: Dependency of estimation error size according to amount: (a) static robots, (b) mobile robots; use of cooperation and PF

Comparison of statistic files characteristics for target localization by different amount of robots is shown in Table 8. Even in this case the characteristics with the rising amount of element of system have decreasing tendency. The exception is the state, when the system comes to ”saturation” and usage of higher amount of robots is not necessary.

5.4 Suppression of error using cooperation, Particle filter and weighting

Particle filters, which were used in recent and previous algorithm, filters just errors occurring by the measurements of individual robots. In the system, there arise also other errors, for example the errors of Particle filters. This inaccuracies are partly suppressed with jointly calculation, however in this case the jointly estimation is strengthening by weighting of results from individual robots.

Robots localize individually of target with use of its sensors (algorithm 2). Their estimations are collectively processed on the base of weighting of individual robots estimation. Collectively estimated position of target is

$$\mathbf{T}^{est}_k = f(\mathbf{x}^{(R_j)}_k, \mathbf{p}^{\mathbf{R}}_k), \tag{18}$$

where $\mathbf{p}^{\mathbf{R}}_k$ is weight of robots in system.

$$\mathbf{p}^{(R_j)}_k = f(|\mathbf{x}^{(R_j)}_{k-1} - \mathbf{T}^{est}_{k-1}|, r^{(R_j)}_k), \tag{19}$$

it means that robot weight in jointly target estimation po-

Table 8: Means and measurement error variance in target locating task in MRS with different number of robots using PF; (a) static robots, (b) mobile robots

(a)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,22506	0,19208	0,02466
2	0,23320	0,19073	0,02861
3	0,18541	0,14639	0,01456
4	0,16644	0,14326	0,01292
5	0,10444	0,09421	0,00288
6	0,13308	0,12840	0,00556
7	0,12965	0,12883	0,00356
8	0,12812	0,11393	0,00414
9	0,09486	0,08582	0,00290
10	0,12102	0,10216	0,00761

(b)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,22692	0,19130	0,02581
2	0,16660	0,14359	0,01148
3	0,13084	0,09808	0,01074
4	0,11481	0,08077	0,00838
5	0,11481	0,08077	0,00838
6	0,09264	0,08367	0,00358
7	0,08539	0,06532	0,00379
8	0,07108	0,06106	0,00196
9	0,08427	0,06843	0,00368
10	0,07543	0,06889	0,00334

sition $\mathbf{T}^{est}$ depends on distance between robot and target $r_k^{(R_j)}$ it time step k, and distance between robot's estimation and jointly position estimation $|\mathbf{x}_{k-1}^{(R_j)} - \mathbf{T}_{k-1}^{est}|$.

Formally the calculation of m-th robot's weight can be expressed as:

$$\begin{aligned}\mathbf{p}_p^{(R_m)} &= C_p \frac{\sum\limits_{j=1}^{n} \frac{1}{\left|\mathbf{x}_{k-1}^{(R_j)} - \mathbf{T}_{k-1}^{est}\right|}}{\left|\mathbf{x}_{k-1}^{(R_m)} - \mathbf{T}_{k-1}^{est}\right|} + \frac{\sum\limits_{j=1}^{n} \mathbf{x}_k^{(R_j)}}{\mathbf{x}_k^{(R_m)}} \qquad (20)\\ \mathbf{p}_k^{(R_m)} &= N_v \frac{\mathbf{p}_p^{(R_m)}}{\sum\limits_{j=1}^{n} \mathbf{p}_p^{(R_j)}},\end{aligned}$$

where $\mathbf{p}_p^{(R_m)}$ is partial computation of weight in which the value of C_p determines the experimentally set constant. This means the rate, of how the position estimation in previous step $k-1$ contributes into computation. The constant N_v in the weight computation $\mathbf{p}_k^{(R_m)}$ gives the normalization to the number of vertexes, which will be used for collective estimation according to algorithm 4. It is obvious, that to the common estimation, the computation of robot m is counted $\mathbf{p}_k^{(R_m)}$-times.

Algorithm 4 Target localization by multirobot system with use of Particle filter and weighting - reference robot

```
1: $k \leftarrow 1$;
2: compute weights $\mathbf{p}_k^{\mathbf{R}}$;
3: set $N_v$ vertices according to estimations of each robot
   and their weights;
4: make n-angle with $N_v$ vertices from step 3;
5: compute the centre $\mathbf{T}_k^{est}$ of n-angle;
6: $k = k + 1$;
7: go to step 2;
```

With this weighting it is assured, that robot which has the best estimation of target position according to joint estimation in time step $k-1$, gets higher weight in joint computation in time step k. Also the robot which is nearer to target according to its estimation, it is considering as more reliable element.

By the simulations, which results are shown in Figure 12, where the constants C_p and N_v was set to 100. Used Particle filter was composed of 5000 particles.

(a) (b)

Figure 12: Dependency of estimation error size according to amount: (a) static robots, (b) mobile robots; use of cooperation, PF and weighting

As can be seen not just on graphs, but also from Table 9, thanks to weighting of robots, the progress of error graph is smoother, which describes very low values of variation $D(err)$. Even in this case the "saturation" of system with amount of active elements has appeared.

Table 9: Means and measurement error variance in target locating task in MRS with different number of robots using PF and weighting; (a) static robots, (b) mobile robots

(a)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,22504	0,22046	0,02000
2	0,21863	0,15567	0,03178
3	0,14307	0,12068	0,00827
4	0,14746	0,11736	0,00869
5	0,14188	0,11843	0,00854
6	0,10709	0,10237	0,00276
7	0,13773	0,13110	0,00461
8	0,10844	0,08882	0,00522
9	0,09475	0,07769	0,00422
10	0,11019	0,10398	0,00485

(b)

No. of robots	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
1	0,23438	0,19568	0,03242
2	0,17774	0,14518	0,01405
3	0,16967	0,15092	0,01067
4	0,09750	0,07763	0,00831
5	0,10024	0,10284	0,00286
6	0,09861	0,09492	0,00260
7	0,07487	0,07423	0,00137
8	0,07180	0,06462	0,00194
9	0,07084	0,05972	0,00217
10	0,07013	0,06147	0,00212

5.5 Comparison of used algorithms

Comparison of used techniques for suppressing arising error in target localization task using static robots is shown in Figure 13. Comparison tests were performed by usage of four robots with sensors. From the graph it is obvious, that in case of target localization without filters, the error *err* is highest and at the same time the graph line is less smooth. Particle filter without weighting and Extended Kalman filter are generating similar values. As expected, the Particle filter with weighting of partial results has best ability to suppress the localization error and at the same time it filters the possible measuring fluctuations.

Comparison of techniques results for system created by mobile robots is shown in Figure 14. Even in this case it is obvious, that the best filtering algorithm is usage of

Figure 13: Comparison of results in static multirobot system consisting of 4 robots

MRS, PF and weighting of partial results. This algorithm has the most accurate target position estimation, and also it can suppress possible fluctuations in target localization.

Figure 14: Comparison of results in mobile multirobot system consisting of 4 robots

Used algorithms are compared in Table 10. Based on statistical characteristics it can be said, that algorithm of collective localization with use of Particle filter and weighting achieves the best results in suppressing of error in target localization task. This fact is valid not just from point of median of measured errors, but also from point of variance. In case of mobile robots, the variance reached better values with use of pure Particle filter, but median of error was significantly higher as in case of weighting. However the graph in Figure 14 documents, that filter with weighting needed first 22 steps for stabilization, and then the measurement error was stabilized. The variance after stabilization of filter would reach the value of 0,000472 m^2.

Table 10: Means and measurement error variance in target locating task in MRS with different number of robots using various algorithms; (a) static robots, (b) mobile robots

(a)

4 static robots in target localization task			
Filter	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
without filter	0,2415	0,2082	0,0223
EKF	0,1126	0,0863	0,0098
PF	0,1145	0,0980	0,0093
PF and weighting	0,0796	0,0743	0,0019

(b)

4 mobile robots in target localization task			
Filter	$\overline{err}[m]$	$\widetilde{err}[m]$	$D(err)[m^2]$
without filter	0,1359	0,1122	0,0119
EKF	0,0810	0,0623	0,0045
PF	0,0733	0,0600	0,0028
PF and weighting	0,0592	0,0316	0,0042

From all of tests and results it is obvious, that using group of mobile robots in target localization task brings not only the redundancy in case of any robot failure, but it also provides the possibility of suppressing the measurement inaccuracies and environment errors. For better suppression of inaccuracies it is possible to use higher amount of robots, use Particle filter and weighting of partial results in collectively calculation of target position. With this combination of elimination techniques it was achieved suppression of measurement, smoothing the localization process and also speeding up of target localization with sufficient accuracy. As it is obvious from the figures and the tables the conclusion is that Particle filters with use of multirobot systems are suitable tools for the target localization task.

6. Conclusions

Main goals of this work is to demonstrate, that use of multiagent and multirobot system brings radical solution improvement in some tasks. Aim is to focus on advantages of use in exploration task.

Exploration task was devided in two partial task, which where tested separately. For terrain coverage was designed marking algorithm for shortened return. This algorithm is able to shorten number of steps according to Spanning Tree Covering and with increasing number of agents, the time of covering has shorten. This conclusions are very useful in tasks, where exploration time is very important. It can be tasks, where searched element can endanger surround environment or searched element can be engadered by stay in environment.

In next experiments it was verified results for levels of information fusion in multirobot system. As was verified, with increase of transferred information, computational requirements on each element of system decrease, but there is higher load of communication subsystem. If the system is able to transfer huge amount of information, for increasing the solution precision, data fusion is appropriate level. For decreasing the amount of communication, which can be in noisy environment really limited, is better to use decisions fusion and to increase number of active robots.

For target localization task in ideal environment is sufficient to use only one robot. But in environment, which is source of uncertainty and the system is affected by inaccuracies or the robots can fail, the task accomplishment is very endangered. Putting more elements assure not only increase of solution quality, but also assure redundancy in case of robot failure. For suppressing the error was used four techniques in this work. The tests confirm that according to expectations, error was best suppressed in algorithm with use of coordination, Particle filter and weighting. Big advantage of this solution is that Particle filters can be used with multimodal probability density distribution.

Suitable realization of multiagent structure presents modern and effective tool with various application possiblitity in the area of production sphere, information technologies, in economic systems, health services and so on.

Reached results can be applied into many various domains of applications for exploration and target localization. Results can serve as basis for future research, because use of multirobot system with prinicples of probabilistic robotics is in these days always new and open area of reseach and it is able to extend the limited horizon of one-agent systems.

References

[1] N. Agmon, N. Hazon, and G. Kaminka. Constructing spanning trees for efficient multi-robot coverage. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp 1698–1703, 2006.

[2] J. L. Blanco Claraco. Development of scientific applications with the mobile robot programming toolkit. Technical report, University of Malaga, 2010.

[3] B. Gerkey and R. Vaughan. The Player/Stage project: Tools for multi-robot and distributed sensor systems. In *Proceedings of the 11th International Conference on Advanced Robotics*, pp 120–125, Coimbra, Portugal, 2003.

[4] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEEE*

Proceedings, volume 140, pp 107–113, 1993.

[5] K. Lerman, M. Mataric, and A. Galstyan. Mathematical modeling of large multi-agent systems. Technical report, 2005.

[6] A. Martinoli, K. Easton, and W. Agassounon. Modeling swarm robotic systems: A case study in collaborative distributed manipulation. *International Journal of Robotics Research*, 23:415–436, 2004.

[7] J. Miček, M. Hyben, M. Frátrik, and J. Púchyová. Voice command recognition in multirobot systems: Information fusion. *International Journal of Advanced Robotic Systems*, pp 891–896, 2012.

[8] J. Púchyová and J. Miček. Development multirobot system. In *Automatics and informaticst'11- International conference*, pp B–185–B–187, 2011.

[9] K. Senthilkumar and K. Bharadwaj. Spanning tree based terrain coverage by multi robots in unknown environments. In *India Conf. INDICON 2008. Annual IEEE*, volume 1, pp 120–125, 2008.

Selected Papers by the Author

J. Púchyová. Behaviour of multiagent systems. In *Zimná škola MICT - Mathematics for information and communication technologies, 6th winter school of mathematics for ICT*, pp 84–86, Banská Bystrica: Science and Research Institute, MBU, 2011.

J. Púchyová. Simulation toolkits for multiagent system. In *TRANSCOM 2011: 9-th European conference of young research and scientific workers*, pp 201-204, June 27-29, 2011, Žilina: University of Žilina, 2011.

J. Púchyová, J. Miček. Development multirobot system. In *Automatics and informaticst'11- International conference*, pp B-185-B-187, Sofia, Bulgaria, October 3-7, 2011, John Atanasoff society of automatics and informatics, 2011.

J. Miček, J. Púchyová, M. Hyben. Speech communication in multirobot system. In *Proceedings of the VIth International scientific and technical conference Computer science and information technologies*, pp 255–257, November 16-19, 2011. Lviv, Ukraine. Lviv : Publishing House Vezha&Co, 2011.

J. Púchyová. Marking coverage algorithm with shortened return. *Journal of Information, Control and Management Systems*, Žilina, 10(1): 97–104, 2012.

J. Púchyová, M. Hyben. Command control of multirobot system. In *10th International conference Process control 2012*, 4 pp, C046b, Kouty nad Desnou, Czech Republic, June 11-14, 2012, Pardubice : University of Pardubice, 2012.

M. Húdik, J. Púchyová. Load balancing in grid with use of agents. In *10th International conference Process control 2012*, 4 pp, C046c, Kouty nad Desnou, Czech Republic, June 11-14, 2012, Pardubice : University of Pardubice, 2012.

J Púchyová. Exploration algorithm with shortened return for group of mobile robots. In *Proceedings of the VIIIth International Conference in MEMS design - MEMSTECHt' 2012 : Perspective technologies and methods in MEMS design*, pp 77-80, Polyana, Ukraine, April 18-21, 2012. Lviv: Publisher Lviv Polytechnic, 2012.

J. Papán, M. Jurečka, J. Púchyová. WSN for forest monitoring to prevent illegal logging. In *Proceedings of the IEEE Conference FedCSIS: Federated conference on computer science and information systems*,pp 809-812, Wrocşaw, Poland, September 9-12, 2012, IEEE: 2012.

J. Miček, M. Hyben, M. Frátrik, J. Púchyová, M. Hyben. Voice Command Recognition in Multirobot Systems: Information Fusion. *International Journal of Advanced Robotic Systems*, Aiguo Song (Ed.), Thomson Reuters, 2012.

M. Kochláň, M. Hodoň, J. Púchyová. Vital Functions Monitoring via Sensor Body Area Network with Smartphone Network Coordinator. In *Proceedings of the IXth International Conference on Perspective Technologies and Methods in MEMS Design - MEMSTECHt' 2013*, pp 143-147, Polyana, Ukraine, April 16-20, 2013.